GPU-enabled ensemble data assimilation for mesh-refined lattice Boltzmann method

Yuta Hasegawa[1,*], *Yasuhiro* Idomura[1], and *Naoyuki* Onodera[1]

[1]Japan Atomic Energy Agency Center for Computational Science and e-Systems, Japan.

Abstract. We implemented the ensemble data assimilation (DA) method, the local ensemble transform Kalman filter (LETKF), into the mesh-refined lattice Boltzmann method (LBM) for turbulent flows. Both the LETKF and the mesh-refined LBM were fully implemented on GPUs, so that they are efficiently computed on modern GPU-based supercomputers. We examined the DA accuracy against the flow around a cylinder. The result showed that our method enabled accurate DA with spatially- and temporarily-sparse observation data; the error of the assimilated velocity field with the observation interval of $\tau_K/2$ and the observation resolution $D/16$ (1.56% of the total computational grids) was smaller than the amplitude of the observation noise, where τ_K is the period of the Kármán vortex and D is diameter of the square cylinder.

1 Introduction

Data assimilation (DA), which merges the two types of data from simulations and measurements, is a key technology to construct digital twins in computational nuclear engineering. One of the challenges in realizing such digital twins is to utilize high-performance computing (HPC). For Example, in the nuclear CFD field, Merzari et al. [1] utilized almost the entire resource of the world-leading supercomputer Frontier to simulate the full-scale small module reactor. In the DA field, Yashiro et al. [2] allocated almost the full scale of the Japanese national flagship supercomputer Fugaku to conduct the DA calculation of the global weather prediction. In this work, we develop high-performance DA and CFD code, especially using the lattice Boltzmann method (LBM) [3] with mesh refinement. Here, the LBM and the mesh refinement method are essential for fast and scalable simulations of turbulent flows. For example, these methods enabled real-time urban wind simulations at a-meter-scale mesh resolution [4, 5]. As the DA model, we chose the local ensemble transform Kalman filter [6] (LETKF), which is suitable for large-scale parallel computation (e.g., [2, 7]). By combining these CFD and DA approaches, we develop the LBM-LETKF on GPU-based supercomputers. As the first step of validation, we address the DA numerical experiments in a simple model, a flow around a 3D square cylinder, and examine the DA accuracy, in particular, at sparse observation conditions both in space and time.

*e-mail: hasegawa.yuta@jaea.go.jp

2 Methods

2.1 Lattice Boltzmann method (LBM)

As the CFD model, we utilize our in-house code `CityLBM`. The code is based on the lattice Boltzmann method (LBM) with the cumulant collision model and the mesh refinement. `CityLBM` is fully implemented on GPUs by C++/CUDA, and is capable of fast, scalable, accurate simulations for single-phase flow problems. By using `CityLBM`, we achieved the real-time simulation of the wind condition and the plume dispersion in 4km × 4km urban areas, where the mesh resolution was 2m or 4m [5].

The LBM treats fluid as the composition of the pseudo-particles called the distribution function $f_{ijkxyzt}$. The time evolution of the distribution function is expressed as

$$f_{ijk(x+i\Delta x)(y+j\Delta x)(z+k\Delta x)(t+\Delta t)} = f_{ijkxyzt} + \Omega_{ijkxyzt}, \tag{1}$$

where (i, j, k) are the indices of the distribution function that are $(i, j, k) \in \{-1, 0, 1\} \otimes \{-1, 0, 1\} \otimes \{-1, 0, 1\}$, (x, y, z) are the Cartesian coordinates, Δt is the time interval, Δx is the grid spacing, and $\Omega_{ijkxyzt}$ is the collision operator. The detail of the collision operator is derived in the literature [8]. The fluid properties, density ρ and velocity $\boldsymbol{u}$, are defined as the moments of the distribution functions, expressed as

$$\rho_{xyzt} = \sum_{i,j,k} f_{ijkxyzt}, \tag{2}$$

$$\boldsymbol{u}_{xyzt} = \frac{1}{\rho_{xyzt}} \sum_{i,j,k} \boldsymbol{c}_{ijk} f_{ijkxyzt}, \tag{3}$$

where $\boldsymbol{c}_{ijk} = (ci, cj, ck)^\top$, with $c = \Delta x/\Delta t$ being the model constant called the lattice speed.

2.2 Local ensemble transform Kalman filter (LETKF)

As the DA model, we utilize the local ensemble transform Kalman filter (LETKF) [6]. The LETKF is one of the ensemble Kalman filters, which are major DA methods in meteolorogy, is suitable for efficient parallelization in both CPU- and GPU-based supercomputers (e.g., [2, 9, 10]).

2.2.1 General description of LETKF

The LETKF utilizes ensemble simulation, which solves the time evolution of nonlinear systems from different initial conditions. Let M be the ensemble size, and let $m = 0, 1, \ldots, M-1$ be the index of each ensemble member. Let $\boldsymbol{x}_m^f$ be the state variables vector of the m-th simulation (here, superscript f means *forecast*). Then, the ensemble statistics, the mean vector $\bar{\boldsymbol{x}}^f$ and the deviation matrix $\delta\boldsymbol{X}^f$, are introduced as

$$\bar{\boldsymbol{x}}^f = \frac{1}{M} \sum_{m=0}^{M-1} \boldsymbol{x}_m^f, \tag{4}$$

$$\delta\boldsymbol{X}^f = \left[\boldsymbol{x}_0^f - \bar{\boldsymbol{x}}^f, \boldsymbol{x}_1^f - \bar{\boldsymbol{x}}^f, ..., \boldsymbol{x}_{M-1}^f - \bar{\boldsymbol{x}}^f\right]. \tag{5}$$

Next, the measurement data (called *observation*) are introduced. Let $\boldsymbol{y}^o$ be the observation vector. Here, to consider the error (or noise) of the observation, the covariance matrix $\boldsymbol{R}$ is also introduced. $\boldsymbol{R}$ is theoretically defined as

$$\boldsymbol{R} = \left(\boldsymbol{y}^o - \boldsymbol{y}^t\right)\left(\boldsymbol{y}^o - \boldsymbol{y}^t\right)^\top. \tag{6}$$

where $\boldsymbol{y}^t$ is the observation vector in the ground truth state. In practice, however, the ground truth state is unknown, then, the matrix $\boldsymbol{R}$ is given as an empirical model constant. In this study, we set $\boldsymbol{R} = \mathrm{diag}\left(\sigma^2, \sigma^2, \ldots, \sigma^2\right)$, by assuming that each element of the observation does not correlate with each other, each element has no biased error, and the observation error follows the Gaussian noise with the standard deviation σ.

The forecast variable converted in the observation space, $\boldsymbol{y}^f_m = \mathcal{H}\boldsymbol{x}^f_m$, is also introduced, where $\mathcal{H}$ is the observation operator. The ensemble mean and deviation of $\boldsymbol{y}^f$ is given as

$$\bar{\boldsymbol{y}}^f = \frac{1}{M}\sum_{m=0}^{M-1} \boldsymbol{y}^f_m, \tag{7}$$

$$\delta\boldsymbol{Y}^f = \left[\boldsymbol{y}^f_0 - \bar{\boldsymbol{y}}^f, \boldsymbol{y}^f_1 - \bar{\boldsymbol{y}}^f, ..., \boldsymbol{y}^f_{M-1} - \bar{\boldsymbol{y}}^f\right]. \tag{8}$$

Finally, the LETKF computes the data-assimilated state (called *analysis*) by using above variables, $\bar{\boldsymbol{x}}^f, \delta\boldsymbol{X}^f, \boldsymbol{y}^o, \boldsymbol{R}, \bar{\boldsymbol{y}}^f, \delta\boldsymbol{Y}^f$. The analysis state vector of the m-th ensemble, $\boldsymbol{x}^a_m$, is computed as follows. In the LETKF, the analysis is first solved in the ensemble space, $\boldsymbol{w}^a \in \mathbb{R}^M$, where $\boldsymbol{x}^a = \bar{\boldsymbol{x}}^f + \delta\boldsymbol{X}^f\boldsymbol{w}^a$. The ensemble mean $\bar{\boldsymbol{w}}^a$ and deviation δW^a are computed by

$$\tilde{\boldsymbol{P}}^a = \left[\frac{M-1}{\beta}\boldsymbol{I} + \delta\boldsymbol{Y}^{f\top}\boldsymbol{R}^{-1}\delta\boldsymbol{Y}^f\right]^{-1}, \tag{9}$$

$$\bar{\boldsymbol{w}}^a = \tilde{\boldsymbol{P}}^a\delta\boldsymbol{Y}^{f\top}\boldsymbol{R}^{-1}\left(\boldsymbol{y}^o - \bar{\boldsymbol{y}}^f\right), \tag{10}$$

$$\delta\boldsymbol{W}^a = \left[(M-1)\tilde{\boldsymbol{P}}^a\right]^{1/2}, \tag{11}$$

where β is the covariance inflation factor [6]. We choose $\beta = 1.05$ in this study. The analysis is converted back into the state vector on each ensemble member, $\boldsymbol{x}^a_m$, as

$$\boldsymbol{x}^a_m = \bar{\boldsymbol{x}}^f + \delta\boldsymbol{X}^f\left(\bar{\boldsymbol{w}}^a + \delta\boldsymbol{w}^a_m\right), \tag{12}$$

where $\delta\boldsymbol{w}^a_m$ is the m-th column of $\delta\boldsymbol{W}^a$.

The detailed definition of the vectors $\boldsymbol{x}, \boldsymbol{y}$ depends on specific problems. We will define these vectors for the LBM in the next section.

2.2.2 Choice of vectors in LBM

To define state variable and observation vectors, $\boldsymbol{x}, \boldsymbol{y}$, we follow the so-called local volume solver approach [11] (see Figure 1 for example). In this approach, the state vector $\boldsymbol{x}$ is solved on each grid point independently (Figure 1, red cell), while each grid point refers to some neighborhood observation points within a certain distance (Figure 1, blue cells within red circle). In this study, the observation points are uniformly sampled at each mesh refinement level. The maximum distance between the grid point and the neighborhood observation points d_{max} is adjusted so that every grid point in the finest mesh refers to at least one neighborhood observation point. Here, d_{max} is set to constant over the different mesh resolutions.

As the state variables vector $\boldsymbol{x}$, we use the macroscopic variables of fluid, namely,

$$\boldsymbol{x} = (\rho, u, v, w)^\top \in \mathbb{R}^4. \tag{13}$$

Although the independent variable of the LBM is the distribution function f_{ijk}, the above choice of vector has a merit that the size of the vector $\boldsymbol{x}$ is reduced, leading to the cost reduction of MPI communication (Note that the ensemble DA needs `MPI_Alltoall` communication of $\boldsymbol{x}^f_m$ and $\boldsymbol{x}^a_m$ once per DA cycle [10]). Instead of the distribution function, we

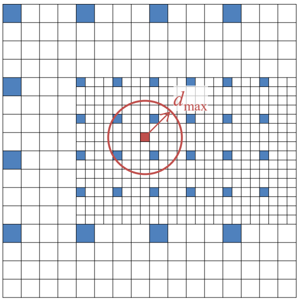


Figure 1: Relation between a DA grid point (Red cell) and observation points (Blue cells) on locally refined grids. Here, neighborhood observation points within local volume (Red circle) are used in the DA.

solve the analysis of the macroscopic variables; after that, we convert them into the distribution function. To conserve the shear stress tensor of fluid through the DA, the distribution function is split into the equilibrium and non-equilibrium parts, and only the equilibrium part is assimilated. This approach is written as

$$f^{a}_{ijk} = f^{eq,a}_{ijk} + f^{neq,f}_{ijk}, \tag{14}$$

where $f^{eq,a}_{ijk}$ and $f^{neq,f}_{ijk}$ are respectively the assimilated equilibrium and non-assimilated non-equilibrium parts of f_{ijk}. The equilibrium part f^{eq}_{ijk} is given by the lattice BGK form [12],

$$f^{eq}_{ijk} = w_\alpha \rho \left(1 + \frac{\boldsymbol{u}\cdot\boldsymbol{c}_{ijk}}{c_s^2} + \frac{(\boldsymbol{u}\cdot\boldsymbol{c}_{ijk})^2 - c_s^2|\boldsymbol{u}|^2}{2c_s^4}\right), \tag{15}$$

$$c_s = \frac{c}{\sqrt{3}}, \tag{16}$$

$$w_\alpha = \begin{cases} 8/27, & |\boldsymbol{c}_{ijk}| = 0 \\ 2/27, & |\boldsymbol{c}_{ijk}| = 1 \\ 1/54, & |\boldsymbol{c}_{ijk}| = \sqrt{2} \\ 1/216, & |\boldsymbol{c}_{ijk}| = \sqrt{3} \end{cases}, \tag{17}$$

and the non-equilibrium part is defined as

$$f^{neq}_{ijk} = f_{ijk} - f^{eq}_{ijk}. \tag{18}$$

As the observation, we assume the fluid density ρ^o and velocity $\boldsymbol{u}^o = (u^o, v^o, w^o)$ are observable. Then, the observation vector $\boldsymbol{y}^o$ is given as

$$\boldsymbol{y}^o = \left(\rho^o_0, u^o_0, v^o_0, w^o_0, \rho^o_1, u^o_1, v^o_1, w^o_1, \ldots, \rho^o_{L-1}, u^o_{L-1}, v^o_{L-1}, w^o_{L-1}\right)^\top, \tag{19}$$

where L is the number of neighborhood observation points within the local volume.

3 Results

To examine the accuracy of the LETKF, we carried out the DA experiment in the flow around a 3D square cylinder. The numerical configuration is shown in Figure 2. The domain size was

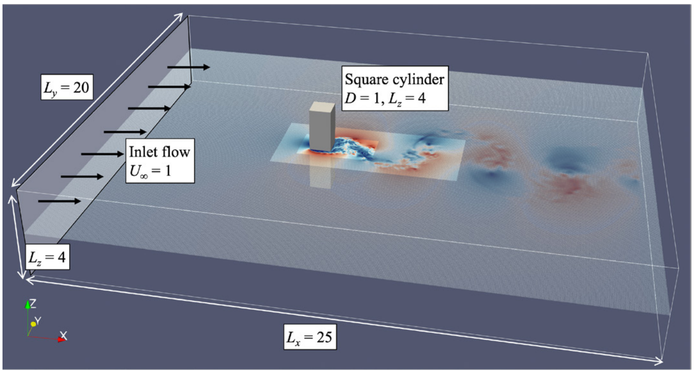


Figure 2: Setup of the flow around a 3D square cylinder.

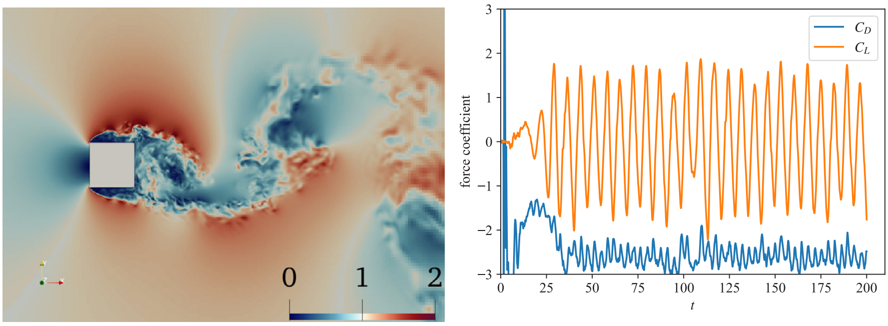


Figure 3: Ground truth state. (Left) snapshot of velocity magnitude profile on a horizontal plane. (Right) time histories of drag and lift coefficients.

$L_x \times L_y \times L_z = 25 \times 20 \times 4$, and the 3D square cylinder with diameter $D = 1$ was placed at the center of the domain. As the mesh refinement, the finer mesh was assigned near the cylinder; the mesh resolutions were $\Delta x = 1/64$ in the region near the cylinder, and $\Delta x = 1/32, 1/16$ in the surrounding area. The number of total grid points was then 23 million. The boundary conditions were set to the inflow-outflow boundary for the streamwise direction (x), and the periodic boundary for the spanwise and vertical directions (y, z). The inlet flow was the uniform flow with the velocity $U_\infty = 1$. The lattice speed was set to $c = 10U_\infty$, corresponding to the time interval $\Delta t = 1/160$. The Reynolds number, with reference velocity and length being U_∞ and D, respectively, was $Re = 22,000$.

First, we show the calculation result in the ground truth state in Figure 3. Here, drag and lift coefficients were defined as

$$C_D = \frac{F_x}{\rho A U_\infty^2/2}, \tag{20}$$

$$C_L = \frac{F_y}{\rho A U_\infty^2/2}, \tag{21}$$

respectively, where $A = L_y L_z$ is the frontal area, (F_x, F_y, F_z) is the fluid force to the cylinder. The fluid force was computed by the momentum exchange method during the bounce-back

Table 1: Configuration of DA experiment

parameter	value
Ensemble size M	32
DA time interval τ_{DA}	$600\Delta t$
observation noise amplitude on each velocity component (u, v, w)	$0.1 \times U_\infty$
observation noise amplitude on density (ρ)	$0.01 \times \rho_0$
spacial interval of observation points $p\Delta x$	$\{1, 2, 4, \ldots, 64\}\Delta x$

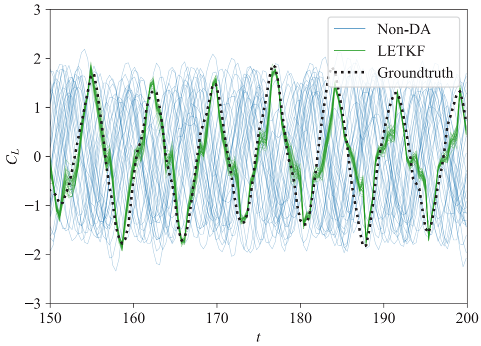


Figure 4: Comparison of time histories of lift coefficients. "Groundtruth" is the same as Figure 3. "Non-DA" shows ensemble simulations without DA. "LETKF" shows ensemble simulations using LETKF with dense observation points $p = 1$.

boundary condition [13]. As shown in the figure, the flow was composed of the semi-periodic flow of the Kármán vortex and the small-scale turbulence. The periodic flow component dominated the properties of the drag and lift coefficient, and its period was $\tau_K = 7.5 = 1200\Delta t$, corresponding to the Strouhal number $St = 0.133$. We also estimated another time scale of the turbulence, the Lyapunov exponent λ [14]. The estimated Lyapunov exponent of this simulation was $\lambda = 2.30$, then, the time scale of the turbulence was estimated as $\tau_T = 1/\lambda \simeq 60\Delta t$, which was an order of magnitude shorter than the Kármán vortex period τ_K.

For the DA experiment, we configured the parameter for the observation as shown in Table 1. The time interval was set to $\tau_{DA} = 600\Delta t$, which was half of the Kármán vortex period $(\tau_K/2)$, or ten times coarser than the turbulence time scale $(10\tau_T)$. Such time interval is recognized as temporarily sparse in terms of the chaotic behavior of turbulence. We also examined the sparsity in space; the spatial interval of observation points is tested from $p = 1$ (dense case), 2, 4, . . . , 64 (sparse cases).

Figure 4 shows the result of the time history of the lift coefficient at the dense observation case $p = 1$. When the DA was not applied (Non-DA case), the phase of the oscillation in the ensemble simulation was randomly distributed, although the amplitude of the lift was consistent. In the LETKF case, the phase of the oscillation was well maintained. However, large jumps were observed after the DA, because the interval of the observation was relatively sparse, and thus the corrections to the macroscopic variables due to the LETKF became large.

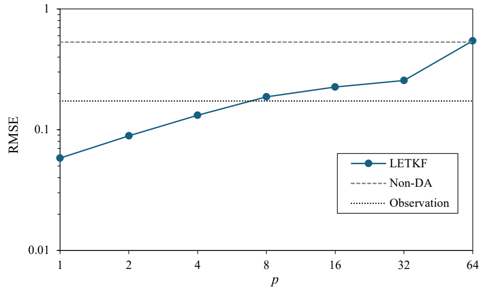
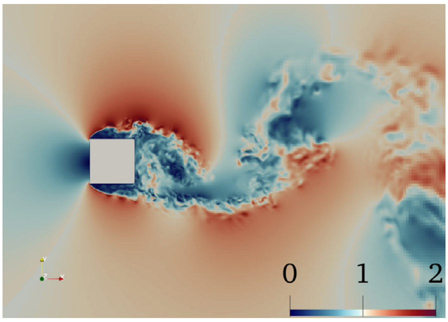
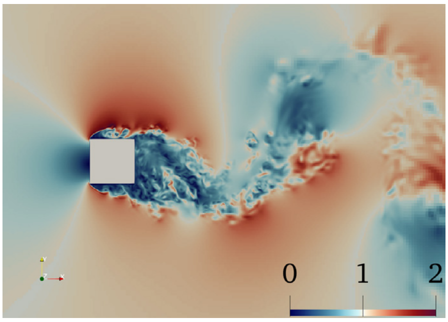


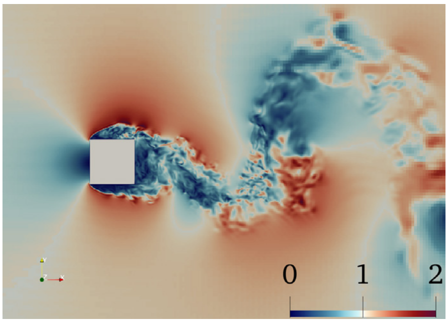
Figure 5: Root mean squared error (RMSE) of the velocity field in sparse observation cases. "Observation" shows the noise amplitude on the velocity, i.e., $0.1 \times \sqrt{3}$.



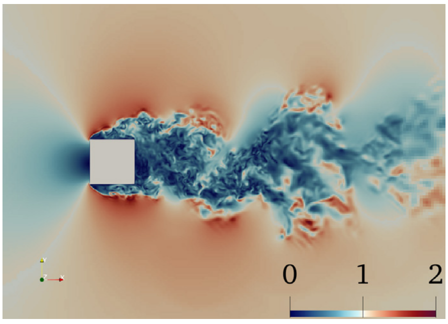
(a) ground truth



(b) LETKF, $p = 1$



(c) LETKF, $p = 4$



(d) LETKF, $p = 64$

Figure 6: Comparison of the snapshots of velocity magnitude on a horizontal slice.

In Figure 5, the root mean squared error (RMSE) of the velocity field is plotted against the spacial interval p. Here, the RMSE is computed as

$$RMSE = \sqrt{\frac{1}{N}\sum_{x,y,z}(\bar{u}^a_{xyz} - u^t_{xyz})^2 + (\bar{v}^a_{xyz} - v^t_{xyz})^2 - (\bar{w}^a_{xyz} - w^a_{xyz})^2} \tag{22}$$

where N is the total number of the computational grid points, the superscripts a and t denote the analysis (data-assimilated) and ground truth states, respectively. The RMSE was smaller than the observation noise in $p = 1, 2, 4$. Here, the spatial interval of $p = 4$ indicated that the minimum required number of observation points was $1/4^3 = 1.56\%$ of the total computational grids.

Figure 6 shows the comparison of the velocity field on a horizontal plane. The figure indicated that the phases of the Kármán vortexes of the ground truth and the LETKF were well synchronized at $p = 1, 4$, while the flow pattern is significantly different at $p = 64$. This result also showed that the result with the minimum required spatial interval $p = 4$ had reasonable DA accuracy.

The above results demonstrate that the LETKF enables robust and accurate DA with sparse observation both in space and time.

4 Conclusion

In this paper, we implemented the local ensemble transform Kalman filter (LETKF) into the mesh-refined lattice Boltzmann method (LBM) code `CityLBM`. We examined the DA accuracy against the flow around a square cylinder, and showed that our method well assimilated the periodic part of the flow due to the Kármán vortex, although the observation was sparse both in space and time. The interval of the observation was enough when $\tau_{DA} = 600\Delta t$, which was equal to half of the period of the Kármán vortex $\tau_T = 1200\Delta t$, or which was ten times coarser than the turbulent time scale $\tau_T \simeq 60\Delta t$. The required spatial resolution of the observation was at least $4\Delta x$, which was only 1.56% number of the computational grids. These results encouraged us to apply our DA method, LBM-LETKF, to more complex problems such as the wind tunnel experiments with multiple buildings and scalar transport [15], or wind condition and plume dispersion in real urban city [16]. We will study these problems in the future works.

Acknowledgment

This work was funded by the JSPS KAKENHI [Grant Numbers JP21K17755, JP22H03599], and the Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures (JHPCN) [Project ID jh230014]. Computations were carried out on HPE SGI8600 at Japan Atomic Energy Agency.

References

[1] E. Merzari, S. Hamilton, T. Evans, M. Min, P. Fischer, S. Kerkemeier, J. Fang, P. Romano, Y.H. Lan, M. Phillips et al., *Exascale Multiphysics Nuclear Reactor Simulations for Advanced Designs*, in *SC '23: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (2023), 3

[2] H. Yashiro, K. Terasaki, Y. Kawai, S. Kudo, T. Miryoshi, T. Imamura, K. Minami, H. Inoue, N. Tatsuo, T. Saji et al., *A 1024-Member Ensemble Data Assimilation with 3. 5-Km Mesh Global Weather Simulations*, in *SC '20: International Conference for High Performance Computing, Networking, Storage and Analysis* (2020), 1

[3] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, E.M. Viggen, *The Lattice Boltzmann Method: Principles and Practice* (2017)

[4] N. Onodera, Y. Idomura, Y. Ali, T. Shimokawabe, *Communication Reduced Multi-time-step Algorithm for Real-time Wind Simulation on GPU-based Supercomputers*, in *ScalA 2018: Proceedings of 2018 IEEE/ACM 9th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)* (2018), pp. 9–16

[5] N. Onodera, Y. Idomura, Y. Hasegawa, H. Nakayama, T. Shimokawabe, T. Aoki, Boundary-Layer Meteorology **179**, 187 (2021)

[6] B.R. Hunt, E.J. Kostelich, I. Szunyogh, Physica D: Nonlinear Phenomena **230**, 112 (2007)

[7] T. Miyoshi, K. Kondo, T. Imamura, Geophysical Research Letters **41**, 5264 (2014)

[8] M. Geier, M. Schönherr, A. Pasquali, M. Krafczyk, Computers & Mathematics with Applications **70**, 507 (2015)

[9] L. Adhianto, S. Banerjee, M. Fagan, M. Krentel, G. Marin, J. Mellor-Crummey, N.R. Tallent, Concurrency and Computation: Practice and Experience **24**, 167 (2012)

[10] Y. Hasegawa, T. Imamura, T. Ina, N. Onodera, Y. Asahi, Y. Idomura, *GPU Optimization of Lattice Boltzmann Method with Local Ensemble Transform Kalman Filter*, in *2022 IEEE/ACM Workshop on Latest Advances in Scalable Algorithms for Large-Scale Heterogeneous Systems (ScalAH)* (2022), pp. 10–17

[11] S. Frolov, A. Shlyaeva, W. Huang, T. Sluka, C. Draper, C. Martin, T. Elless, K. Bhargava, J. Whitaker, Journal of Advances in Modeling Earth Systems **16**, e2023MS003692 (2024)

[12] Y.H. Qian, D. D'Humières, P. Lallemand, Europhysics Letters **17**, 479 (1992)

[13] D. Yu, R. Mei, L.S.S. Luo, W. Shyy, Progress in Aerospace Sciences **39**, 329 (2003)

[14] G. Nastac, J.W. Labahn, L. Magri, M. Ihme, Physical Review Fluids **2**, 094606 (2017)

[15] R. Yoshie, A. Mochida, Y. Tominaga, H. Kataoka, K. Harimoto, T. Nozu, T. Shirasawa, Journal of Wind Engineering and Industrial Aerodynamics **95**, 1551 (2007)

[16] M.J. Leach, *Final Report for the Joint Urban 2003 Atmospheric Dispersion Study in Oklahoma City : Lawrence Livermore National Laboratory participation* (2005)