Next-Generation Local Time Stepping for the ADER-DG Finite Element Method

IPDPS 2022

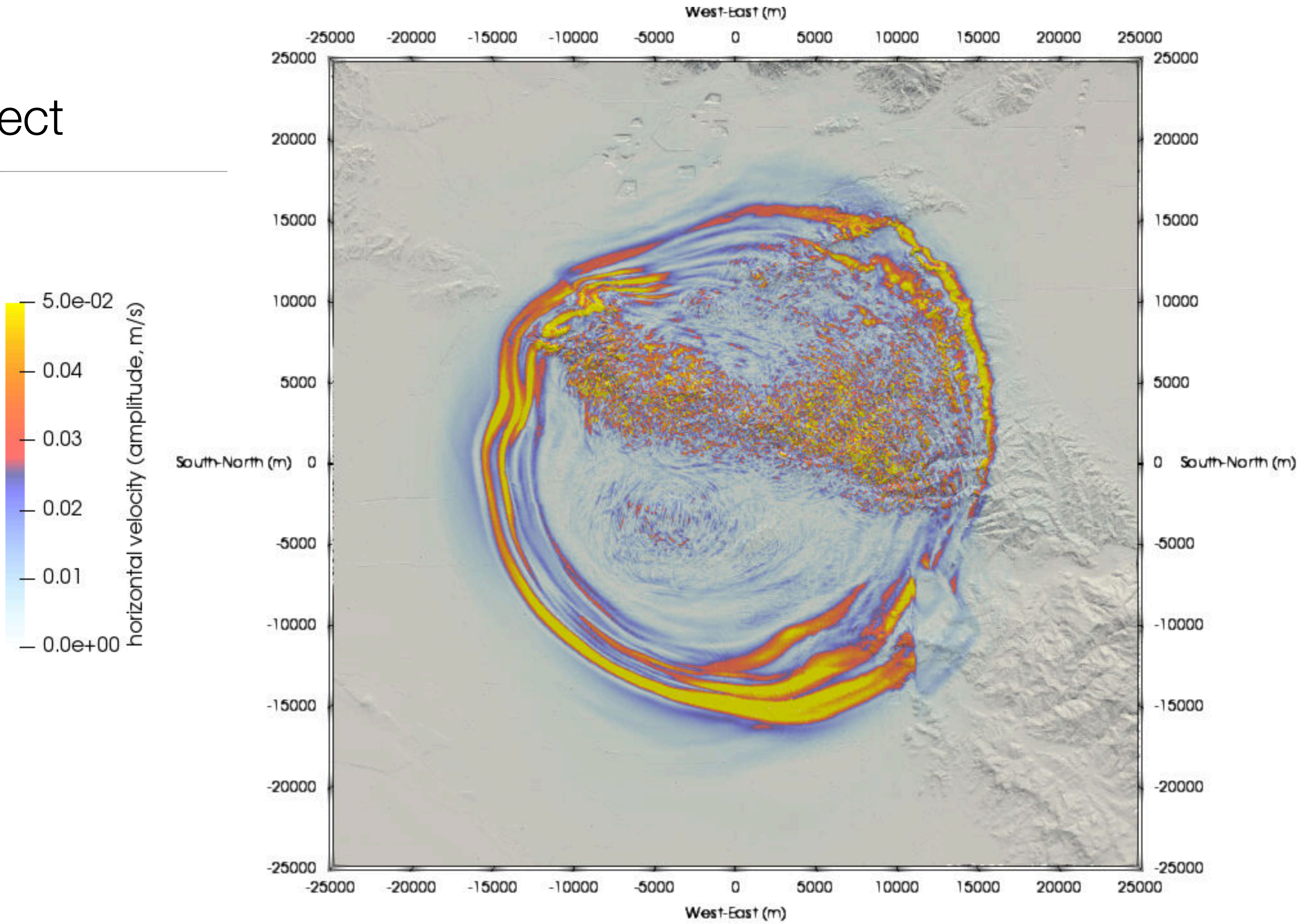
Alex Breuer (alex.breuer@uni-jena.de),
Alex Heinecke (alexander.heinecke@intel.com)

High-Frequency Ground Motion Simulations

Study area of the Southern California Earthquake Center's High Frequency project, simulating the 2014 Mw 5.1 La Habra, California earthquake. Shown are the "small" domain through the inner box and partially the "large" domain through the outer box. Additionally, the locations of the earthquake's epicenter and three stations are given.

The screenshot was obtained from: http://u.osmfr.org/m/560152/.

High-F Project

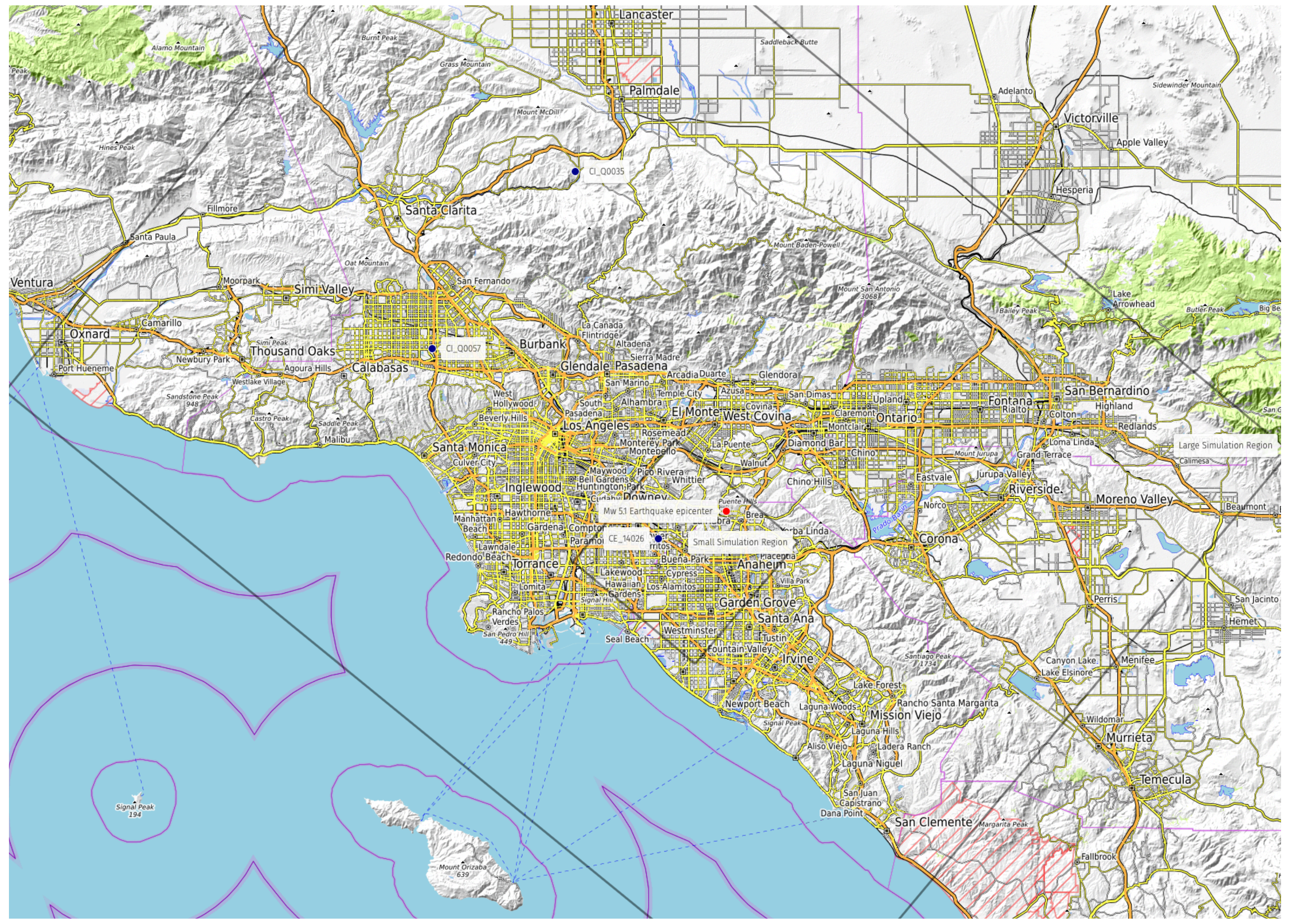


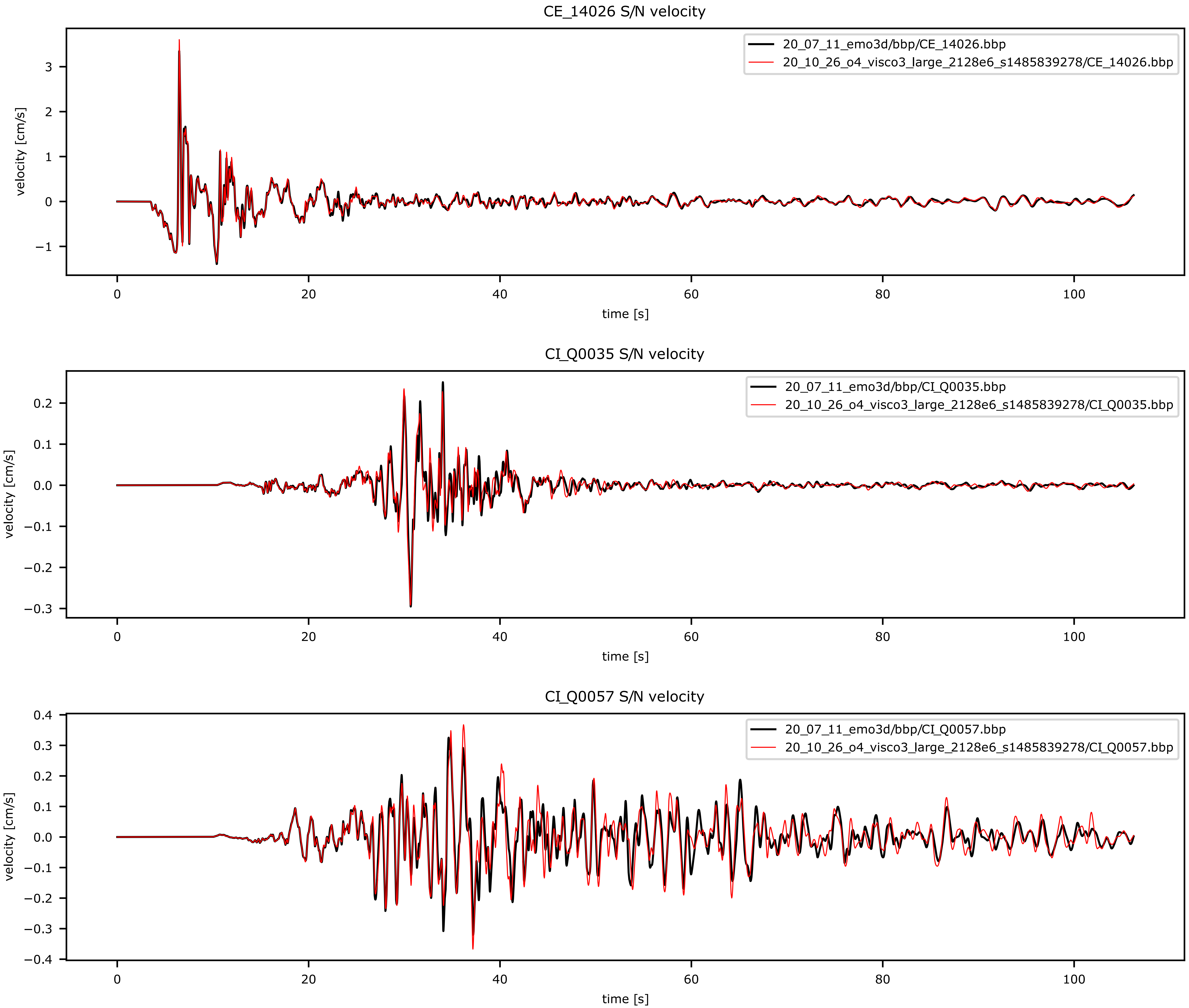
Visualization of the seismic wave field for a simulation of the 2014 Mw 5.1 La Habra Earthquake. Shown are the amplitudes of the horizontal particle velocities after seven seconds of simulated time.

Time: 0.00s
7.5e-02
0.06
0.04
0.02
0.0e+00
particle velocity (amplitude, m/s)

Verification

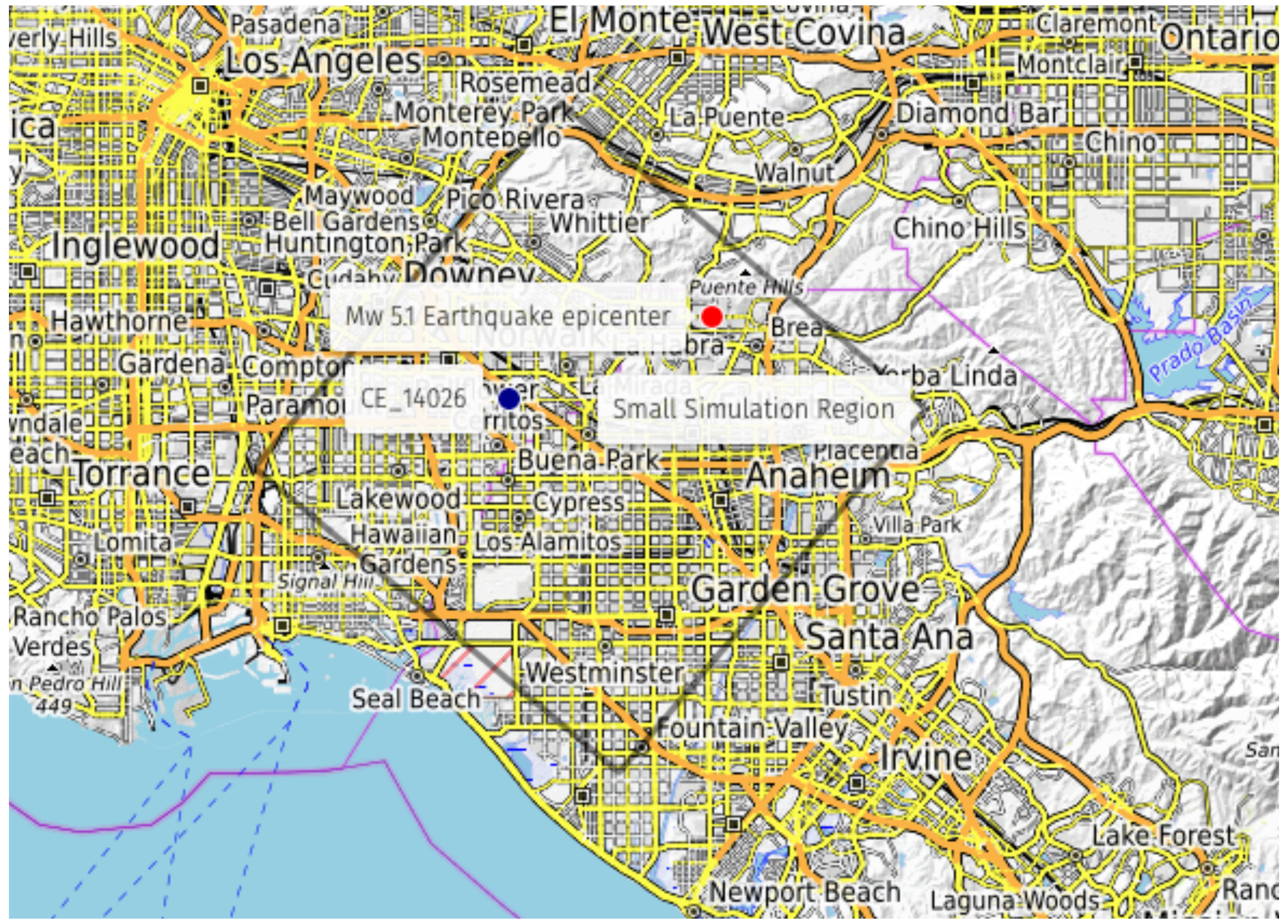
Right figure: Comparison of EDGE's South-North velocity component (red) to EMO3D's High-F solution (black). Shown are synthetic seismograms for the three stations depicted below. The seismograms were low-pass filtered at 5Hz. EDGE's respective ground motion simulation harnessed 1,536 nodes of the Frontera machine for a total of 48 hours to advance the used 2.1 billion tetrahedral elements in time.

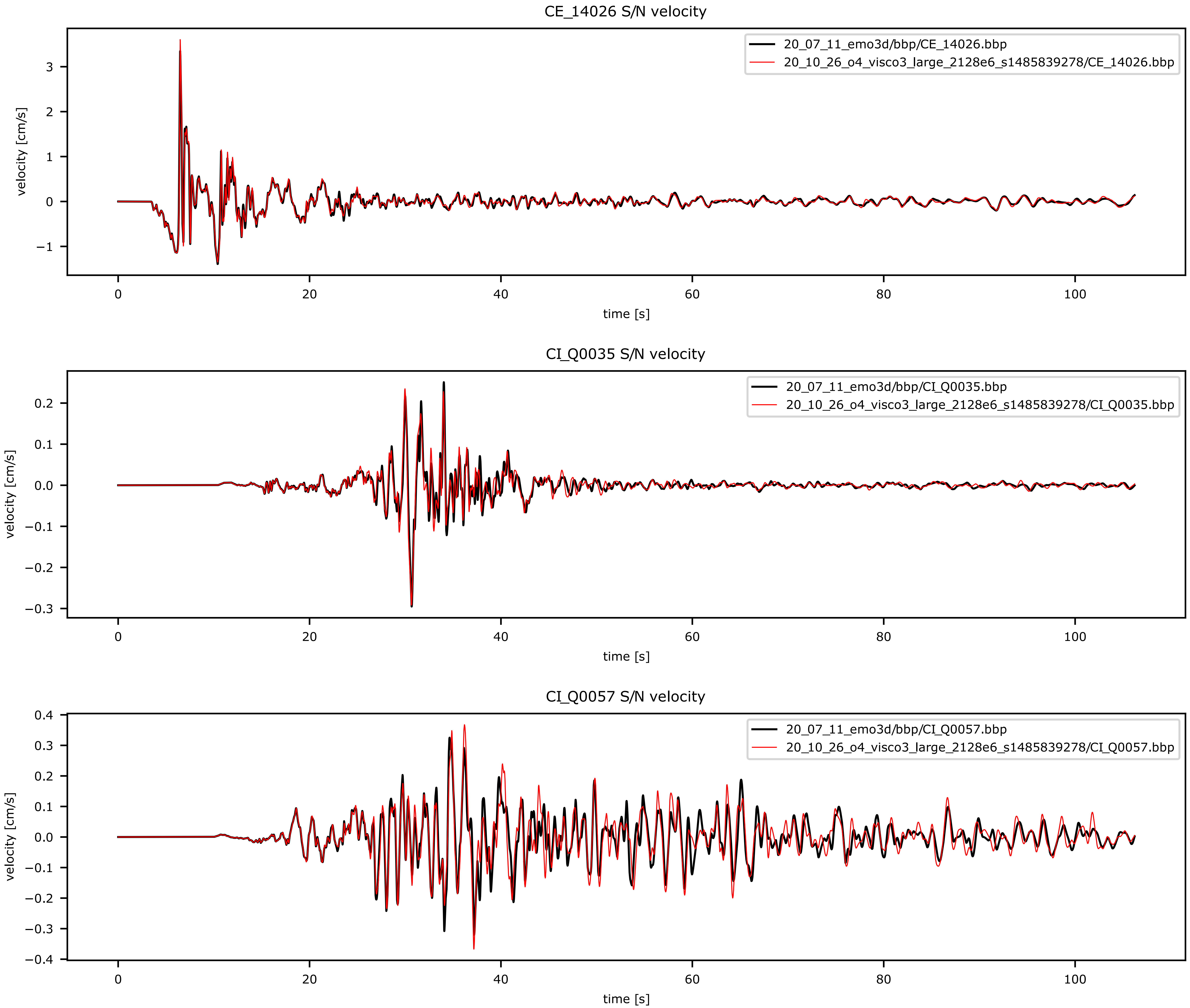




Verification

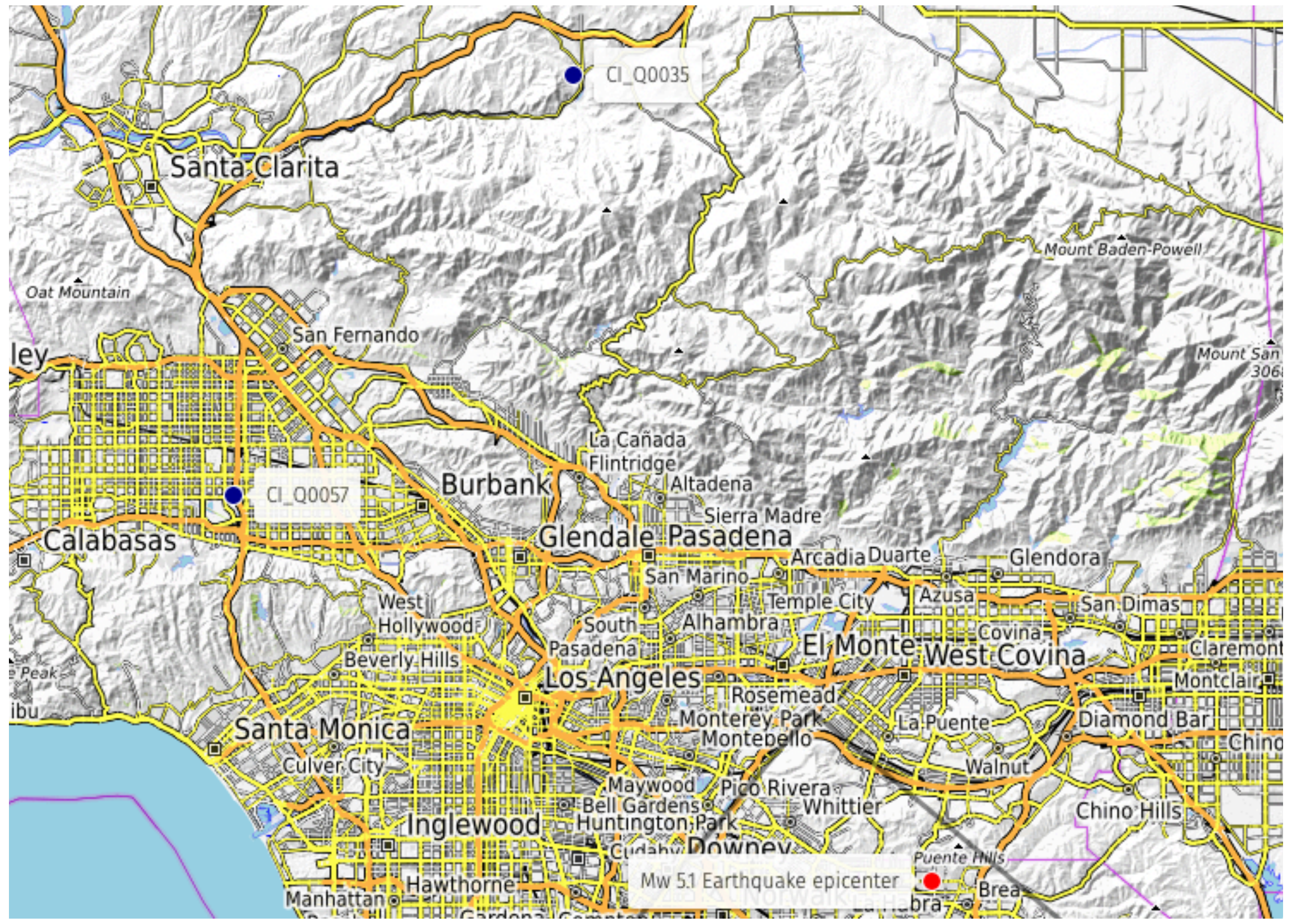
Right figure: Comparison of EDGE's South-North velocity component (red) to EMO3D's High-F solution (black). Shown are synthetic seismograms for the three stations depicted below. The seismograms were low-pass filtered at 5Hz. EDGE's respective ground motion simulation harnessed 1,536 nodes of the Frontera machine for a total of 48 hours to advance the used 2.1 billion tetrahedral elements in time.

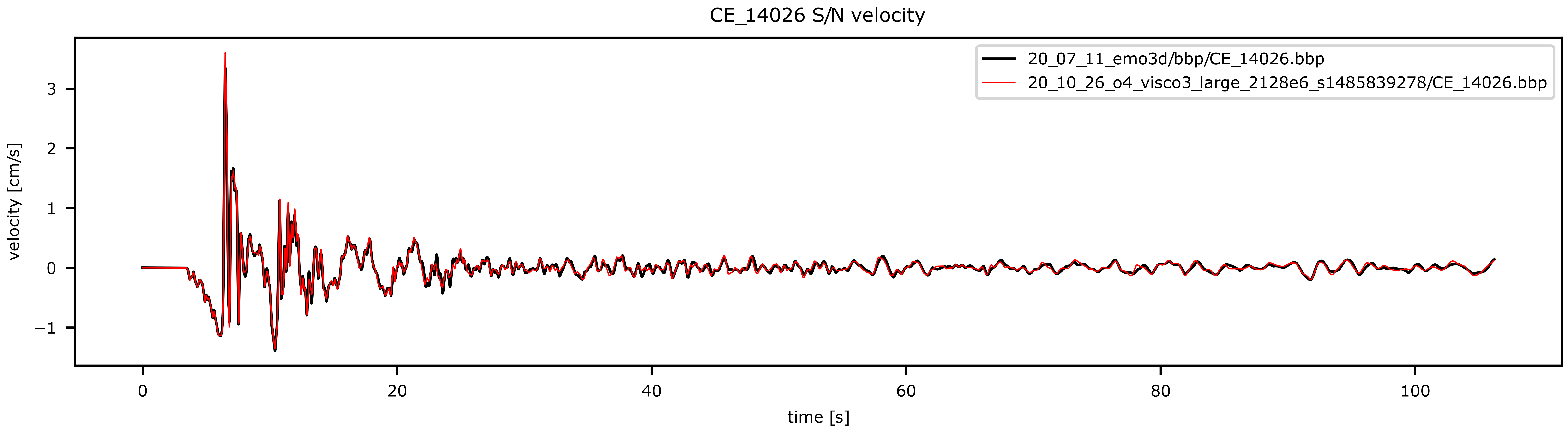


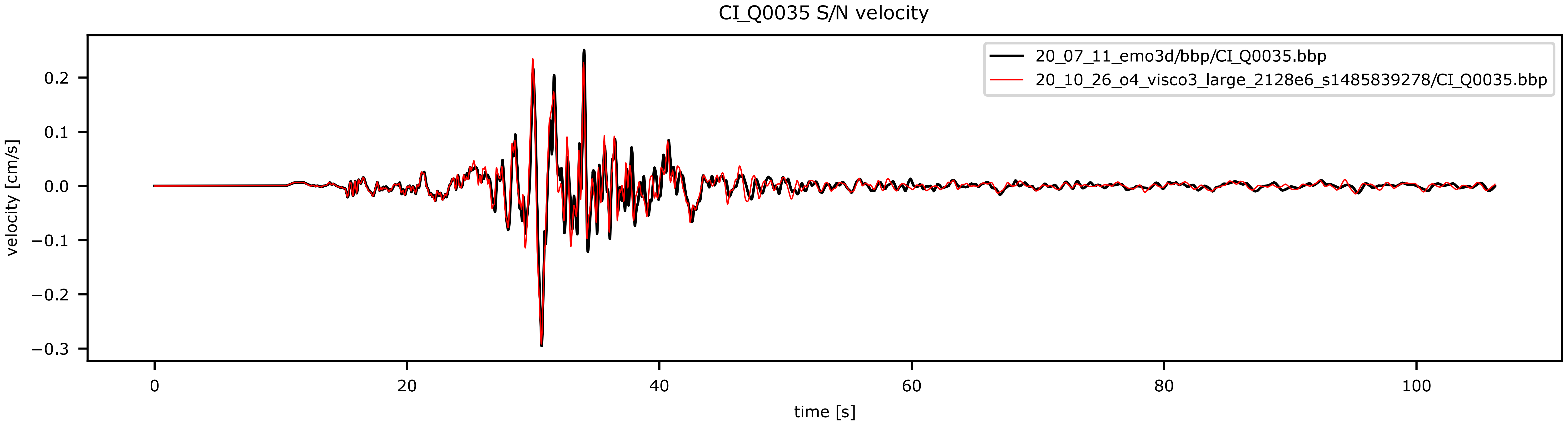


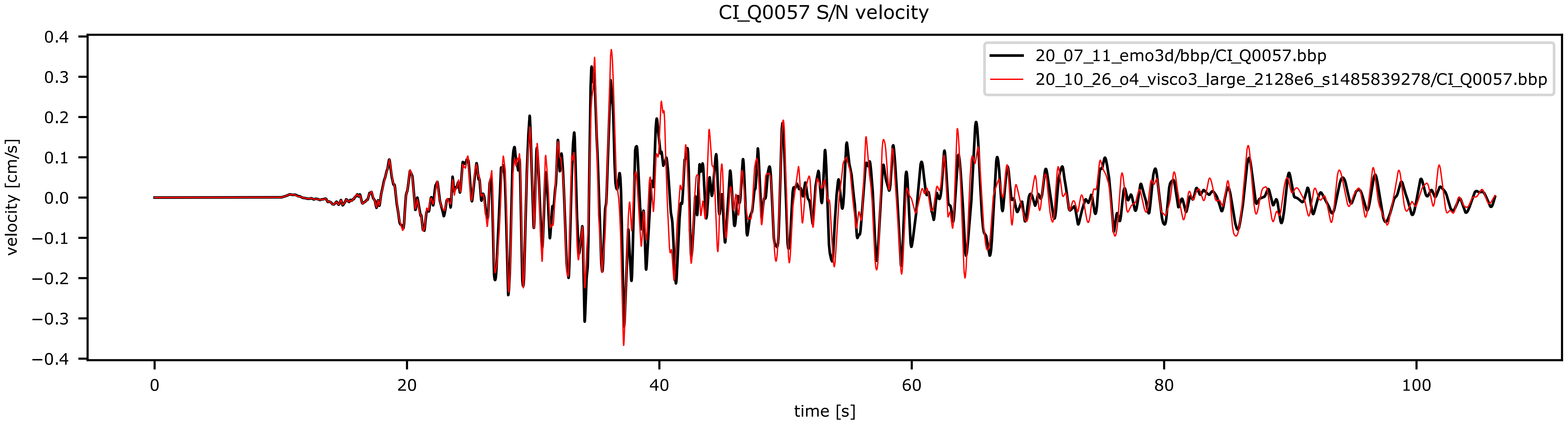
Verification

Right figure: Comparison of EDGE’s South-North velocity component (red) to EMO3D’s High-F solution (black). Shown are synthetic seismograms for the three stations depicted below. The seismograms were low-pass filtered at 5Hz. EDGE’s respective ground motion simulation harnessed 1,536 nodes of the Frontera machine for a total of 48 hours to advance the used 2.1 billion tetrahedral elements in time.



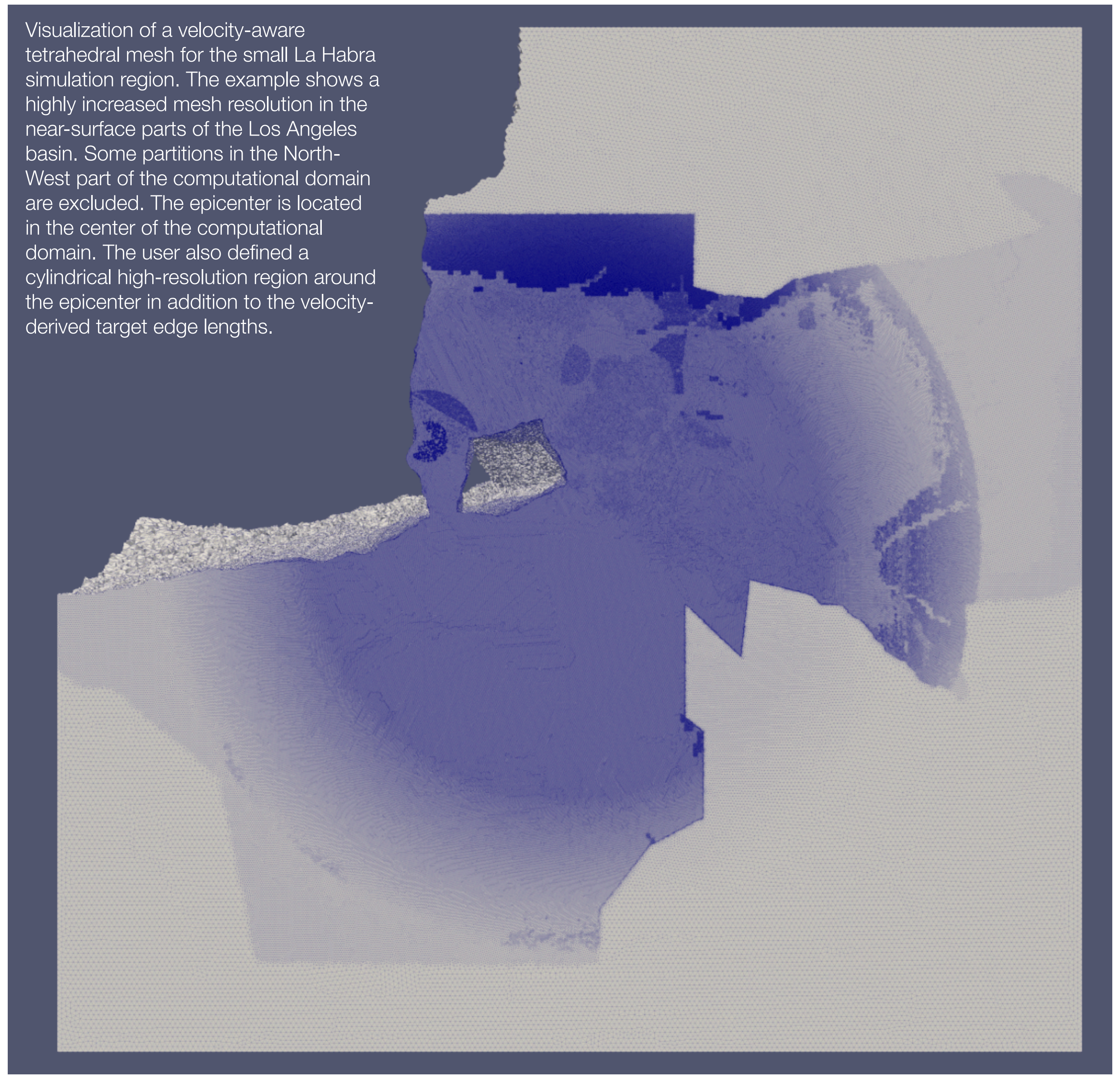






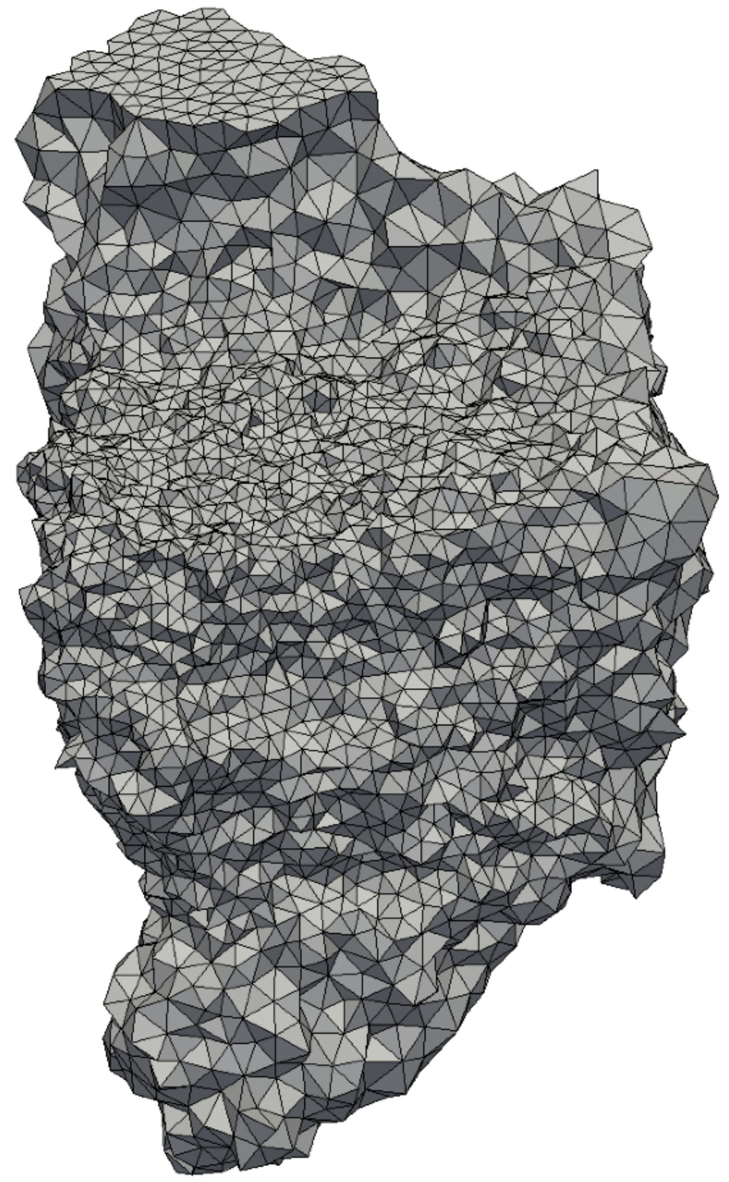
Extreme Scale Discontinuous Galerkin Environment (EDGE)

- Uses ADER Discontinuous Galerkin Finite Element Method with tetrahedral elements
- Focus: Static meshes with high geometric complexity
- Unique support for fused simulations exploiting inter-simulation parallelism
- Parallelization: JITted kernels for highest performance on many recent CPU architectures (AVX, AVX2, AVX512, ASIMD, SVE); advanced OpenMP+MPI for hidden communication
- Extremely scalable with sustained petascale performance: 10.4 FP64 PFLOPS on Cori II, 1.1 FP32 PFLOPS in AWS
- Supporting tools for surface meshing, constrained velocity-aware volume meshing and partitioning
- Core solver and all tools are open source software (BSD-3), modeling and simulation pipeline relies exclusively on open source software (https://dial3343.org)



Visualization of a velocity-aware tetrahedral mesh for the small La Habra simulation region. The example shows a highly increased mesh resolution in the near-surface parts of the Los Angeles basin. Some partitions in the North-West part of the computational domain are excluded. The epicenter is located in the center of the computational domain. The user also defined a cylindrical high-resolution region around the epicenter in addition to the velocity-derived target edge lengths.

Exemplary illustration of an MPI-partition for an unstructured tetrahedral mesh.



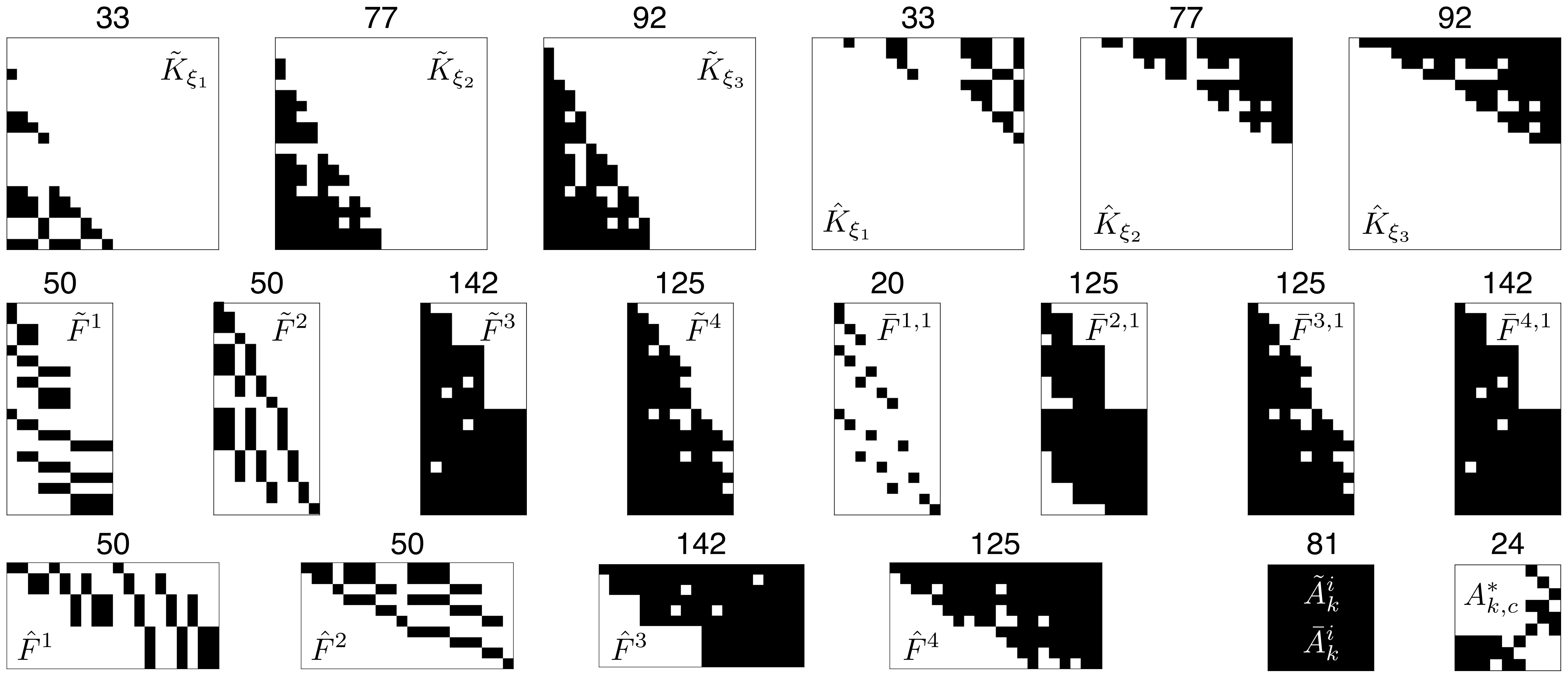


Illustration of all involves sparsity patterns for a fourth order ADER-DG discretization in EDGE. The numbers on top give the non-zero entries in the sparse matrices.

Next-Generation Local Time Stepping

Challenge: Heterogeneous Time Steps

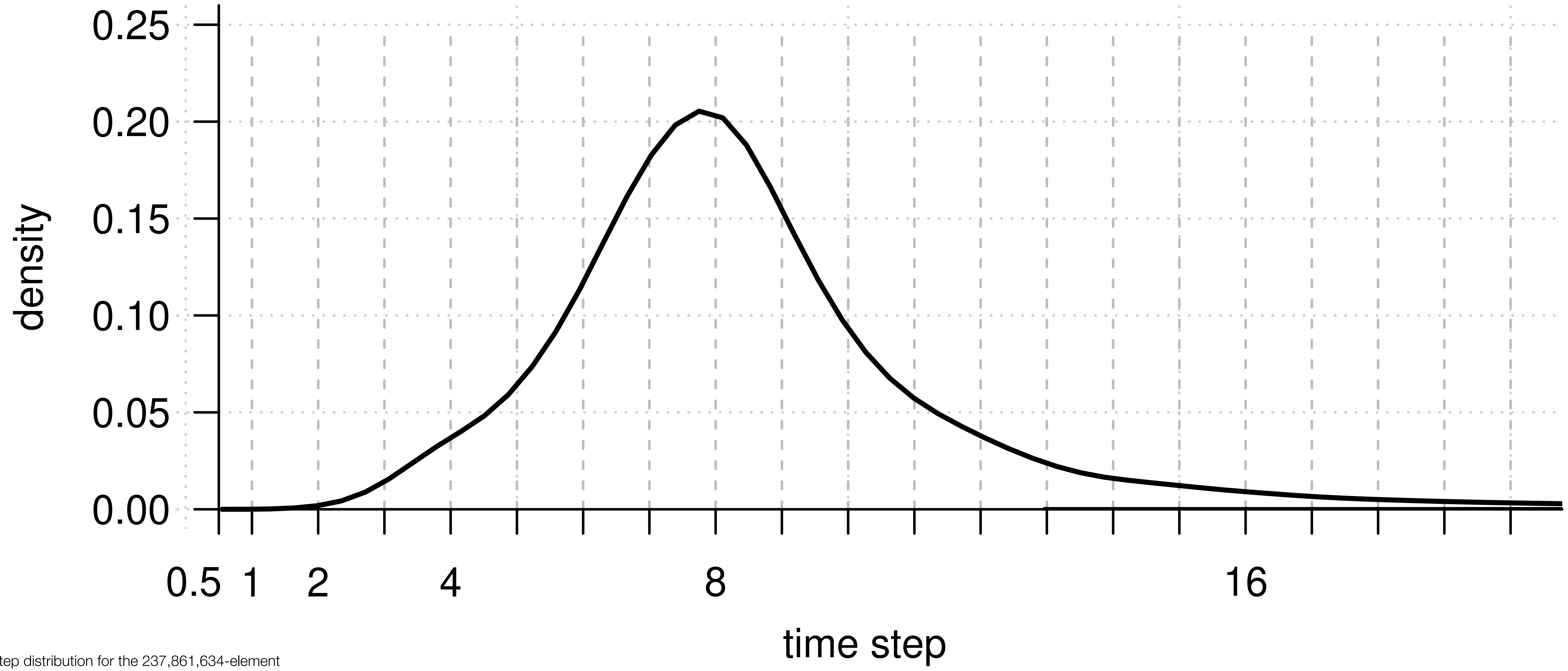


Illustration of the time step distribution for the 237,861,634-element 2014 Mw 5.1 La Habra setting. The solid line shows the time step density of the mesh elements. The time step relative to $\Delta t^{\mathrm{CFL}}_{\mathrm{min}}$ is given on the x-axis and the element density on the y-axis.

Solution: Next-Generation Local Time Stepping Scheme

$$
\begin{aligned}
C_1 &= [\lambda \cdot \Delta t_{\min}^{\mathrm{CFL}}, & 2\lambda \cdot \Delta t_{\min}^{\mathrm{CFL}}),\\
C_2 &= [2\lambda \cdot \Delta t_{\min}^{\mathrm{CFL}}, & 2^2\lambda \cdot \Delta t_{\min}^{\mathrm{CFL}}),\\
&\cdots\\
C_{N_c} &= [2^{N_c-1}\lambda \cdot \Delta t_{\min}^{\mathrm{CFL}}, & \infty).
\end{aligned}
$$

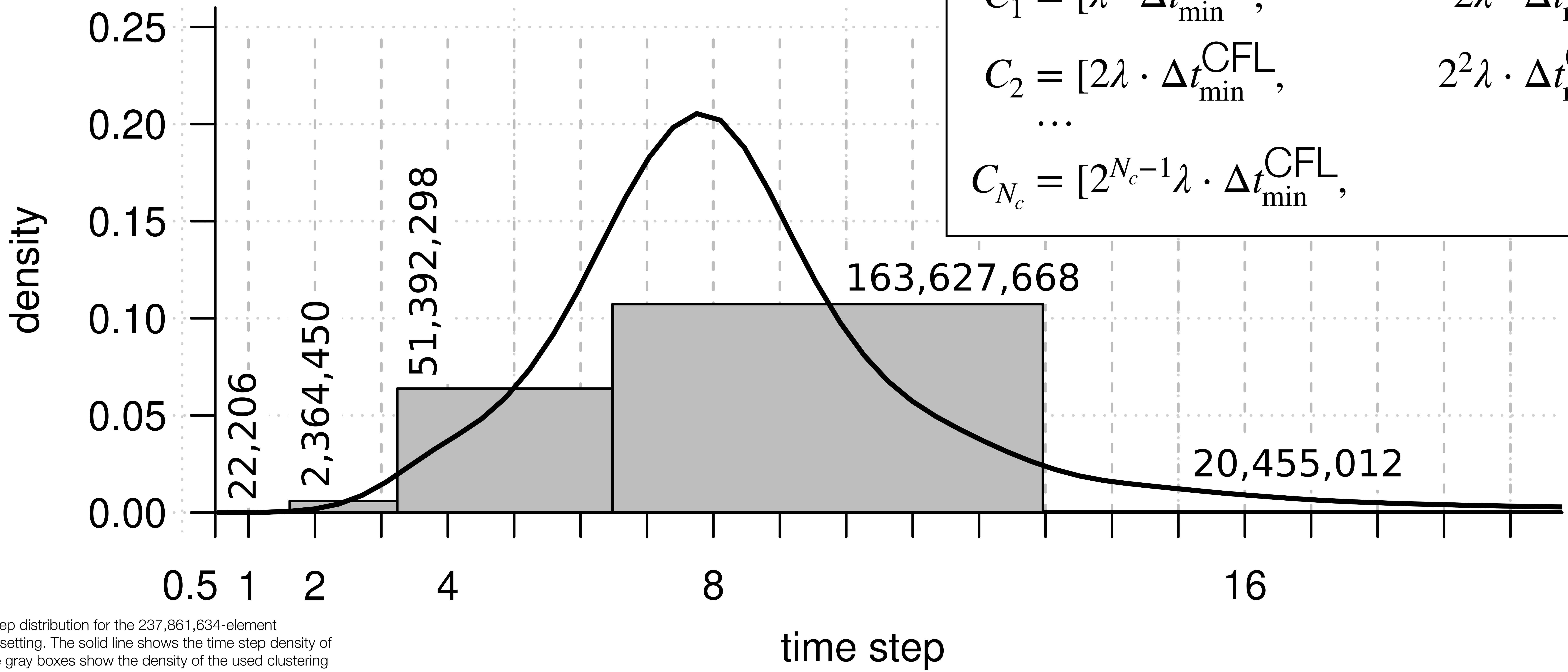


Illustration of the time step distribution for the 237,861,634-element 2014 Mw 5.1 La Habra setting. The solid line shows the time step density of the mesh elements. The gray boxes show the density of the used clustering scheme with $\lambda = 0.81$. On top of each cluster the number of contained elements is given.The time step relative to $\Delta t_{\min}^{\mathrm{CFL}}$ is given on the x-axis and the element density on the y-axis.

Parallel LTS Scheme

- Developed tailored data layout to share an element's time integrated DOFs required by face-neighboring elements; options:
  - Neighbor has same time step (1x)
  - Neighbor has smaller time step (0.5x)
  - Neighbor has larger time step (2x)
- Designed efficient parallelization for shared and distributed memory domain
- Added LTS support to preprocessing:
  - Derive optimal parameter $\lambda \in]0.5,1]$
  - Mesh partitioning considers weights for elements (computation) and faces (communication)
  - Reorder elements for partitioning and LTS algorithm

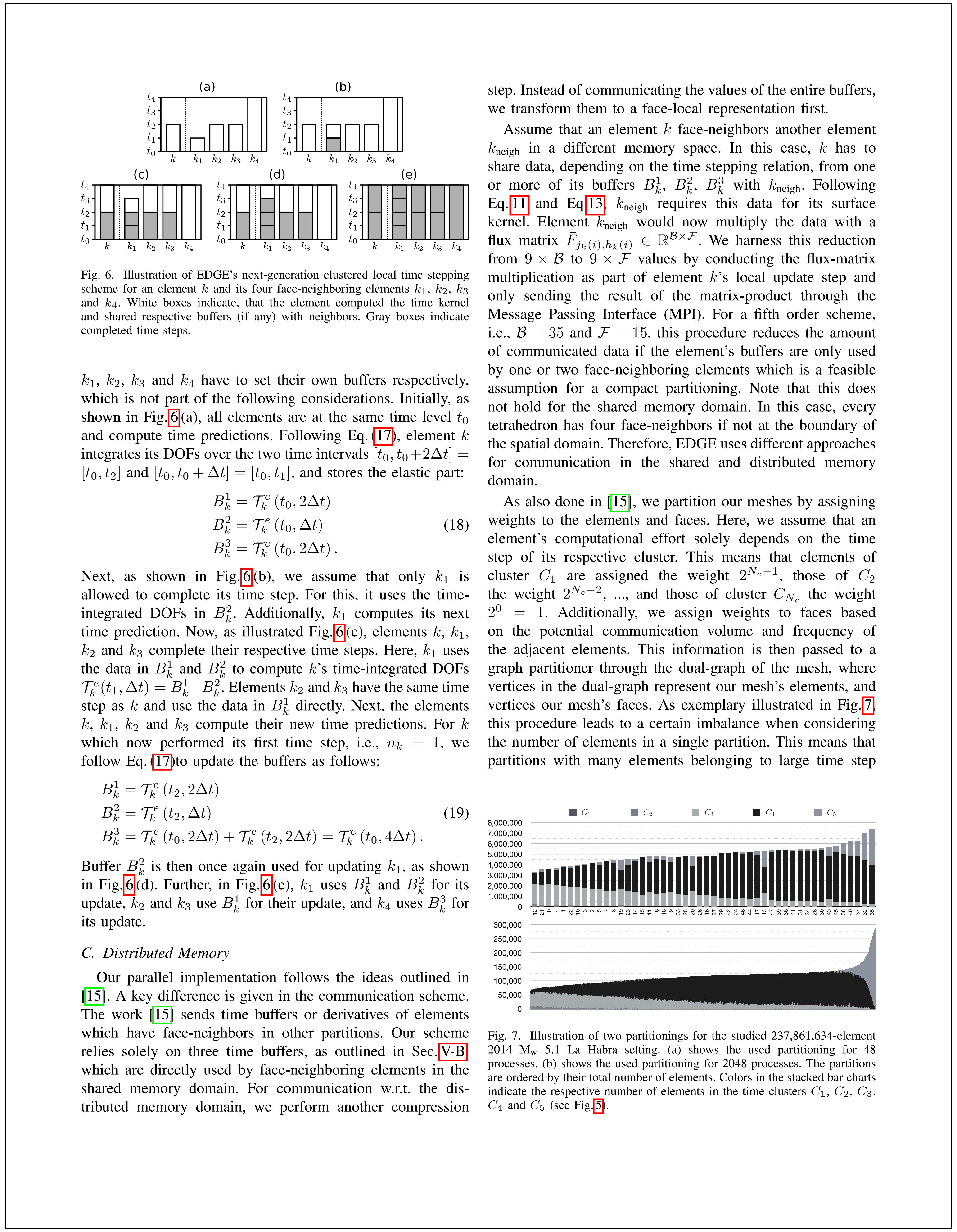


Fig. 6. Illustration of EDGE's next-generation clustered local time stepping scheme for an element k and its four face-neighboring elements k_1, k_2, k_3 and k_4. White boxes indicate, that the element computed the time kernel and shared respective buffers (if any) with neighbors. Gray boxes indicate completed time steps.

k_1, k_2, k_3 and k_4 have to set their own buffers respectively, which is not part of the following considerations. Initially, as shown in Fig. 6(a), all elements are at the same time level t_0 and compute time predictions. Following Eq. (17), element k integrates its DOFs over the two time intervals $[t_0, t_0+2\Delta t] = [t_0, t_2]$ and $[t_0, t_0 + \Delta t] = [t_0, t_1]$, and stores the elastic part:

$$
\begin{aligned}
B_k^1 &= \mathcal{T}_k^e\,(t_0, 2\Delta t)\\
B_k^2 &= \mathcal{T}_k^e\,(t_0, \Delta t)\\
B_k^3 &= \mathcal{T}_k^e\,(t_0, 2\Delta t)\,.
\end{aligned}
\tag{18}
$$

Next, as shown in Fig. 6(b), we assume that only k_1 is allowed to complete its time step. For this, it uses the time-integrated DOFs in B_k^2. Additionally, k_1 computes its next time prediction. Now, as illustrated Fig. 6(c), elements k, k_1, k_2 and k_3 complete their respective time steps. Here, k_1 uses the data in B_k^1 and B_k^2 to compute k's time-integrated DOFs $\mathcal{T}_k^e(t_1, \Delta t) = B_k^1 - B_k^2$. Elements k_2 and k_3 have the same time step as k and use the data in B_k^1 directly. Next, the elements k, k_1, k_2 and k_3 compute their new time predictions. For k which now performed its first time step, i.e., $n_k = 1$, we follow Eq. (17)to update the buffers as follows:

$$
\begin{aligned}
B_k^1 &= \mathcal{T}_k^e\,(t_2, 2\Delta t)\\
B_k^2 &= \mathcal{T}_k^e\,(t_2, \Delta t)\\
B_k^3 &= \mathcal{T}_k^e\,(t_0, 2\Delta t) + \mathcal{T}_k^e\,(t_2, 2\Delta t) = \mathcal{T}_k^e\,(t_0, 4\Delta t)\,.
\end{aligned}
\tag{19}
$$

Buffer B_k^2 is then once again used for updating k_1, as shown in Fig. 6(d). Further, in Fig. 6(e), k_1 uses B_k^1 and B_k^2 for its update, k_2 and k_3 use B_k^1 for their update, and k_4 uses B_k^3 for its update.

C. Distributed Memory

Our parallel implementation follows the ideas outlined in [15]. A key difference is given in the communication scheme. The work [15] sends time buffers or derivatives of elements which have face-neighbors in other partitions. Our scheme relies solely on three time buffers, as outlined in Sec. V-B, which are directly used by face-neighboring elements in the shared memory domain. For communication w.r.t. the distributed memory domain, we perform another compression step. Instead of communicating the values of the entire buffers, we transform them to a face-local representation first.

Assume that an element k face-neighbors another element k_{neigh} in a different memory space. In this case, k has to share data, depending on the time stepping relation, from one or more of its buffers B_k^1, B_k^2, B_k^3 with k_{neigh}. Following Eq. 11 and Eq. 13, k_{neigh} requires this data for its surface kernel. Element k_{neigh} would now multiply the data with a flux matrix $\bar{F}_{j_k(i),h_k(i)} \in \mathbb{R}^{\mathcal{B}\times\mathcal{F}}$. We harness this reduction from $9 \times \mathcal{B}$ to $9 \times \mathcal{F}$ values by conducting the flux-matrix multiplication as part of element k's local update step and only sending the result of the matrix-product through the Message Passing Interface (MPI). For a fifth order scheme, i.e., $\mathcal{B} = 35$ and $\mathcal{F} = 15$, this procedure reduces the amount of communicated data if the element's buffers are only used by one or two face-neighboring elements which is a feasible assumption for a compact partitioning. Note that this does not hold for the shared memory domain. In this case, every tetrahedron has four face-neighbors if not at the boundary of the spatial domain. Therefore, EDGE uses different approaches for communication in the shared and distributed memory domain.

As also done in [15], we partition our meshes by assigning weights to the elements and faces. Here, we assume that an element's computational effort solely depends on the time step of its respective cluster. This means that elements of cluster C_1 are assigned the weight 2^{N_c-1}, those of C_2 the weight 2^{N_c-2}, ..., and those of cluster C_{N_c} the weight $2^0 = 1$. Additionally, we assign weights to faces based on the potential communication volume and frequency of the adjacent elements. This information is then passed to a graph partitioner through the dual-graph of the mesh, where vertices in the dual-graph represent our mesh's elements, and vertices our mesh's faces. As exemplary illustrated in Fig. 7, this procedure leads to a certain imbalance when considering the number of elements in a single partition. This means that partitions with many elements belonging to large time step

Fig. 7. Illustration of two partitionings for the studied 237,861,634-element 2014 M_w 5.1 La Habra setting. (a) shows the used partitioning for 48 processes. (b) shows the used partitioning for 2048 processes. The partitions are ordered by their total number of elements. Colors in the stacked bar charts indicate the respective number of elements in the time clusters C_1, C_2, C_3, C_4 and C_5 (see Fig. 5).

Performance

Single CPU Performance

- Single processor of Frontera: 28-core Intel Xeon Platinum 8280
- Seismic wave propagation setting using viscoelastic attenuation (LOH3)
- ADER-DG (order 5) with 743,066 tetrahedral elements
- Comparison to solver SeisSol which implements 1st geneneration LTS scheme designed for elastic wave equations [IPDPS16]
- Theoretical speedups for LTS:
  - $\lambda = 1.0$: $2.28\times$
  - $\lambda = 0.80$: $2.67\times$

Performance of the two solvers EDGE and SeisSol on a single processor of Frontera when running the 743,066-element LOH.3 setting. The FP32 TFLOPS in hardware and the speedups over EDGE's single simulation GTS performance are given.

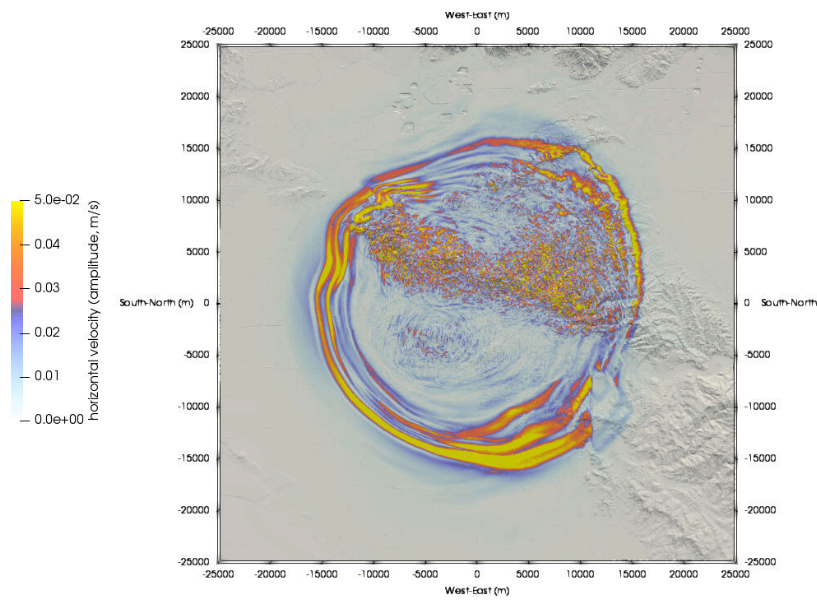
Solver	**Metric**	**GTS**		**LTS (1.0)**		**LTS (0.8)**	
		1	***16***	***1***	***16***	***1***	***16***
EDGE	TFLOPS	1.08	0.78[a]	1.01	0.74[a]	1.02	0.74[a]
SeisSol	TFLOPS[b]	1.34	–	1.09	–	–	–
EDGE	speedup	1.00	1.80	2.14	3.91	2.51	4.51
SeisSol	speedup	0.92	–	1.70	–	–	–

[a] EDGE's fused simulations use sparse matrix kernels.

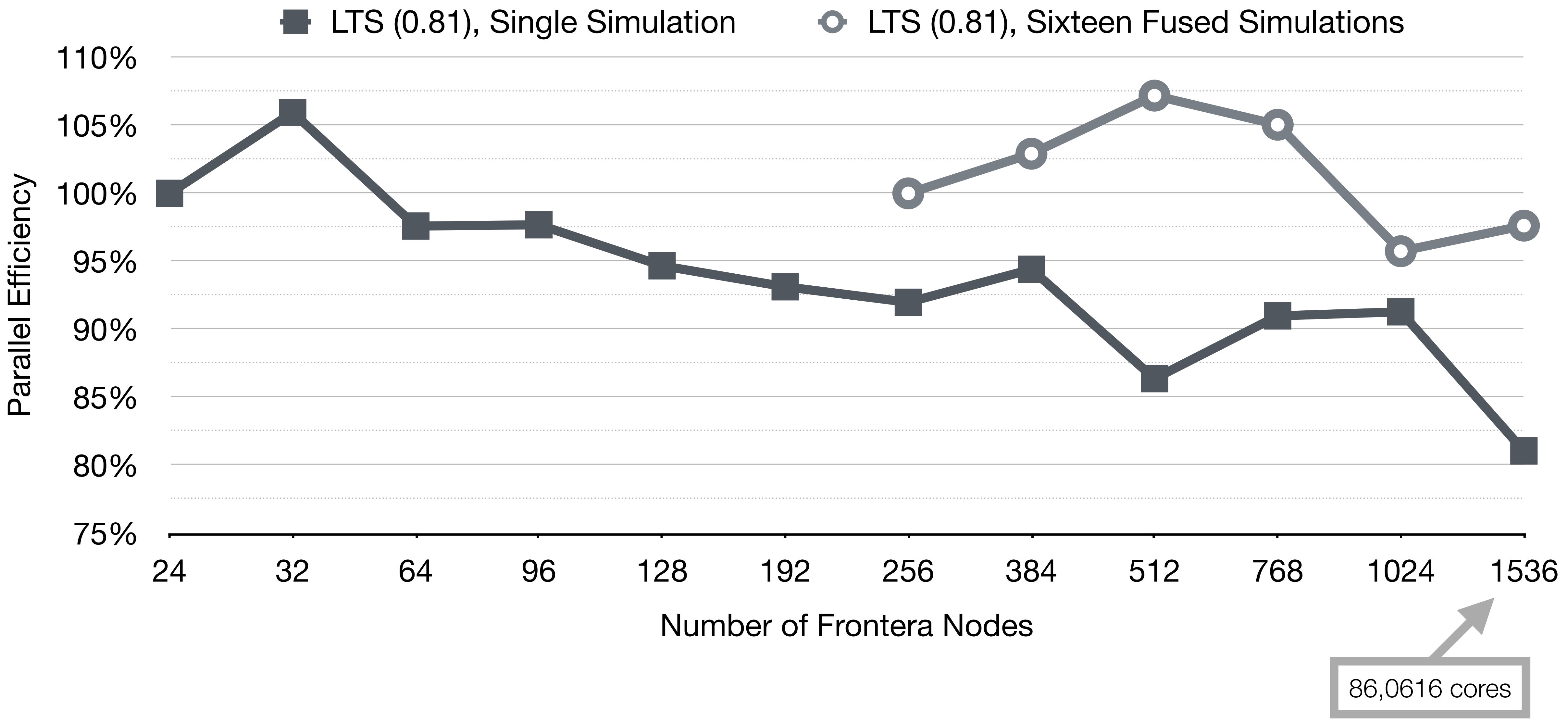
[b] We report SeisSol's hardware performance as printed by the solver itself.

[IPDPS16] A. Breuer, A. Heinecke, and M. Bader, "Petascale Local Time Stepping for the ADER-DG Finite Element Method" in 2016 IEEE international parallel and distributed processing symposium (IPDPS). IEEE, 2016, pp. 854-863.

Strong Scalability

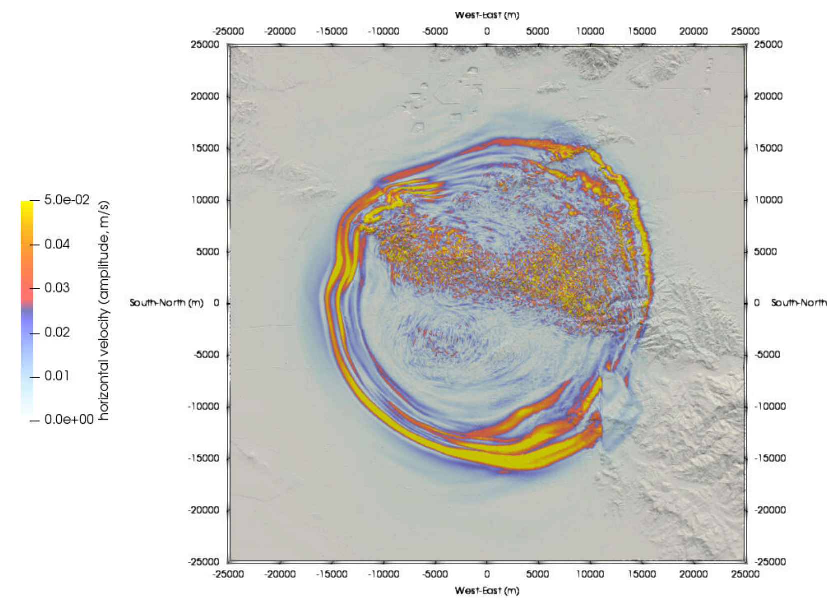


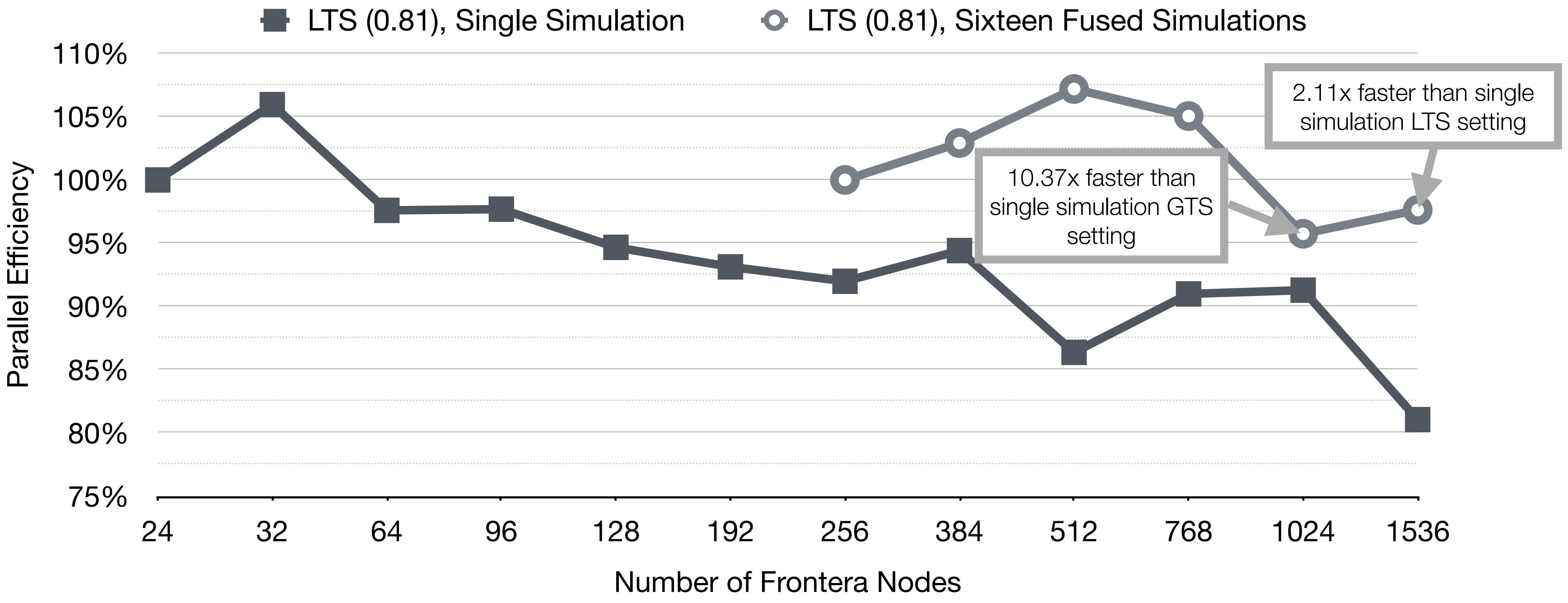
- Setup for the 2014 Mw 5.1 La Habra earthquake
- Model enhancements over High-F version used for verification:
  - Incorporated topography information
  - Reduced cutoff for minimal shear wave velocity from 500m/s to 250m/s
- 237,861,634 tetrahedral elements, fifth order ADER-DG
- Sustained LTS performance on Frontera:
  - 2.25 FP32 PFLOPS for single simulation (includes zero ops)
  - 1.91 FP32 PFLOPS for fused simulations (sparse kernels, no zero ops)

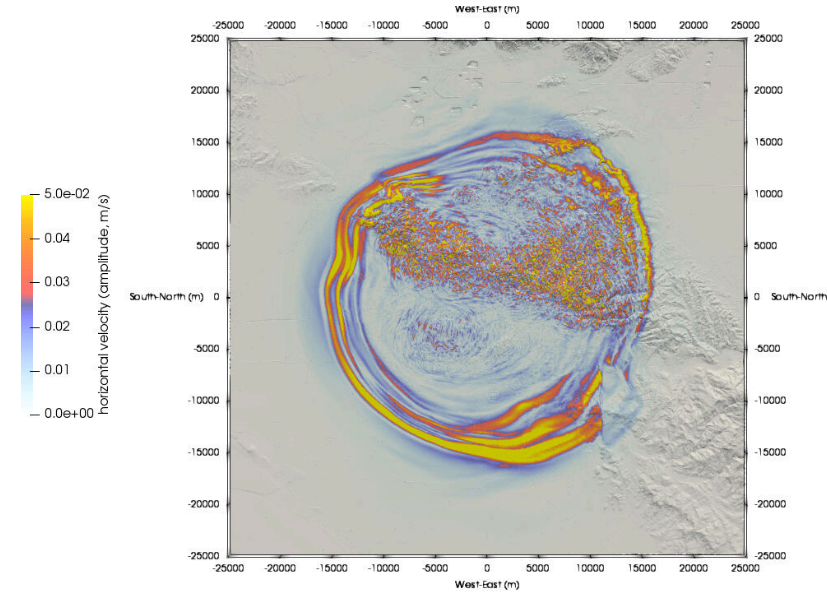


Strong Scalability

- Setup for the 2014 Mw 5.1 La Habra earthquake
- Model enhancements over High-F version used for verification:
  - Incorporated topography information
  - Reduced cutoff for minimal shear wave velocity from 500m/s to 250m/s
- 237,861,634 tetrahedral elements, fifth order ADER-DG
- Sustained LTS performance on Frontera:
  - 2.25 FP32 PFLOPS for single simulation (includes zero ops)
  - 1.91 FP32 PFLOPS for fused simulations (sparse kernels, no zero ops)





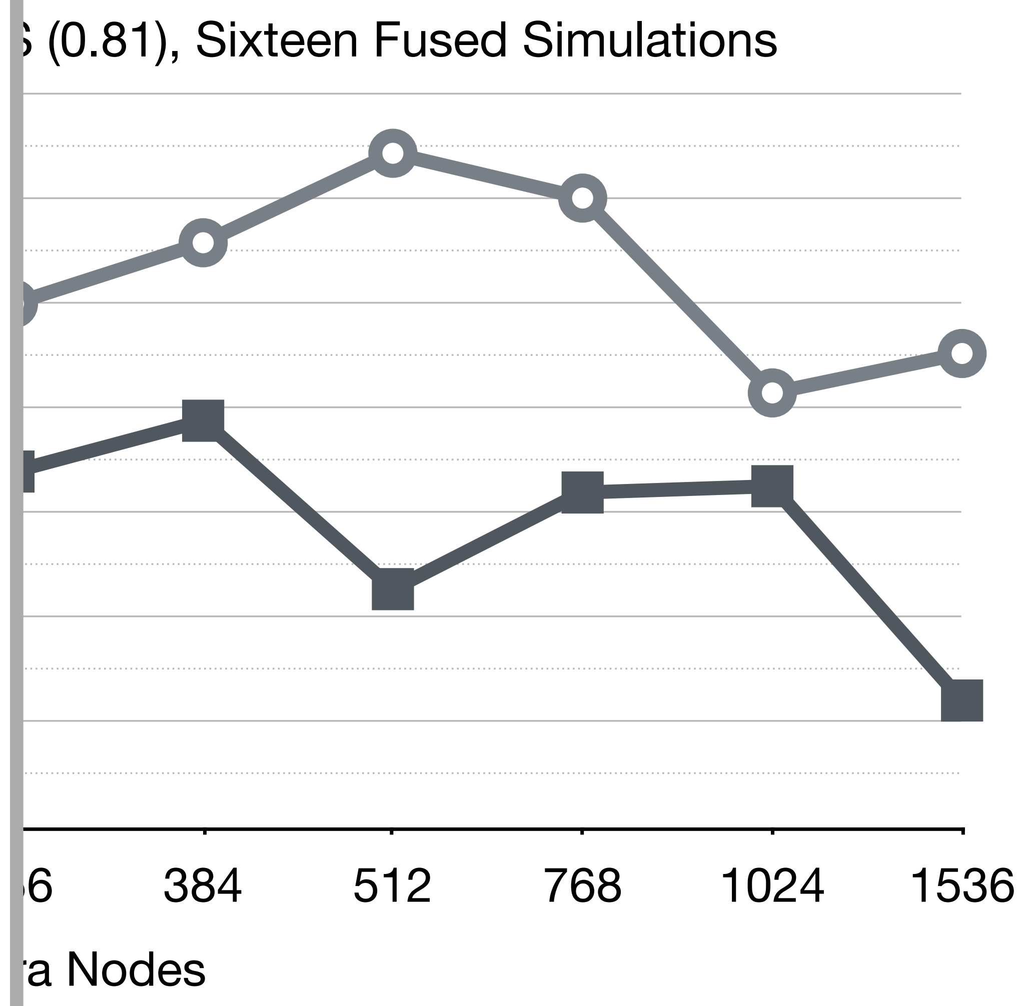


Strong Scalability

- Setup for the 2014 Mw 5.1 La Habra earthquake
- Model enhancements over High-F version used for verification:
  - Incorporated topography i
  - Reduced cutoff for minima from 500m/s to 250m/s
- 237,861,634 tetrahedral elem DG
- Sustained LTS performance
  - 2.25 FP32 PFLOPS for single simulation (includes zero ops)
  - 1.91 FP32 PFLOPS for fused simulations (sparse kernels, no zero ops)

Strong scaling was basis for a series of follow-up simulations with 16 fused kinematic sources (not part of IPDPS22 work):

Min vs	#Elements	Theoretical LTS Speedup	#Frontera Nodes	Runtime
250m/s	242,595,220	9.92	128	12.23h
150m/s	434,379,012	10.76	768	19.00h
100m/s	421,290,625	6.53	2,048	14.86h



The Many Details

Contributions

- Incorporated anelastic wave equations into solver EDGE
- Designed new local time stepping scheme for highly efficient simulations with viscoelastic attenuation
- Developed new communication scheme minimizing the pressure on the memory and network
- Introduced end-to-end preprocessing pipeline which enables efficient and large scale high-frequency ground motion simulations

Next-Generation Local Time Stepping for the ADER-DG Finite Element Method

1st Alexander Breuer
Friedrich Schiller University Jena
Jena, Germany
alex.breuer@uni-jena.de

2nd Alexander Heinecke
Intel Corporation
Santa Clara, USA
alexander.heinecke@intel.com

***Abstract*—High-frequency ground motion simulations pose a grand challenge in computational seismology. Two main factors drive this challenge. First, to account for higher frequencies, we have to extend our numerical models, e.g., by considering anelasticity, or by including mountain topography. Second, even if we were able to keep our models unchanged, simply doubling the frequency content of a seismic wave propagation solver requires a sixteen-fold increase in computational resources due to the used four-dimensional space-time domains.**

This work presents the Extreme Scale Discontinuous Galerkin Environment (EDGE) in the context of high-frequency ground motion simulations. Our presented enhancements cover the entire spectrum of the unstructured finite element solver. This includes the incorporation of anelasticity, the introduction of a next-generation clustered local time stepping scheme, and the introduction of a completely revised communication scheme. We close the modeling and simulation loop by presenting our new and rich preprocessing, which drives the high problem-awareness and numerical efficiency of the core solver.

In summary, the presented work allows us to conduct large scale high-frequency ground motion simulations efficiently, routinely and conveniently. The soundness of our work is underlined by a set of high-frequency verification runs using a realistic setting. We conclude the presentation by studying EDGE's combined algorithmic and computational efficiency in a demanding setup of the 2014 M_w 5.1 La Habra earthquake. Our results are compelling and show an improved time-to-solution by over 10× while scaling strongly from 256 to 1,536 nodes of the Frontera supercomputer with a sustained non-zero performance of 1.91 FP32-PFLOPS.

***Index Terms*—local time stepping, ADER-DG, unstructured meshes, large scale simulations, seismic wave propagation, anelasticity**

I. 2014 Mw 5.1 La Habra Earthquake

The presented work uses the Extreme Scale Discontinuous Galerkin Environment (EDGE) to tackle the grand challenge of high-frequency ground motion simulations. A series of simulations of the 2014 M_w 5.1 La Habra earthquake guided and accompanied our developments. The High-Frequency (High-F) ground motion verification project of the Southern California Earthquake Center built the initial umbrella of the conducted runs. High-F specifies inputs for the solvers which are used by the participating modelers:

- the modeling assumptions, i.e, anelastic attenuation with a frequency-independent Q-definition;
- the targeted frequency content of the simulations, i.e, requiring results which are accurate up to 5 Hz;
- the temporal and spatial extent of the simulations, a map showing High-F's "small" and "large" domains is given in Fig. 1;
- the used seismic velocity model (CVM-S4.26.M01) with a set of parameter constraints;
- the kinematic description of the assumed earthquake rupture; and
- the set of seismic stations for which the synthetic seismograms are compared.

The goal of the High-F verification effort is a high agreement of the synthetic seismograms when using diverse solvers but the same input. The project's challenges are driven by the high complexity of the targeted simulations and the high computational demands of the individual forward simulations. An exemplary result of this work is given in Fig. 2. We observe an excellent agreement of EDGE and the finite-difference

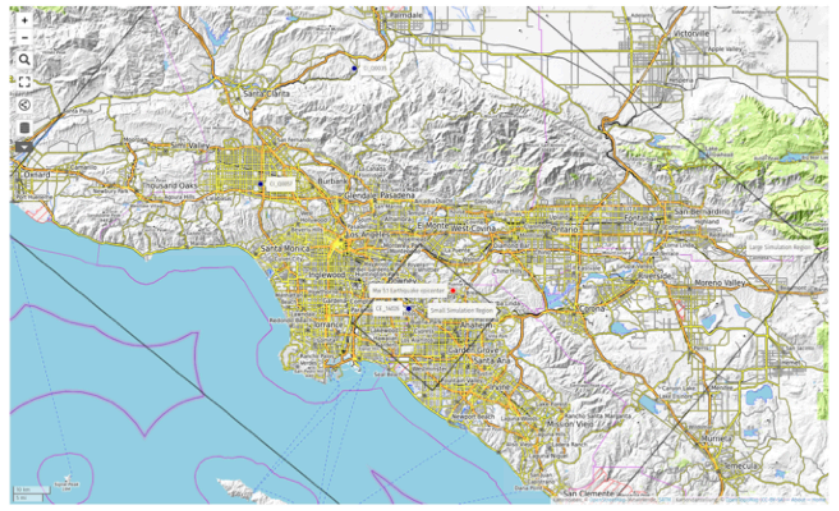


Fig. 1. Study area of the Southern California Earthquake Center's High Frequency project, simulating the 2014 M_w 5.1 La Habra, California earthquake. Shown are the "small" domain through the inner box and partially the "large" domain through the outer box. Additionally, the locations of the earthquake's epicenter and three stations are given. The screenshot was obtained from http://u.osmfr.org/m/560152/.

The authors acknowledge the Large-Scale Community Partnership "SCEC Earthquake Modeling, Ground Motion, and Hazard Simulations" at the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources that have contributed to the research results reported within this manuscript. This work was supported through the project "Hocheffiziente und Flexible Deep Learning-Bausteine für Arm- und Power-Prozessoren" funded by the Carl Zeiss Foundation.

Acknowledgements

- Large-Scale Community Partnership "SCEC Earthquake Modeling, Ground Motion, and Hazard Simulations" at the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources that have contributed to the research results reported within this work

- Project "Hocheffiziente und Flexible Deep Learning-Bausteine für Arm- und Power-Prozessoren" funded by the Carl Zeiss Foundation

Searching for a Ph.D. or postdoc position?

Get in touch:

alex.breuer@uni-jena.de
https://scalable.uni-jena.de