A study on efficiency of semi-implicit, density-based solver for simulation of evaporating particle-laden flow

Younghwa CHO*, Rahul BALE**, Makoto TSUBOKURA*** and Nobuyuki OSHIMA****

*Graduate School of Engineering, Hokkaido University
Kita 13, Nishi 8, Sapporo, Hokkaido 060-8628, Japan
E-mail: younghwa@eis.hokudai.ac.jp
**RIKEN Center for Computational Science (R-CCS)
7-1-26 Minatojima-minami-machi, Chuo-ku, Kobe, Hyogo 650-0047, Japan
***Graduate School of System Informatics, Kobe University
1-1 Rokkodai, Nada-ku, Kobe, Hyogo 657-8501, Japan
****Faculty of Engineering, Hokkaido University
Kita 13, Nishi 8, Sapporo, Hokkaido 060-8628, Japan

Received: 12 April 2021; Revised: 18 May 2021; Accepted: 20 August 2021

Abstract
In this work, we propose a semi-implicit, density-based solver for compressible, evaporating particle-laden flow, and investigate its efficiency. It is established on a Cartesian-grid-based, scalable, numerical framework named CUBE. In this solver, the governing equation system is divided into three subsystems (compressible Navier–Stokes, species transport, and Lagrangian), and these subsystems are weakly coupled in two ways. In the Lagrangian domain, the fuel spray is treated as a set of discrete particles, and the particle–source-in-cell (PSI-Cell) method is employed for the coupling between the Eulerian and Lagrangian domains. Furthermore, the species transport and Lagrangian subsystems are subcycled with smaller time step, and the Navier–Stokes equation is temporally integrated with a larger step size. The proposed solver's verification and evaluation is conducted on the supercomputer Fugaku by comparing the results with those of the original, fully explicit solver where all equations have the same time step. The results show that this solver reduces the computational cost while ensuring similar accuracy. The solution of the proposed solver is consistent with that of the original solver. Finally, we brief our perspective on the future application of the proposed solver to our target problem: the large-scale simulation of evaporating particle-laden flow in a combustor of an aviation engine.

Keywords: Semi-implicit solver, Density-based algorithm, Subcycling, Euler–Lagrangian coupling, Droplet evaporation, Large-eddy simulation, Supercomputer Fugaku

1. Introduction

In an aviation engine, highly compressed air flows into the combustor and accelerates through combustion up to subsonic conditions. Thus, there is a strong coupling between thermodynamics and dynamics (i.e., compressibility) in the combustor flow, which affects the fuel evaporation. Fuel evaporation also contributes to the distribution of the air-to-fuel ratio inside the combustor, which governs the efficiency, emissions, and safety of the engine. Thus, it is important to perform numerical analysis to determine the compressibility of the flow, and pressure- or density-based algorithms are the two means for doing so. A representative example that adopts the former is KIVA-II, a pressure-based solver for reacting spray flow (Amsden et al., 1989). However, pressure-based algorithms solve mass, momentum, and energy separately and so are disadvantageous for describing such compressibility when compared to density-based algorithms, where the mass, momentum, and energy are solved simultaneously. Therefore, the fully compressible Navier–Stokes equation using a density-based algorithm is considered the most appropriate for such an analysis.

Also, because the combustor flow is described as a multiphysical phenomenon, there are disparities in the timescales
of flow, droplet dynamic response, droplet evaporation, etc. Therefore, to employ a solver where the time step is unified for all physical phenomena, one must choose the minimum timescale to obtain an accurate numerical solution. This approach is inefficient and can limit real-time simulation, an ultimate goal of computational mechanics.

Vali et al. (2008) investigated subcycling time stepping strategies for subsystems of multiphysical systems and proposed some basic algorithms. In addition, they outlined the possible combination of time marching one subsystem explicitly and another implicitly by employing different time steps. However, they employed a decoupled scheme where a physical, two-way coupled phenomenon is reduced into a one-way coupled one, ignoring weaker interaction. Although Valli et al. (2008) applied subcycling time stepping only to single phase flow, it is also useful when handling particle-laden flow, which is a representative multiphysical phenomenon. Euler–Lagrangian coupling is also often applied to the simulation of particle-laden flow (Sommerfeld et al., 2017 and 2019; Kieckhefen et al., 2020). Subcycling was introduced as the “semi-unsteady approach” by Sommerfeld (1997) and applied to unsteady Reynolds-averaged Navier–Stokes (URANS) simulation of particle-laden flow. Apte et al. (2009) applied such a strategy to the large-eddy simulation (LES) of evaporating particle-laden flow. Both indicated the disparities in timescales of the multiphysics of turbulent, particle-laden flow and applied the subcycling time stepping strategy (or semi-unsteady approach) to the spray equation to determine the particle dispersion within a time step. Furthermore, they considered two-way coupling between the spray and flow through the particle-source-in-cell (PSI-Cell) method. However, their approach to coupling flow and spray was based on a pressure-based algorithm. To the best of our knowledge, the application to density-based algorithms has not previously been reported.

In the present study, we applied subcycling (the semi-unsteady approach) to a density-based algorithm and established a semi-implicit solver, where the subcycling method was applied to the Euler–Lagrangian simulation that incorporated compressibility and evaporation. The solver was established for large-scale simulation of the evaporating particle-laden flow in a turbofan engine combustor, which is our target problem for future research. We verified our semi-implicit solver on the supercomputer Fugaku and demonstrated its efficiency. Firstly, in this paper, the main concept of the proposed solver is briefly introduced. Then, we show the results of the verification and investigation of the computational efficiency. Finally, we briefly discuss our perspective on the future application of the solver to a large-scale simulation of a fuel-spray-laden flow in an aviation engine combustor.

2. Numerical framework

2.1 Governing equations

2.1.1 Eulerian domain

The fully compressible Navier–Stokes equation, accompanied by the species transport equation employed in this work, can be expressed in a compact form as follows:

$$\frac{\partial U}{\partial t} + \nabla \cdot F = S, \quad U = \begin{bmatrix} \rho \\ \rho u_i \\ \rho u_j \\ \rho u_k \\ \rho e \\ \rho Y_k \end{bmatrix}, \quad F_i = \begin{bmatrix} \rho u_i \\ \rho u_i u_j + p \delta_{ij} - \tau_{ij} \\ \rho u_i u_j + p \delta_{ij} - \tau_{ij} \\ \rho u_i u_j + p \delta_{ij} - \tau_{ij} \\ (p e + p) u_i - \tau_{ij} u_j - \lambda \left( \nabla T \right)_i \\ \rho Y_k u_i - \rho D_k \left( \nabla Y_k \right)_i \end{bmatrix}, \quad S = \begin{bmatrix} S_\rho \\ S_{\rho u_i} \\ S_{\rho u_j} \\ S_{\rho u_k} \\ S_{\rho e} \\ S_{\rho Y_k} \end{bmatrix}, \quad i, j \in \{1, 2, 3\},$$

where $\rho, u_i, e, Y_k, p, T, \tau_{ij}, \lambda,$ and $D_k$ are the gaseous density, velocity component along the principal direction $i$, total specific energy, species mass fraction, pressure, temperature, shear-stress tensor, thermal conductivity, and species diffusivity, respectively. $S$ is a source vector, and each component is the source term of the corresponding conservative variable. These source terms represent two-way coupling between the gas and liquid phases. Their specific expressions can be found in Section 2.1.2.
2.1.2 Lagrangian domain

The Lagrangian equation that governs droplet motion is expressed as follows:

\[ \frac{dx_d}{dt} = u_d, \]
\[ \frac{du_d}{dt} = f_1(u - u_d), \]
\[ \frac{dT_d}{dt} = \frac{Nu}{3Pr} \left( \frac{f_2}{\tau_d} \right) (T - T_d) + \frac{1}{m_d} \frac{dm_d}{dt} L_v, \]
\[ \frac{dm_d}{dt} = -\frac{m_d}{\tau_d} \left( \frac{Sh}{3Sc} \right) \ln(1 + B_M), \]

where \( x_d, u_d, T_d, \tau_d, \) and \( m_d \) are the droplet’s position, velocity, temperature, response time, and mass, respectively. \( L_v, c_{p,d}, \) and \( c_p \) are the latent heat of evaporation, droplet’s specific heat, and gaseous specific heat, respectively. \( Nu, Pr, Sh, \) and \( Sc \) are the Nusselt, Prandtl, Sherwood, and Schmidt numbers, respectively. Also, \( B_M \) is the mass transfer number. Finally, \( f_1 \) and \( f_2 \) are the corrections of the Stokes drag and heat transfer for an evaporating droplet, respectively. Here, \( \tau_d, Nu, B_M, f_1, \) and \( f_2 \) are given in the works of Nakamura et al. (2005) and Kitano et al. (2014).

The source terms introduced in Section 2.1.1 are given as follows:

\[ S_\rho = -\frac{1}{\Delta V} \sum_{N} \frac{dm_d}{dt}, \]
\[ S_{\rho u_i} = -\frac{1}{\Delta V} \sum_{N} \frac{d(m_d u_{d,i})}{dt}, \]
\[ S_{\rho e} = -\frac{1}{\Delta V} \sum_{N} \frac{d(m_d h_{d})}{dt}, \]
\[ S_{\rho Y_k} = -\frac{1}{\Delta V} \sum_{N} \frac{dm_d}{dt}, \text{ only for } Y_k = Y_{\text{fuel}}, \]

where \( \Delta V, N, \) and \( h_d \) are the volume of a computational grid that includes the droplets, the number of droplets in the grid, and the specific enthalpy of each droplet, respectively.

2.2 Main concept of the proposed algorithm

We adopted an algorithm, validated by Bale et al. (2020), as the original algorithm, and it has two subsolvers: one Eulerian and one Lagrangian. The two are temporally explicit and have the same time step, \( \Delta t_{\text{unified}} \), which must be chosen based on the minimum timescale, including the computational stability. In addition, in the Eulerian domain of the original solver, the Navier–Stokes equation is strongly coupled with the species transport equation.

In this study, we divided the equation system of the Eulerian domain into two subsystems: the compressible Navier–Stokes and species transport equations. By assuming moderate and low Mach number flow, we could employ implicit time marching for the Navier–Stokes, allowing reasonably larger time steps and better stability. Also, being separated from the system of the compressible Navier–Stokes equation, the subsystem of species transport equation is able to avoid the severe time step constraint due to acoustic CFL number \( \text{CFL}_{\text{acoustic}} \), and a larger time step can be allowed for the computational stability governed by flow CFL number \( \text{CFL}_{\text{flow}} \), (i.e., \( \Delta t_{\text{sub}} > \Delta t_{\text{unified}} \)). Here, the \( \text{CFL}_{\text{acoustic}}, \) \( \text{CFL}_{\text{flow}}, \) and time step \( \Delta t_{\text{unified}} \) and \( \Delta t_{\text{sub}} \) which meet the CFL condition are described as follows:
\[
CFL_{\text{flow}} = \Delta t \sum_i \frac{|u_i|}{\Delta x_i}, \quad CFL_{\text{acoustic}} = \Delta t \sum_i \frac{\max(|u_i| \pm a_i)}{\Delta x_i} \quad (i: \text{index of spatial dimension, e.g., } x, y, \text{and } z).
\]

\[
\Delta t_{\text{unified}} < \min \left( \sum_i \frac{|u_i|}{\Delta x_i} \right)^{-1}, \quad \left( \sum_i \frac{\max(|u_i| \pm a_i)}{\Delta x_i} \right)^{-1} = \left( \sum_i \frac{a_i}{\Delta x_i} \right)^{-1} \quad (\text{if } Ma < 1) \approx \left( \sum_i \frac{a_i}{\Delta x_i} \right)^{-1},
\]

\[
\Delta t_{\text{sub}} < \left( \sum_i \frac{|u_i|}{\Delta x_i} \right)^{-1} = \left( \sum_i \frac{a_i}{Ma \Delta x_i} \right) \approx Ma^{-1} \left( \sum_i \frac{a_i}{\Delta x_i} \right)^{-1}.
\]

where \( \Delta x_i, a, \) and \( Ma \) are mesh spacing, acoustic speed, and Mach number, respectively. Consequently, it becomes possible to focus more on the physical timescale of each component phenomenon when deciding on the time step. For this reason, implicit and explicit solvers were employed for the Navier–Stokes and species transport equations, respectively. Therefore, after including the Lagrangian solver, the proposed solver has three subsolvers (Navier–Stokes, species transport, and Lagrangian), and each is weakly coupled in two ways. A schematic illustration of the proposed algorithm and its computational flow is shown in Fig. 1.

**Fig. 1** Comparison of original, fully explicit algorithm (upper-left) and the proposed semi-implicit algorithm (upper-right), and schematic of the computational flow of the proposed algorithm (lower).
Meanwhile, in the evaporating particle-laden flow, the mass fraction of fuel vapor and evaporation of droplets affect each other considerably. Thus, in the proposed solver, the Lagrangian and species transport equations shared the same time step ($\Delta t_{\text{sub}}$), and they were explicitly solved via a subcycling strategy that ensured a sufficient temporal resolution of their physics in a time step of the flow field ($\Delta t_{\text{global}}$). Here, for instance, the subcycling time step ($\Delta t_{\text{sub}}$) can be decided as the minimum value between the time step allowed by the maximum flow CFL number, the time scale of droplets’ motion based on the minimum response time ($\tau_d$), the evaporation time scale based on the maximum value of droplets’ evaporation rate ($dm_d/dt$) as given in Eq. (5), and the maximum value of droplets’ traveling distance per each time step. Hence, the means to automatically decide $\Delta t_{\text{sub}}$ can be described as follows:

$$\Delta t_{\text{sub}} = \min \left( \varepsilon_1 \min \text{ in domain} \left( \frac{|u_i|}{\Delta x_i} \right)^{-1}, \ v_2 \min \text{ in domain} (\tau_d), \ v_3 \min \text{ in domain} \left( \frac{dm_d}{dt} \right)^{-1}, \ v_4 \min \text{ in domain} \left( \frac{\Delta x_i}{u_{d,i}} \right) \right).$$  \hspace{1cm} (13)

where the first, second, third, and fourth terms in the right-hand side correspond to criteria given by the flow CFL number, particle response time ($\tau_d$), evaporation time scale ($dm_d/dt$), and time required to travel a mesh spacing ($\Delta x_i$), respectively. $\varepsilon_i$ is a parameter smaller than unity and concerns the degree of temporal resolution of the corresponding physics. Particularly, $\varepsilon_1$ corresponds to the critical flow CFL number (~1), and $\varepsilon_4$ is generally 0.5 which allows particle to travel a half of mesh spacing during the decided time step (the fourth term in the right-hand side of Eq. (13)). Furthermore, if more physics are considered (e.g., combustion), one accordingly needs to include the time scale in Eq. (13), additionally. Finally, the proposed solver was implemented in a scalable numerical framework named CUBE (Jansson et al., 2019), developed by the RIKEN Center for Computational Science, Japan.

3. Verification of the proposed solver

Setting the results of the original, fully explicit solver as the basis, the proposed solver was verified. We compared the spatial distribution of fuel vapor, carrier species, and gaseous velocity. In addition, the temporal profiles of the total mass of the spray droplet and the fuel vapor released were examined. The access to the supercomputer Fugaku was provided by the RIKEN Center for Computational Science, Japan.

3.1 Time integration methods

The third-order strong stability preserving Runge-Kutta method (Gottlieb et al., 2001) was applied to the Eulerian domain of the original solver and species transport of the proposed solver. In addition, the first-order explicit Euler method was employed for the Lagrangian domain of both solvers. Finally, the lower-upper symmetric Gauss–Seidel (Yoon et al., 1988) with dual time stepping (Shuen et al., 1993) method was applied to the compressible Navier–Stokes equation of the proposed solver.

3.2 Computational setup

To verify the proposed solver, spray injection simulation was carried out under a pseudo-engine condition where the ambient temperature, pressure, and velocity were 500 K, 10 atm, and 0 m/s, respectively. The simulation time was 0.05 s, and fuel injection was continued during the time with 1 g/s of injection rate. Droplet size was decided by Rosin–Rammler distribution, and the Sauter mean diameter was 50 µm. The computational domain is cubic with a side length of approximately 0.124 m. A compressible immersed boundary method was applied to the entire wall boundary, and the no-slip condition was imposed (Li et al., 2016). Gravity is not considered in the present study. Putting a length of the domain and the initial velocity of droplet at the nozzle as $L$ and $U$, respectively, the maximum value of a timescale of droplets’ penetration ($L/U$) becomes approximately $1.24 \times 10^{-3}$ s. Also, at the nozzle, response time ($\tau_d$) of droplets with the Sauter mean diameter is around $4.18 \times 10^{-3}$ s. Thus, the ratio of droplets’ maximum penetrating timescale ($L/U$) to the response time ($\tau_d$) is approximately 30%, and this ratio is considered to be enough to observe the dynamic interaction between droplets and flow.
The domain is discretized by Cartesian cubic mesh, the spacing of which is uniformly 2 mm. Here, being compared to a twice refined mesh, this mesh showed an L2-norm of relative error of around 26% and 16% in gaseous velocity and density at the center-axis of the spray nozzle, respectively. Also, both results showed qualitatively good agreement. In this study, on the aspect of verification of the proposed algorithm, we put our focus on the computational efficiency. Thus, we believe the mesh spacing of 2 mm is enough for the present study, ensuring shorter turnaround. The time step ($\Delta t_{\text{unified}}$ in Fig. 1) of the original solver is $1.0 \times 10^{-6}$ s, the maximum size allowed for computational stability including Eq. (11). For the proposed solver, two cases with different global time steps ($\Delta t_{\text{global}}$ in Fig. 1) were investigated, namely $1.0 \times 10^{-4}$ s (Case 1) and $1.0 \times 10^{-3}$ s (Case 2), with the same subcycling time step ($\Delta t_{\text{sub}}$ in Fig. 1; $5.0 \times 10^{-6}$ s). Here, although $\Delta t_{\text{sub}}$ was able to be automatically decided by Eq. (13), we decided $\Delta t_{\text{sub}}$ via trial and error in this study because the corresponding algorithm was not implemented at the time. Using Eq. (13) as the basis, we estimated the order of magnitude of each timescale. As the results, timescales decided by $CFL_{\text{flow}}$, $\tau_d$, $dm_d/dt$, and time required to travel a mesh spacing were around $1 \times 10^{-5}$, $1 \times 10^{-3}$, $1 \times 10^{-6}$, and $1 \times 10^{-5}$ s, respectively. Accordingly, we set $\Delta t_{\text{sub}}$ as $1 \times 10^{-6}$ s and $5 \times 10^{-6}$ s, and both results’ difference was negligible. Thus, $\Delta t_{\text{sub}}$ was fixed to $5 \times 10^{-6}$ s in the present study. A schematic of the computational domain is shown in Fig. 2.

![Fig. 2 The schematic of the computational domain for the present study.](image)

3.3 Results and discussion

Figure 3 shows the distribution of the density of the fuel vapor, density of carrier species, and velocity magnitude. Case 2 captured the distribution of these three parameters, showing good agreement using a time step ($\Delta t_{\text{global}}$) that was 1,000 times larger than that of the original solver ($\Delta t_{\text{unified}}$). In particular, the results of the distribution of velocity magnitude did not show remarkable disparities between the original solver and the solver with Case 2. Figure 4 shows the temporal profiles of the mass of spray remaining and fuel vapor released. As shown in the figure, Case 2 shows an excellent fit with the results of the original solver in both profiles. Thus, Case 2 ensured not only a similar solution, but also a quantitatively consistent solution. Also, at the end of simulation time, $t = 0.05$ s, the mass of spray evaporated is around 68% of that of fuel injected. Thus, it can be confirmed that the simulation time was long enough to observe thermodynamic interaction between droplets and flow, represented by droplet evaporation. As can be seen in the results of Case 1 (Fig. 3), the solution of the proposed solver shows significantly good temporal convergence to the original solver results when the time step is refined. In summary, via the present verification, we were able to verify the accuracy of the proposed solver and temporal convergence to the solution of the original solver. The computational cost was defined as the ratio of each case’s wall-clock time to that of the original solver. Setting the computational cost of the original solver as 100%, the computational costs of Case 1 and 2 were approximately 45% and 25%, respectively.
Fig. 3 Results of original (left column) and proposed solver (middle and right columns) after 0.05 s. The first, second, and third rows show the results of the density of fuel vapor, density of carrier species, and velocity magnitude, respectively.
In the present verification case, the time step allowed by acoustic CFL number was around $1.0 \times 10^{-6}$ s, the same order of magnitude as the timescale of the fuel evaporation. However, when running a high-resolution large-eddy simulation with much smaller mesh spacing, the time step decided by acoustic CFL number may be much smaller than all of the physical timescales. In these cases, the proposed solver may be computationally more efficient because it is possible to avoid time step constraints due to acoustic CFL condition. Thus, we need to examine such cases before applying the proposed solver to our future target: the fuel-spray-laden flow in an aviation engine combustor. Also, as with the present study, it is very time-consuming to decide the subcycling time step ($\Delta t_{\text{sub}}$) in each different case. So, to improve versatility of the proposed solver, we need to implement an algorithm for automatically deciding $\Delta t_{\text{sub}}$ using Eq. (13) in Section 2.2, which will make the implementation much more straightforward.

![Fig. 4 Mass profile of spray remaining in the domain (left) and fuel vapor released in the domain (right). Both results of the proposed, semi-implicit, solver show good consistency with those of the original, fully explicit solver.](image)

4. Conclusion

In this work, we proposed a semi-implicit, density-based solver for compressible, evaporating particle-laden flow. We divided the equation system into three subsystems based on an assumption of moderate and low Mach number flow and employed an implicit time integration with $\Delta t_{\text{global}}$ for a subsystem affected by time step constraints from acoustic CFL number: the subsystem of the compressible Navier–Stokes equation. For the spray and species transport equation subsystems not affected by time step constraint from the acoustic CFL number, we employed an explicit subcycling strategy with $\Delta t_{\text{sub}}$ so that the appropriate temporal resolution was ensured within a time step $\Delta t_{\text{global}}$. In other words, we allowed different time steps for each component phenomenon of the compressible, evaporating particle-laden flow.

We carried out the verification using the results of the original, explicit-in-time solver as a basis, and fuel injection simulation under a pseudo-engine condition was selected as the verification problem. Here, two cases were considered: Case 1 and 2. The two cases had the same time step for the subcycling, $\Delta t_{\text{sub}}$, but different time steps were employed for the implicit time integration of each case, $\Delta t_{\text{global}}$. Although a 1,000 times larger time step was employed in Case 2, the results showed good agreement with those of the original solver. For Case 1, where the time step for the flow field $\Delta t_{\text{global}}$ was refined, the temporal convergence of the solution of the proposed solver to that of the original solver was confirmed. Furthermore, the required computational cost of the proposed solver was reduced being compared to that of the original solver. Consequently, it was reasonable and efficient to divide the entire equation system into three subsystems weakly coupled with each other, allowing different time steps for each component physics.
Acknowledgments

This work was carried out as a part of collaborative research between Hokkaido University, RIKEN Center for Computational Science, and Kawasaki Heavy Industries, Ltd. This work employed computational resources of the supercomputer Fugaku provided by RIKEN Center for Computational Science (R-CCS), Japan, via the High-Performance Computing Infrastructure (HPCI) System Research Project (Project ID: hp200255). This work was also partially supported by the f3 Engineering Education and Research Center, Faculty of Engineering, Hokkaido University, Japan.

References

Amsden, A. A., O'Rourke, P. J. and Butler T. D., KIVA-II: A computer program for chemically reactive flows with sprays, Los Alamos National Lab., No. LA-11560-MS (1989).
Apte, S. V., Mahesh, K. and Moin, P., Large-eddy simulation of evaporating spray in a coaxial combustor, Proceedings of the Combustion Institute, Vol.32, No.2 (2009), pp. 2247–2256.
Bale, R., Wang, W. H., Li, C. G., Onishi, K., Uchida, K., Fujimoto, H., Kurose, R. and Tsubokura, M., A scalable framework for numerical simulation of combustion in internal combustion engines, Proceedings of the Platform for Advanced Scientific Computing Conference (2020), pp. 1–10.
Gottlieb, S., Shu, C. W. and Tadmor, E., Strong stability-preserving high-order time discretization methods, SIAM Review, Vol.43, No.1 (2001), pp. 89–112.
Jansson, N., Bale, R., Onishi, K. and Tsubokura, M., CUBE: A scalable framework for large-scale industrial simulations, The International Journal of High Performance Computing Applications, Vol.33, No.4 (2019), pp. 678–698.
Kieckhefen, P., Pietsch, S., Dosta, M. and Heinrich, S., Possibilities and Limits of Computational Fluid Dynamics–Discrete Element Method Simulations in Process Engineering: A Review of Recent Advancements and Future Trends, Annual review of chemical and biomolecular engineering, Vol.11 (2020), pp. 397–422.
Kitano, T., Nishio, J., Kurose, R. and Komori, S., Effects of ambient pressure, gas temperature and combustion reaction on droplet evaporation, Combustion and Flame, Vol.161, No.2 (2014), pp. 551–564.
Li, C. G., Tsubokura, M. and Bale, R., Framework for simulation of natural convection in practical applications, Vol.75 (2016), pp. 52–58.
Nakamura, M., Akamatsu, F., Kurose, R. and Katsuki, M., Combustion mechanism of liquid fuel spray in a gaseous flame, Physics of Fluids, Vol.17, No.12 (2005), 123301.
Shuen, J. S., Chen, K. H. and Choi, Y., A coupled implicit method for chemical non-equilibrium flows at all speeds, Journal of Computational Physics, Vol.106, No.2 (1993), pp. 306–318.
Sommerfeld, M., Decker, S. and Kohnen, G., Time-dependent calculation of bubble columns based on Reynolds-averaged Navier Stokes equations with ke turbulence model, Proceedings of the 2nd Japanese-German Symposium on Multi-Phase Flow (1997), pp. 323.
Sommerfeld, M., Numerical Methods for Dispersed Multiphase Flows, Particles in Flows (2017), pp. 327–396, Springer International Publishing.
Sommerfeld, M., Cui, Y. and Schmalfuß S., Potential and constraints for the application of CFD combined with Lagrangian particle tracking to dry powder inhalers, European Journal of Pharmaceutical Sciences, Vol.128 (2019), pp. 299–324.
Valli, A. M. P., Carey, G. F. and Coutinho, A. L. G. A., On decoupled time step/subcycling and iteration strategies for multiphysics problems, Communications in numerical methods in engineering, Vol.24, No.12 (2008), pp. 1941–1952.
Yoon, S. and Jameson, A., Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations, AIAA journal, Vol.26, No.9 (1988), pp. 1025–1026.