



Detailed simulation of viral propagation in the built environment

Rainald Löhner¹ · Harbir Antil² · Sergio Idelsohn^{3,4} · Eugenio Oñate⁴

Received: 18 June 2020 / Accepted: 29 June 2020 / Published online: 5 August 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

A summary is given of the mechanical characteristics of virus contaminants and the transmission via droplets and aerosols. The ordinary and partial differential equations describing the physics of these processes with high fidelity are presented, as well as appropriate numerical schemes to solve them. Several examples taken from recent evaluations of the built environment are shown, as well as the optimal placement of sensors.

Keywords Covid-19 · Particle methods · Finite elements · Computational fluid dynamics

1 The Covid-19 crisis

Starting in Wuhan, China, in the fall of 2019, the Covid-19 pandemic has claimed and will continue to claim millions of infected patients and hundreds of thousands of deaths. The lockdowns that followed its outbreak have led to mass unemployment, stalled economic development and loss of productivity that will take years to recover. Some changes in habits and lifestyles may be permanent: in the future, working from home or in a ‘socially distanced manner’ may be the prevalent *modus operandi* for large segments of society.

The present paper gives a short description of computational techniques that can elucidate the flow and propagation of viruses or other contaminants in built environments in order to mitigate [31] or avoid [47] infections.

2 Virus infection

Before addressing the requirements for the numerical simulation of virus propagation a brief description of virus propagation and lifetime are given. Covid-19 is one of many corona-viruses. The virus is usually present in the air or some surface, and makes its way into the body either via inhalation (nose, mouth), ingestion (mouth) or attachment (eyes, hands, clothes). In many cases the victim inadvertently touches an infected surface or viruses are deposited on its hands, and then the hands touch either the nose, the eyes or the mouth, thus allowing the virus to enter the body.

An open question of great importance for all that will follow is how many viruses it takes to overwhelm the body’s natural defense mechanism and trigger an infection. This number, which is sometimes called the *viral load* or the *infectious dose* will depend on numerous factors, among them the state of immune defenses of the individual, the timing of viral entry (all at once, piece by piece), and the amount of hair and mucous in the nasal vessels. In principle, a single organism in a favourable environment may replicate sufficiently to cause disease [86]. Data from research performed on biological warfare agents [33] suggests that both bacteria and viruses can produce disease with as few as 1–100 organisms (e.g. brucellosis 10–100, Q fever 1–10, tularaemia 10–50, smallpox 10–100, viral haemorrhagic fevers 1–10 organisms, tuberculosis 1). Compare these numbers and consider that as many as 3,000 organisms can be produced by talking for 5 min or a single cough, with sneezing producing many more [48,72,78,90,95]. Figure 1, reproduced from [90], shows a typical number and size distribution.

✉ Eugenio Oñate
onate@cimne.upc.edu

¹ Center for Computational Fluid Dynamics, College of Science, George Mason University, Fairfax, VA 22030-4444, USA

² Center for Mathematics and Artificial Intelligence, College of Science, George Mason University, Fairfax, VA 22030-4444, USA

³ ICREA, Catalan Institution for Research and Advanced Studies, Barcelona, Spain

⁴ CIMNE, International Center for Numerical Methods in Engineering, Barcelona, Spain

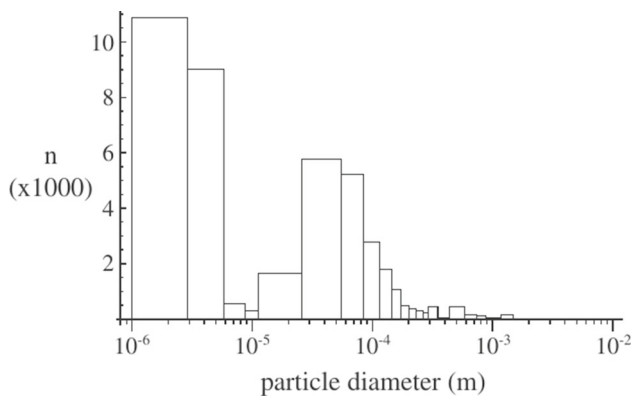


Fig. 1 Counts of particles of various diameters in air expelled by (90) coughs [72]

3 Virus lifetime outside the body

Current evidence points to lifetimes outside the body that can range from 1–2 h in air to several days on particular surfaces [45,94]. There has also been some documentation of lifetime variation depending on humidity.

4 Virus transmission

4.1 Human sneezing and coughing

In the sequel, we consider human sneezing and coughing as the main conduits of virus transmission. Clearly, breathing and talking will lead to the exhalation of air, and, consequently the exhalation of viruses for infected victims [2–4]. However, it stands to reason that the size and amount of particles released—and hence the amount of viruses in them—is much higher and much more concentrated when sneezing or coughing [3,4,32,44,49,90].

The velocity of air at a person's mouth during sneezing and coughing has been a source of heated debate, particularly in the media. The experimental evidence points to exit velocities of the order of 2–14 m/s [25,36–39,87–89]. A typical amount and size of particles can be seen in Fig. 1.

4.2 Sink velocities

If, for the sake of argument, we consider Stoke's law for the drag of spherical particles, valid below Reynolds numbers of $Re = 1$, the terminal sink velocity (also known as the settling velocity) of particles will be given by [26]:

$$v_s = \frac{(\rho_p - \rho_g)g \cdot d^2}{18\mu}, \quad (1)$$

Table 1 Sink velocities and Reynolds number for water particles in air

Diameter (mm)	Sink velocity (m/s)	Re
1.00E-00	3.01E+01	1.99E+03
1.00E-01	3.01E-01	1.99E+00
1.00E-02	3.01E-03	1.99E-03
1.00E-03	3.01E-05	1.99E-06
1.00E-04	3.01E-07	1.99E-09

where $\rho_p, \rho_g, g, \mu, d$ denote the density of the particles (essentially water in the present case), density of the gas (air), gravity, dynamic viscosity of the gas and diameter of the particle respectively. The equivalent Reynolds' number is:

$$Re = \frac{\rho_g v_s d}{\mu} = \frac{\rho_g (\rho_p - \rho_g) g d^3}{18\mu^2} \quad (2)$$

With $\rho_p = 1$ gr/cc, $\rho_g = 0.0012$ gr/cc, $g = 981$ cm/s², $\mu = 1.81 \cdot 10^{-4}$ gr/(cm s) this yields a limiting diameter for $Re = 1$ of

$$d_{Re=1} = 0.0079 \text{ cm}, \quad (3)$$

i.e. approximately 1/10th of a millimeter in diameter—a particle size that would still be perceived by the human eye. The corresponding sink velocity is given by:

$$v_s = 3 \cdot 10^5 d^2 \text{ cm/s}, \quad (4)$$

with d in cm, i.e. for $Re = 1$

$$v_s (Re = 1) = 18 \text{ cm/s}. \quad (5)$$

Note the quadratic dependency of the sink velocity with diameter. Table 1 lists the sink velocities for water droplets of different diameters in air. One can see that below diameters of $O(0.1 \text{ mm})$ the sink velocity is very low, implying that these particles remain in and move with the air for considerable time (and possibly distances).

4.3 Evaporation

Depending on the relative humidity and the temperature of the ambient air, the smaller particles can evaporate in milliseconds. However, as the mucous and saliva evaporate, they build a gel-like structure that surrounds the virus, allowing it to survive. This implies that extremely small particles with possible viruses will remain infectious for extended periods of times—up to an hour according to some studies.

An important question is then whether a particle/droplet will first reach the ground or evaporate. Figure 2, taken from

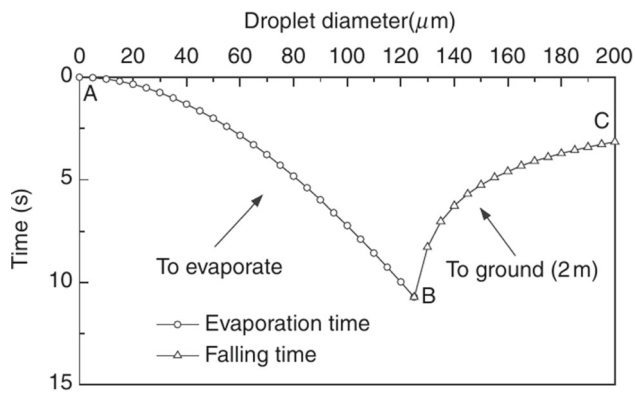


Fig. 2 Evaporation time and falling time of droplets of varying diameter ($T_{p0} = 33^\circ\text{C}$, $T_\infty = 18^\circ\text{C}$, $RH = 0\%$) from [96]

[96], shows that below $120\ \mu\text{m}$ the particles evaporate before falling 2 m (i.e. reaching the ground).

4.4 Viral load

A central question that requires an answer is then: how many viruses are in these small particles? An approximate answer may be obtained from the experiments that are being carried out on animals to trace and monitor infections. For ferrets [46] $O(10^5 - 10^6)$ have been used to infect via intranasal swabs, while for mice [92] $O(10^4)$ seem to suffice. Viral titers can vary a lot, but one may assume on the order of $O(10^6)$ viruses/ml for a nasopharyngeal swab [46,92]. Table 2 lists the number of viruses per droplet and the number of droplets needed to contain just 1 virus. Note that while for a droplet with a diameter of 1 mm one can expect $O(500)$ viruses, only every 2000th particle of diameter $10\ \mu\text{m}$ does contain a single virus.

Similar numbers are seen in field studies as well. The size of viruses varies from $0.02\text{--}0.3\ \mu\text{m}$, while the size of bacteria varies from $0.5\text{--}10\ \mu\text{m}$. The influenza virus RNA detected by quantitative polymerase chain reaction in human exhaled breath suggests that it may be contained in fine particles generated during tidal breathing and not only coughs [32,48,49,78]. Influenza RNA and Mycobacterium tuberculosis have been reported in particles that range in size from $0.5\text{--}4.0\ \mu\text{m}$ ([32,48,49,78] and references cited therein).

5 Physical modeling of aerosol propagation

When solving the two-phase equations, the air, as a continuum, is best represented by a set of partial differential equations (the Navier–Stokes equations) that are numerically solved on a mesh. Thus, the gas characteristics are calculated at the mesh points within the flowfield. However, as the

droplets/particles may be relatively sparse in the flowfield, they can be modeled by either:

- A continuum description, i.e. in the same manner as the fluid flow, or
- A particle (or Lagrangian) description, where individual particles (or groups of particles) are monitored and tracked in the flow.

Although the continuum (so-called multi-fluid) method has been used for some applications, the inherent assumptions of the continuum approach lead to several disadvantages which may be countered with a Lagrangian treatment for dilute flows. The continuum assumption cannot robustly account for local differences in particle characteristics, particularly if the particles are polydispersed. In addition, the only boundary conditions that can be considered in a straightforward manner are slipping and sticking, whereas reflection boundary conditions, such as specular and diffuse reflection, may be additionally considered with a Lagrangian approach. Numerical diffusion of the particle density is eliminated by employing Lagrangian particles due to their pointwise spatial accuracy.

While a Lagrangian approach offers many potential advantages, this method also creates problems that need to be addressed. For instance, large numbers of particles may cause a Lagrangian analysis to be memory intensive. This problem is circumvented by treating parcels of particles, i.e. doing the detailed analysis for one particle and then applying the effect to many. In addition, continuous mapping and remapping of particles to their respective elements may increase computational requirements, particularly for unstructured grids.

5.1 Equations describing the motion of the air

As seen from the experimental evidence, the velocities of air encountered during coughing and sneezing never exceed a Mach-number of $Ma = 0.1$. Therefore, the air may be assumed as a Newtonian, incompressible liquid, where buoyancy effects are modeled via the Boussinesq approximation. The equations describing the conservation of momentum, mass and energy for incompressible, Newtonian flows may be written as

$$\begin{aligned} \rho \mathbf{v}_{,t} + \rho \mathbf{v} \cdot \nabla \mathbf{v} + \nabla p \\ = \nabla \cdot \mu \nabla \mathbf{v} + \rho \mathbf{g} + \beta \rho \mathbf{g} (T - T_0) + \mathbf{s}_v, \end{aligned} \quad (6.1)$$

$$\nabla \cdot \mathbf{v} = 0. \quad (6.2)$$

$$\rho c_p T_{,t} + \rho c_p \mathbf{v} \cdot \nabla T = \nabla \cdot k \nabla T + s_e, \quad (6.3)$$

Table 2 Estimated number of viruses for different particle diameters

Droplet diameter (mm)	Volume (mm ³)	Viruses/droplet	Droplets needed for 1 virus
1.00E+00	5.24E−01	5.24E+02	1.00E+00
1.00E−01	5.24E−04	5.24E−01	1.91E+00
1.00E−02	5.24E−07	5.24E−04	1.91E+03
1.00E−03	5.24E−10	5.24E−07	1.91E+06

Here ρ , $\mathbf{v}$, p , μ , $\mathbf{g}$, β , T , T_0 , c_p , k denote the density, velocity vector, pressure, viscosity, gravity vector, coefficient of thermal expansion, temperature, reference temperature, specific heat coefficient and conductivity respectively, and $\mathbf{s}_v$, s_e momentum and energy source terms (e.g. due to particles or external forces/heat sources). For turbulent flows both the viscosity and the conductivity are obtained either from additional equations or directly via a large eddy simulation (LES) assumption through monotonicity induced LES (MILES) [18,34,35,40,42,98].

5.2 Equations describing the motion of particles/droplets

In order to describe the interaction of particles/droplets with the flow, the mass, forces and energy/work exchanged between the flowfield and the particles must be defined. As before, we denote for **fluid (air)** by ρ , p , T , k , v_i , μ and c_p the density, pressure, temperature, conductivity, velocity in direction x_i , viscosity, and the specific heat at constant pressure. For the **particles**, we denote by ρ_p , T_p , v_{pi} , d , c_{pp} and Q the density, temperature, velocity in direction x_i , equivalent diameter, and heat transferred per unit volume. In what follows, we will refer to droplet and particles, collectively as particles.

Making the classical assumptions that the particles may be represented by an equivalent sphere of diameter d , the drag forces $\mathbf{D}$ acting on the particles will be due to the difference of fluid and particle velocity:

$$\mathbf{D} = \frac{\pi d^2}{4} \cdot c_D \cdot \frac{1}{2} \rho |\mathbf{v} - \mathbf{v}_p| (\mathbf{v} - \mathbf{v}_p). \quad (7)$$

The **drag coefficient** c_D is obtained empirically from the Reynolds-number Re :

$$Re = \frac{\rho |\mathbf{v} - \mathbf{v}_p| d}{\mu} \quad (8)$$

as (see, e.g. [83]):

$$c_D = \max \left(0.1, \frac{24}{Re} \left(1 + 0.15 Re^{0.687} \right) \right) \quad (9)$$

The lower bound of $c_D = 0.1$ is required to obtain the proper limit for the Euler equations, when $Re \rightarrow \infty$. The heat transferred between the particles and the fluid is given by

$$Q = \frac{\pi d^2}{4} \cdot \left[h_f \cdot (T - T_p) + \sigma^* \cdot (T^4 - T_p^4) \right], \quad (10)$$

where h_f is the film coefficient and σ^* the radiation coefficient. For the class of problems considered here, the particle temperature and kinetic energy are such that the radiation coefficient σ^* may be ignored. The film coefficient h_f is obtained from the Nusselt-number Nu :

$$Nu = 2 + 0.459 Pr^{0.333} Re^{0.55}, \quad (11)$$

where Pr is the Prandtl-number of the gas

$$Pr = \frac{k}{\mu}, \quad (12)$$

as

$$h_f = \frac{Nu \cdot k}{d}. \quad (13)$$

Having established the forces and heat flux, the particle motion and temperature are obtained from Newton's law and the first law of thermodynamics. For the particle velocities, we have:

$$\rho_p \frac{\pi d^3}{6} \cdot \frac{d\mathbf{v}_p}{dt} = \mathbf{D} + \rho_p \frac{\pi d^3}{6} \mathbf{g}. \quad (14)$$

This implies that:

$$\begin{aligned} \frac{d\mathbf{v}_p}{dt} &= \frac{3\rho}{4\rho_p d} \cdot c_d |\mathbf{v} - \mathbf{v}_p| (\mathbf{v} - \mathbf{v}_p) + \mathbf{g} \\ &= \alpha_v |\mathbf{v} - \mathbf{v}_p| (\mathbf{v} - \mathbf{v}_p) + \mathbf{g}. \end{aligned} \quad (15)$$

where $\alpha_v = 3\rho c_d / (4\rho_p d)$. The particle positions are obtained from:

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p. \quad (16)$$

The temperature change in a particle is given by:

$$\rho_p c_{pp} \frac{\pi d^3}{6} \cdot \frac{dT_p}{dt} = Q, \quad (17)$$

which may be expressed as:

$$\frac{dT_p}{dt} = \frac{3k}{4c_{pp}\rho_p d^2} \cdot Nu \cdot (T - T_p) = \alpha_T (T - T_p), \quad (18)$$

with $\alpha_T = 3k/(4c_{pp}\rho_p d^2)$. Equations (15, 16, 18) may be formulated as a system of Ordinary Differential Equations (ODEs) of the form:

$$\frac{d\mathbf{u}_p}{dt} = \mathbf{r}(\mathbf{u}_p, \mathbf{x}, \mathbf{u}_f), \quad (19)$$

where $\mathbf{u}_p$, $\mathbf{x}$, $\mathbf{u}_f$ denote the particle unknowns, the position of the particle and the fluid unknowns at the position of the particle.

5.3 Numerical integration of the motion of the air

The last six decades have seen a large number of schemes that may be used to solve numerically the incompressible Navier–Stokes equations given by Eqs. (6.1–6.3). In the present case, the following design criteria were implemented:

- Spatial discretization using **unstructured grids** (in order to allow for arbitrary geometries and adaptive refinement);
- Spatial approximation of unknowns with **simple linear finite elements** (in order to have a simple input/output and code structure);
- Edge-based data structures (for reduced access to memory and indirect addressing);
- Temporal approximation using **implicit integration of viscous terms and pressure** (the interesting scales are the ones associated with advection);
- Temporal approximation using **explicit, high-order integration of advective terms**;
- **Low-storage, iterative solvers** for the resulting systems of equations (in order to solve large 3-D problems); and
- Steady results that are **independent from the timestep** chosen (in order to have confidence in convergence studies).

The resulting discretization in time is given by the following projection scheme [60,63]:

- *Advective-Diffusive Prediction*: $\mathbf{v}^n, p^n \rightarrow \mathbf{v}^*$

$$\mathbf{s}' = -\nabla p^n + \rho \mathbf{g} + \beta \rho \mathbf{g}(T^n - T_0) + \mathbf{s}_v, \quad (20)$$

$$\mathbf{v}^i = \mathbf{v}^n + \alpha^i \gamma \Delta t \left(-\mathbf{v}^{i-1} \cdot \nabla \mathbf{v}^{i-1} \nabla \cdot \mu \nabla \mathbf{v}^{i-1} + \mathbf{s}' \right); \quad (21a)$$

$$\begin{aligned} i &= 1, k-1; \\ \left[\frac{1}{\Delta t} - \theta \nabla \cdot \mu \nabla \right] (\mathbf{v}^k - \mathbf{v}^n) &+ \mathbf{v}^{k-1} \cdot \nabla \mathbf{v}^{k-1} \\ &= \nabla \cdot \mu \nabla \mathbf{v}^{k-1} + \mathbf{s}'. \end{aligned} \quad (21b)$$

– *Pressure Correction*: $p^n \rightarrow p^{n+1}$

$$\nabla \cdot \mathbf{v}^{n+1} = 0; \quad (22)$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^*}{\Delta t} + \nabla(p^{n+1} - p^n) = 0; \quad (23)$$

which results in

$$\nabla^2(p^{n+1} - p^n) = \frac{\nabla \cdot \mathbf{v}^*}{\Delta t}; \quad (24)$$

– *Velocity Correction*: $\mathbf{v}^* \rightarrow \mathbf{v}^{n+1}$

$$\mathbf{v}^{n+1} = \mathbf{v}^* - \Delta t \nabla(p^{n+1} - p^n). \quad (25)$$

θ denotes the implicitness-factor for the viscous terms ($\theta = 1$: 1st order, fully implicit, $\theta = 0.5$: 2nd order, Crank–Nicholson). α^i are the standard low-storage Runge–Kutta coefficients $\alpha^i = 1/(k+1-i)$. The $k-1$ stages of Eq. (21a) may be seen as a predictor (or replacement) of $\mathbf{v}^n$ by $\mathbf{v}^{k-1}$. The original right-hand side has not been modified, so that at steady-state $\mathbf{v}^n = \mathbf{v}^{k-1}$, preserving the requirement that the steady-state be independent of the timestep Δt . The factor γ denotes the local ratio of the stability limit for explicit timestepping for the viscous terms versus the timestep chosen. Given that the advective and viscous timestep limits are proportional to:

$$\Delta t_a \approx \frac{h}{|\mathbf{v}|}; \Delta t_v \approx \frac{\rho h^2}{\mu}, \quad (26)$$

we immediately obtain

$$\gamma = \frac{\Delta t_v}{\Delta t_a} \approx \frac{\rho |\mathbf{v}| h}{\mu} \approx Re_h, \quad (27)$$

or, in its final form:

$$\gamma = \min(1, Re_h). \quad (28)$$

In regions away from boundary layers, this factor is $O(1)$, implying that a high-order Runge–Kutta scheme is recovered. Conversely, for regions where $Re_h = O(0)$, the scheme reverts back to the usual 1-stage Crank–Nicholson scheme. Besides higher accuracy, an important benefit of explicit multistage advection schemes is the larger timestep one can

employ. The increase in allowable timestep is roughly proportional to the number of stages used (and has been exploited extensively for compressible flow simulations [43]). Given that for an incompressible solver of the projection type given by Eqs. (20–25) most of the CPU time is spent solving the pressure-Poisson system Eq. (24), the speedup achieved is also roughly proportional to the number of stages used.

At steady state, $\mathbf{v}^* = \mathbf{v}^n = \mathbf{v}^{n+1}$ and the residuals of the pressure correction vanish, implying that the result does not depend on the timestep Δt .

The spatial discretization of these equations is carried out via linear finite elements. The resulting matrix system is rewritten as an edge-based solver, allowing the use of consistent numerical fluxes to stabilize the advection and divergence operators [63].

The energy (temperature) equation [Eq. (6.3)] is integrated in a manner similar to the advective-diffusive prediction [Eq. (21)], i.e. with an explicit, high order Runge-Kutta scheme for the advective parts and an implicit, 2nd order Crank–Nicholson scheme for the conductivity.

5.4 Numerical integration of the motion of particles/droplets

The equations describing the position, velocity and temperature of a particle [Eqs. (15)–(19)] may be formulated as a system of nonlinear Ordinary Differential Equations of the form:

$$\frac{d\mathbf{u}_p}{dt} = \mathbf{r}(\mathbf{u}_p, \mathbf{x}, \mathbf{u}_f). \quad (29)$$

They can be integrated numerically in a variety of ways. Due to its speed, low memory requirements and simplicity, we have chosen the following k -step low-storage Runge–Kutta procedure to integrate them:

$$\mathbf{u}_p^{n+i} = \mathbf{u}_p^n + \alpha^i \Delta t \cdot \mathbf{r}(\mathbf{u}_p^{n+i-1}, \mathbf{x}^{n+i-1}, \mathbf{u}_f^{n+i-1}), \quad i = 1, k. \quad (30)$$

For linear ODEs the choice

$$\alpha^i = \frac{1}{k+1-i}, \quad i = 1, k \quad (31)$$

leads to a scheme that is k -th order accurate in time. Note that in each step the location of the particle with respect to the fluid mesh needs to be updated in order to obtain the proper values for the fluid unknowns. The default number of stages used is $k = 4$. This would seem unnecessarily high, given that the flow solver is of second-order accuracy, and that the particles are integrated separately from the flow solver before the next (flow) timestep, i.e. in a staggered manner. However, it was found that the 4-stage particle integration preserves

very well the motion in vortical structures and leads to less ‘wall sliding’ close to the boundaries of the domain [70]. The stability/accuracy of the particle integrator should not be a problem as the particle motion will always be slower than the maximum wave speed of the fluid (fluid velocity).

The transfer of forces and heat flux between the fluid and the particles must be accomplished in a conservative way, i.e. whatever is added to the fluid must be subtracted from the particles and vice-versa. The finite element discretization of the fluid equations will lead to a system of ODE’s of the form:

$$\mathbf{M}\Delta\mathbf{u} = \mathbf{r}, \quad (32)$$

where $\mathbf{M}$, $\Delta\mathbf{u}$ and $\mathbf{r}$ denote, respectively, the consistent mass matrix, increment of the unknowns vector and right-hand side vector. Given the ‘host element’ of each particle, i.e. the fluid mesh element that contains the particle, the forces and heat transferred to $\mathbf{r}$ are added as follows:

$$\mathbf{r}_D^i = \sum_{el \text{ surr } i} N^i(\mathbf{x}_p) \mathbf{D}_p. \quad (33)$$

Here $N^i(\mathbf{x}_p)$ denotes the shape-function values of the host element for the point coordinates $\mathbf{x}_p$, and the sum extends over all elements that surround node i . As the sum of all shape-function values is unity at every point:

$$\sum N^i(\mathbf{x}) = 1 \forall \mathbf{x}, \quad (34)$$

this procedure is strictly conservative.

From Eqs. (15–18) and their equivalent numerical integration via Eq. (30), the change in momentum and energy for one particle is given by:

$$\mathbf{f}_p = \rho_p \frac{\pi d^3}{6} \frac{(\mathbf{v}_p^{n+1} - \mathbf{v}_p^n)}{\Delta t}, \quad (35)$$

$$q_p = \rho_p c_{pp} \frac{\pi d^3}{6} \frac{(T_p^{n+1} - T_p^n)}{\Delta t}. \quad (36)$$

These quantities are multiplied by the number of particles in a packet in order to obtain the final values transmitted to the fluid. Before going on, we summarize the basic steps required in order to update the particles one timestep:

- Initialize Fluid Source-Terms: $\mathbf{r} = 0$
- DO: For Each Particle:
 - DO: For Each Runge–Kutta Stage:
 - Find Host Element of Particle: IELEM, $N^i(\mathbf{x})$
 - Obtain Fluid Variables Required
 - Update Particle: Velocities, Position, Temperature, ...

- ENDDO
- Transfer Loads to Element Nodes
- ENDDO

5.4.1 Particle parcels

For a large number of very small particles, it becomes impossible to carry every individual particle in a simulation. The solution is to:

- (a) Agglomerate the particles into so-called packets of N_p particles;
- (b) Integrate the governing equations for one individual particle; and
- (c) Transfer back to the fluid N_p times the effect of one particle.

Beyond a reasonable number of particles per element (typically > 8), this procedure produces accurate results without any deterioration in physical fidelity.

5.4.2 Other numerics

In order to achieve a robust particle integrator, a number of additional precautions and algorithms need to be implemented. The most important of these are:

- Agglomeration/Subdivision of Particle Parcels: As the fluid mesh may be adaptively refined and coarsened in time, or the particle traverses elements of different sizes, it may be important to adapt the parcel concentrations as well. This is necessary to ensure that there is sufficient parcel representation in each element and yet, that there are not too many parcels as to constitute an inefficient use of CPU and memory.
- Limiting During Particle Updates: As the particles are integrated independently from the flow solver, it is not difficult to envision situations where for the extreme cases of very light or very heavy particles physically meaningless or unstable results may be obtained. In order to prevent this, the changes in particle velocities and temperatures are limited in order not to exceed the differences in velocities and temperature between the particles and the fluid [70].
- Particle Contact/Merging: In some situations, particles may collide or merge in a certain region of space.
- Particle Tracking: A common feature of all particle-grid applications is that the particles do not move far between timesteps. This makes physical sense: if a particle jumped ten gridpoints during one timestep, it would have no chance to exchange information with the points along the way, leading to serious errors. Therefore, the assumption

that the new host elements of the particles are in the vicinity of the current ones is a valid one. For this reason, the most efficient way to search for the new host elements is via the vectorized neighbour-to-neighbour algorithm described in [51,63].

6 Examples

The techniques described above were implemented in FEFLO, a general-purpose computational fluid dynamics (CFD) code based on the following general principles:

- Use of unstructured grids (automatic grid generation and mesh refinement);
- Finite element discretization of space;
- Separate flow modules for compressible and incompressible flows;
- Edge-based data structures for speed;
- Optimal data structures for different architectures;
- Bottom-up coding from the subroutine level to assure an open-ended, expandable architecture.

The code has had a long history of relevant applications involving compressible flow simulations in the areas of transonic flow [55,73–77], store separation [7,10,12,14,15], blast-structure interaction [6,8,9,11,13,16,17,58,64,82,85,93], incompressible flows [5,57,60,68,79,81,91], free-surface hydrodynamics [54,61,62], dispersion [20–22,59], patient-based haemodynamics [1,23,24,55,65] and aeroacoustics [50]. The code has been ported to vector [56], shared memory [53,84], distributed memory [52,66,79,80] and GPU-based [27–30,67,69] machines.

The cases shown all simulate sneezing/coughing in different environments. The ambient temperature was assumed to be 20°C . In order to simulate a sneeze/cough, the velocity and temperature in a spherical region of radius ($r = 5\text{ cm}$) near the patient's mouth was reset at the beginning of each timestep according to the following triangular function:

$$f(t) = \begin{cases} \frac{t}{t_{mid}} & \text{if } 0 \leq t \leq t_{mid} \\ 1 - \frac{t-t_{mid}}{t_{mid}} & \text{if } t_{mid} \leq t \leq 2t_{mid} \\ 0 & \text{if } 2t_{mid} \leq t \end{cases} \quad (37)$$

$$v(t) = 5f(t) \text{ [m/s]}, \quad T(t) = 20 + f(t)(37 - 20). \quad (38)$$

The droplets were initialized with 4 different sizes and different velocities, and released every 0.005 s during 0.1 s. This resulted in a final number of particle packets of $n_p = 25,662$. The temperature was set to $T_p = 37^\circ\text{C}$ and the velocity to $v = 5\text{ m/s}$. Table 3 summarizes the diameters and resulting mass distribution.

In the cases shown different temporal scales appear:

- The fast, ballistic drop of the larger ($d = 1$ mm) particles, occurring in the range of $O(1)$ s;
- The slower drop of particles of diameter $d = O(0.1)$ mm, occurring in the range of $O(10)$ s; and
- The transport of the even smaller particles through the air, occurring in the range of $O(100)$ s.

We have attempted to show these phases in the results, and for this reason the results are not displayed at equal time intervals. Unless otherwise noted, the particles have been colored according to the **logarithm** of the diameter, with red colors representing the largest and blue the smallest particles.

The examples given show clearly the dangers of droplet- and aerosol- based infections in the built environment.

6.1 Sneezing in transportation security agency (TSA) queues

One of the obvious vectors for viral contamination and spread are security and passport examination queues in airports. Air flow is moderate, passengers are in very close proximity, and in some airports queues wind back and forth in narrow lanes. Figure 3a, b show the arrangement of pedestrians, as well as the discretization chosen. Note the smaller elements close to the bodies and in the region of interest between the two pedestrians in the middle row. This particular mesh had $12.74\text{Me}1\text{s}$. The distribution of particles and the absolute value of the velocity in the centerplane over time can be discerned from Figures 3c–e. One can see that the large (red) particles follow a ballistic path and have some influence on the flow (e.g. at time $t = 0.20$). This ‘ballistic phase’ ends at about $t = 1$ s. The (green) particles of size $d = 0.1$ mm are quickly stopped by the air, and then sink slowly towards the floor in close proximity to the individual sneezing. The even smaller (cyan, blue) particles rise with the cloud of warmer air exhaled by the sneezing individual, and disperse much further at later times.

6.2 Sneezing in a generic hospital room

This case considers a typical hospital room. Of interest here was the dispersion of particles in the **first minute** after cough-

ing, in particular the reach into neighbouring halls and the amount of ‘negative pressure’ needed to keep all contaminants in the room. Figure 4a shows the arrangement of the room, with patient and caregiver clearly visible. This particular mesh had $2.25\text{Me}1\text{s}$. The distribution of particles over time can be discerned from Figures 4c–l. As before, one can see that the large (red) particles follow a ballistic path. This ‘ballistic phase’ ends at about $t = 1$ s. The (green) particles of size $d = 0.1$ mm are quickly stopped by the air, and then sink slowly towards the patient. The even smaller (cyan, blue) particles rise with the cloud of warmer air exhaled by the sneezing individual, and disperse much further at later times, covering almost the entire room. The velocity distribution in the room may be inferred from Figure 4m.

7 Reopening after the crisis

A lingering question facing all levels of society is how and when to reopen facilities where people congregate in close proximity. One key technology that would allow opening is testing and sensing. We consider sensing in the sequel. Several vendors have announced measuring devices for Covid-19 in the next half year. Given that these sensors are expensive, and that a hospital or university many need hundreds of these, the question becomes how best to deploy them. In other words: given an arbitrary number of contamination or infection scenarios, which is the minimum number of sensors needed to detect them, and where should they be placed? A partial answer to this non-trivial question was given in [59,97]. If we assume a given number of sensors, every contaminant/infection scenario (location and amount of release, flow conditions, etc.) will lead to a sensor input. The data recorded from all the possible release scenarios at all possible sensor locations allows the identification of the best or optimal sensor locations. Clearly, if only one sensor is to be placed, it should be at the location that recorded the highest number of releases. This argument can be used recursively by removing from further consideration all releases already recorded by sensors previously placed. The procedure is repeated recursively until no undetected release cases are left, or the available sensors have been exhausted.

See [19,41] for an in-depth analysis of robust sensor placement under uncertainty.

Table 3 Initial conditions for particles

Droplet diameter (mm)	Mass (gr^3)	Nr. of packets	Nr. of particles
1.00E+00	5.50E+00	1.05E+03	1.05E+04
1.00E−01	0.11E+00	2.10E+03	2.10E+05
1.00E−02	0.58E−02	1.12E+04	1.12E+07
1.00E−03	0.58E−05	1.12E+04	1.12E+08

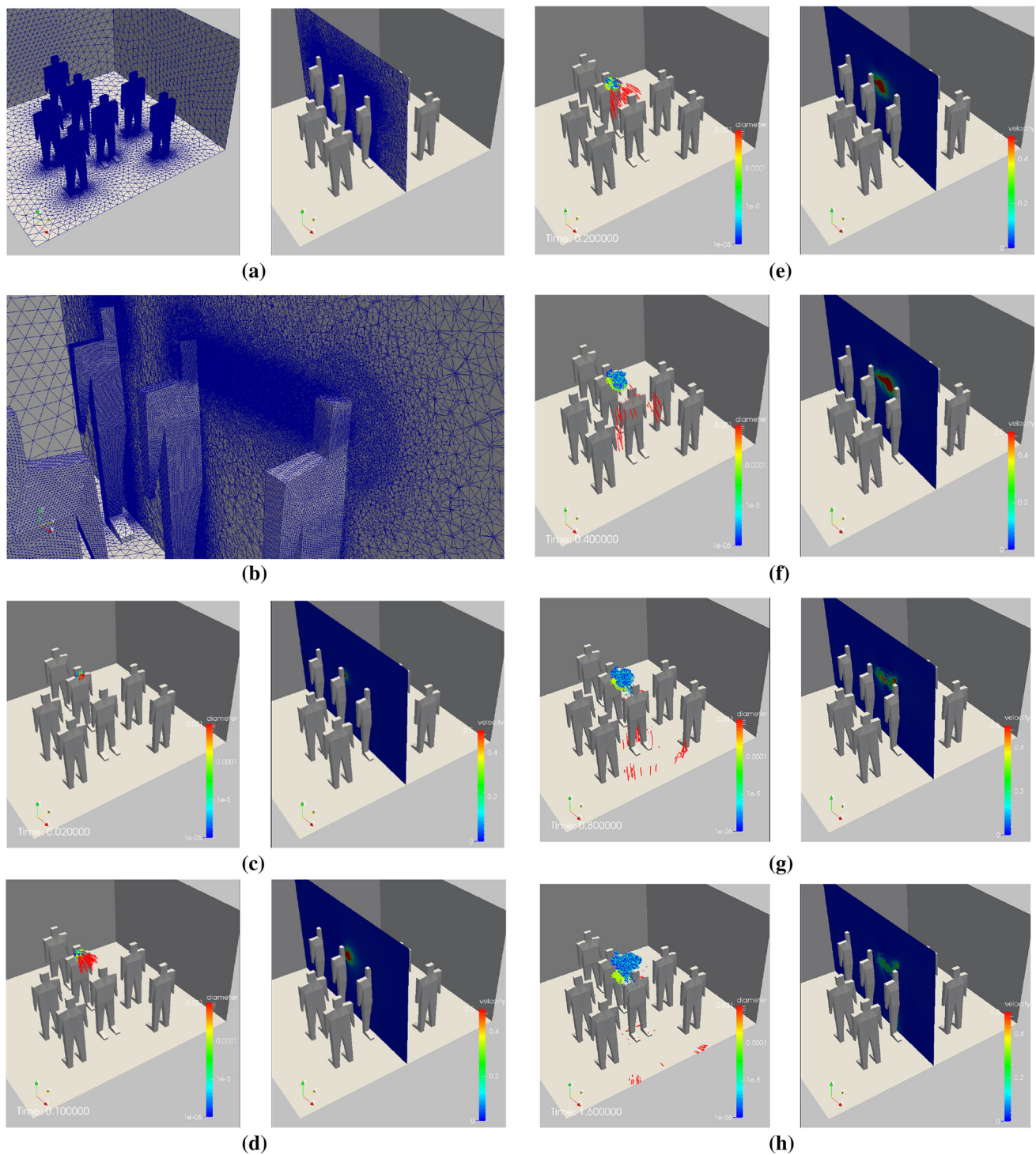


Fig. 3 TSA Queue: **a** arrangement of pedestrians and surface mesh, **b** surface mesh and cut plane, **c** particle distribution at $t = 0.02$ s, **d** particle distribution at $t = 0.10$ s, **e** particle distribution at $t = 0.20$ s, **f** particle distribution at $t = 0.40$ s, **g** particle distribution at $t = 0.80$ s, **h** particle distribution at $t = 1.60$ s

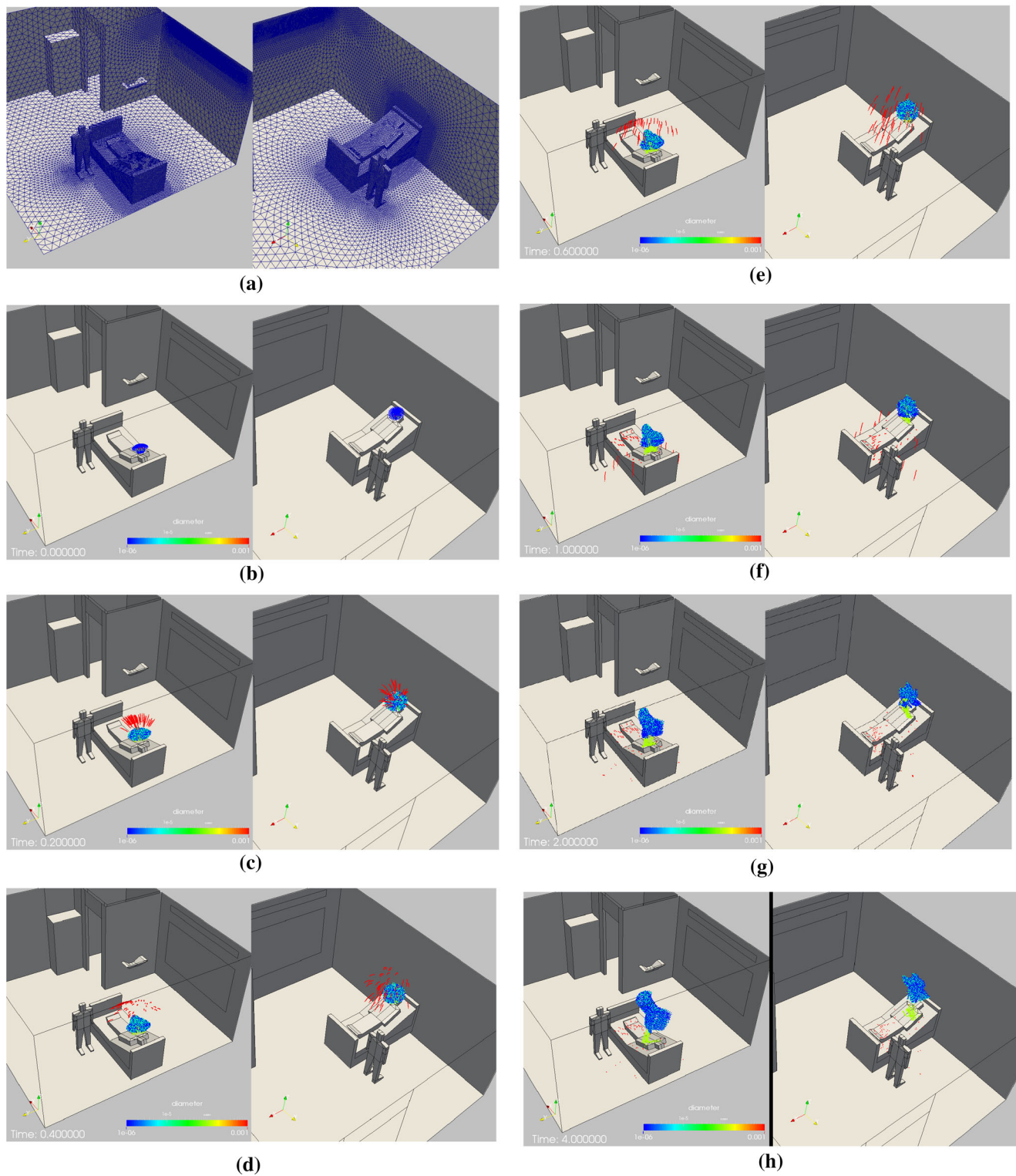


Fig. 4 Hospital room: **a** surface mesh, **b** particle distribution at $t = 0.0$ s, **c** particle distribution at $t = 0.2$ s, **d** particle distribution at $t = 0.4$ s, **e** particle distribution at $t = 0.6$ s, **f** particle distribution at $t = 1.0$ s, **g** particle distribution at $t = 2.0$ s, **h** particle distribution at

$t = 4.0$ s, **i** particle distribution at $t = 10.0$ s, **j** particle distribution at $t = 20.0$ s, **k** particle distribution at $t = 30.0$ s, **l** particle distribution at $t = 50.0$ s, **m** surface velocities at $t = 60.0$ s

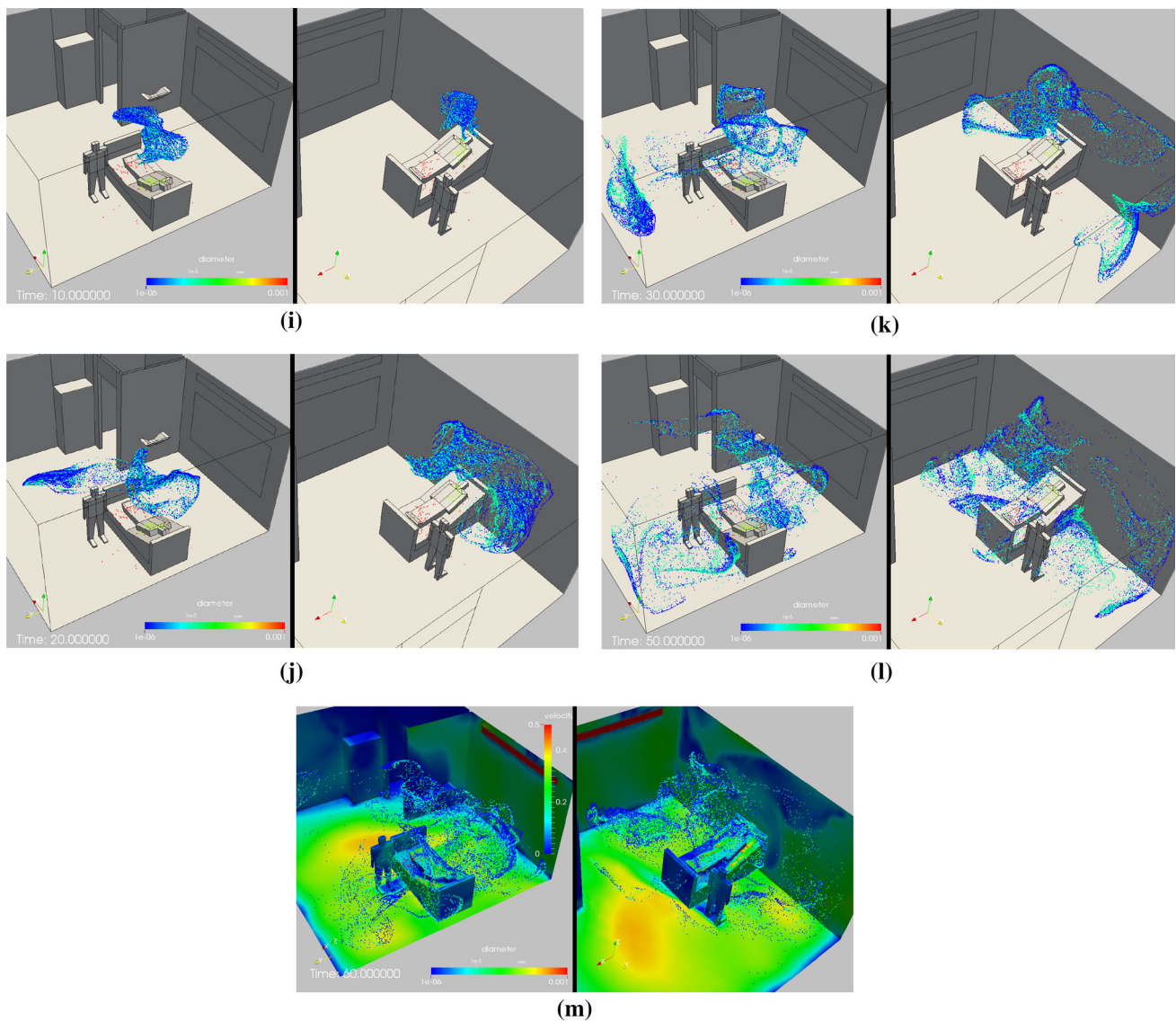


Fig. 4 continued

7.1 Hospital room

This case considers the same hospital room as shown before. The boundary conditions determining the flow are assumed as steady, with air entering the room through vents 1-3 and exiting the room through the bathroom exhaust or the door. Figures 5a–c show the outlay of the room, average velocities and the ‘age of air’ after 5 min. Note the high values for the age of air in the corners and the back of the room. This particular mesh had 2.2Me1s . Four contaminant release scenarios were considered: cases 1-3 assumed contaminant coming in through each of the vents (separately) during the first minute, while case 4 assumed virus production from the patient for a period of 10 s. The case was run for 5 min of real time, and the contaminant concentration was measured on all

walls/ceilings. The maximum concentrations measured have been summarized in Fig. 5d. Note the different areas covered depending on the release scenario. It was assumed that sensors should only be allowed above a certain height, and should be located on a wall or the ceiling. Table 4 summarizes the points that measured data above a set threshold. As one can see, none of the possible sensor locations is able to measure/detect all 4 cases, and many possible sensor locations do not detect even a single case. There are many possible pairs of sensors that can detect all 4 cases. The pair selected is the one that achieves the highest relative measurement values, and is shown in Fig. 5e. Note that this makes good sense: one sensor close the HVAC exits, and one close to the patient.

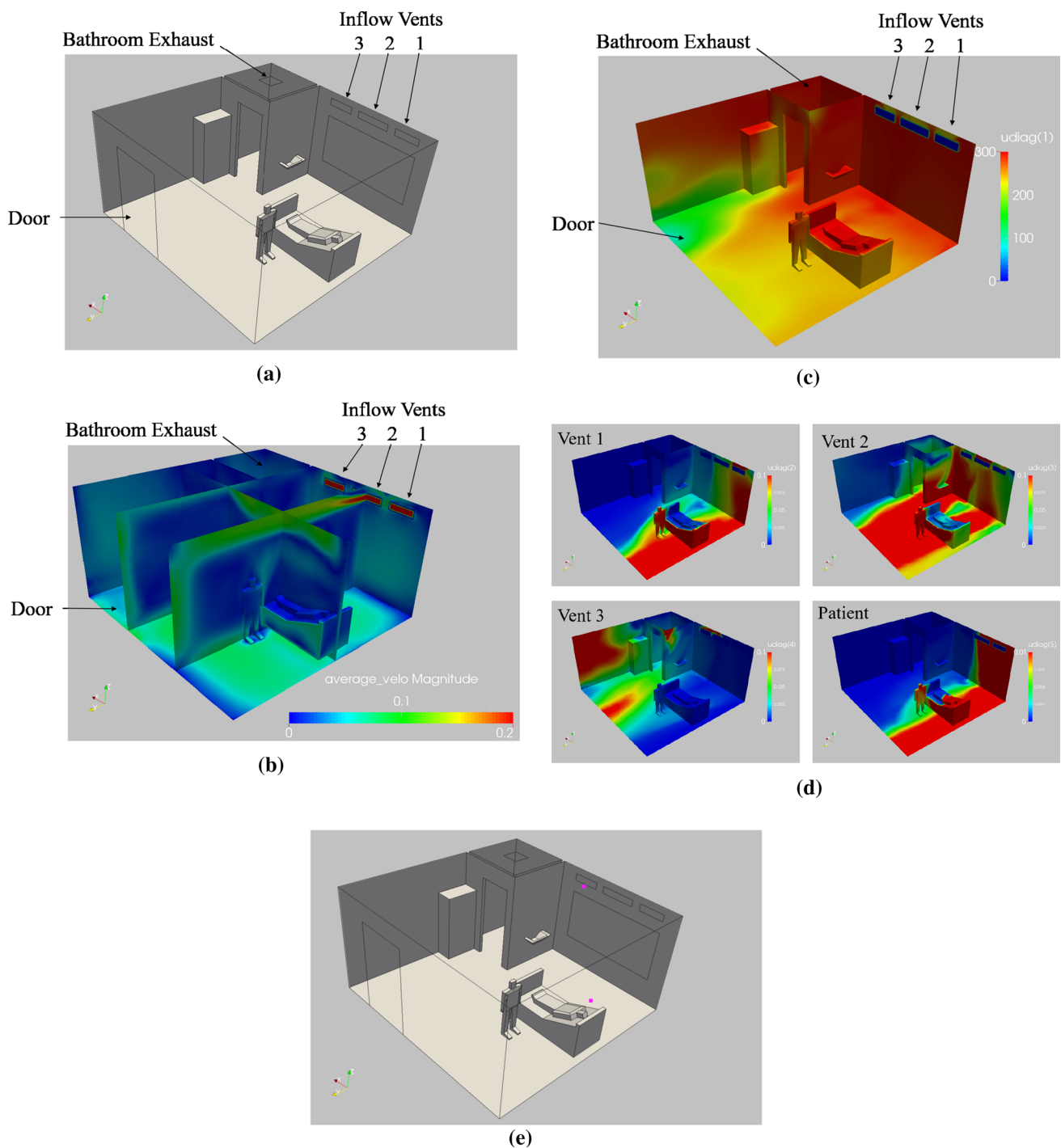


Fig. 5 Hospital room: **a** outlay of room and boundary conditions, **b** average velocities (5 min), **c** age of air (5 min), **d** maximum contaminant concentration over 5 min, **e** optimal sensor locations

8 Conclusions and outlook

The present paper has summarized some of the mechanical characteristics of virus contaminants and the transmission via droplets and aerosols. The ordinary and partial differential equations describing the physics of these processes with

high fidelity were given, as well as appropriate numerical schemes to solve them. Several examples taken from recent evaluations of the built environment were given, as well as the optimal placement of sensors.

Current efforts are directed at increasing the realism of the physical processes modeled (e.g. by adding the effect

Table 4 Data measurement summary

Cases measured	Number
0	4308
1	3377
2	1010
3	0
4	0

of moving pedestrians [71]), streamlining the simulation toolbox and workflow, and fielding these tools so that the post-pandemic opening can occur as smoothly as possible.

Acknowledgements H. Antil is partially supported by NSF Grants DMS-1818772, DMS-1913004, the Air Force Office of Scientific Research (AFOSR) under Award No.: FA9550-19-1-0036, and Department of Navy, Naval PostGraduate School under Award No.: N00244-20-1-0005. E. Oñate and S. Idelsohn acknowledge financial support from the Spanish Ministry of Economy and Competitiveness, through the “Severo Ochoa Programme for Centres of Excellence in R&D” (CEX2018-000797-S).

References

- Appanaboyina S, Mut F, Löhner R, Putman CM, Cebal JR (2008) Computational fluid dynamics of stented intracranial aneurysms using adaptive embedded unstructured grids. *Int J Numer Methods Fluids* 57(5):475–493
- Asadi S, Wexler AS, Cappa CD, Barreda S, Bouvier NM, Ristenpart W (2019) Aerosol emission and superemission during human speech increase with voice loudness. *Nat Sci Rep* 9(1):2348. <https://doi.org/10.1038/s41598-019-38808-z>
- Asadi S, Wexler AS, Cappa CD, Barreda S, Bouvier NM, Ristenpart W (2020) Effect of voicing and articulation manner on aerosol particle emission during human speech. *PLoS One* 15(1):e0227699. <https://doi.org/10.1371/journal.pone.0227699>
- Asadi S, Bouvier NM, Wexler AS, Ristenpart W (2020) The coronavirus pandemic and aerosols: does COVID-19 transmit via expiratory particles? *Aerosol Sci Technol*. <https://doi.org/10.1080/02786826.2020.1749229>
- Aubry R, Mut F, Löhner R, Cebal JR (2008) Deflated preconditioned conjugate gradient solvers for the pressure-Poisson equation. *J Comput Phys* 227(24):10196–10208
- Baum JD, Löhner R (1991) Numerical simulation of shock interaction with a modern main battlefield tank. AIAA-91-1666
- Baum JD, Löhner R (1993) Numerical simulation of pilot/seat ejection from an F-16. AIAA-93-0783
- Baum JD, Luo H, Löhner R (1993) Numerical simulation of a blast inside a Boeing 747. AIAA-93-3091
- Baum JD, Luo H, Löhner R (1993) Numerical simulation of a blast within a multi-room shelter. In: *Proceedings of MABS-13 conference*, The Hague, Netherlands, pp 451–463
- Baum JD, Luo H, Löhner R (1994) A new ALE adaptive unstructured methodology for the simulation of moving bodies. AIAA-94-0414
- Baum JD, Luo H, Löhner R (1995) Numerical simulation of blast in the World Trade Center. AIAA-95-0085
- Baum JD, Luo H, Löhner R (1995) Validation of a new ALE, adaptive unstructured moving body methodology for multi-store ejection simulations. AIAA-95-1792
- Baum JD, Luo H, Löhner R, Yang C, Pelessone D, Charman C (1996) Coupled fluid/structure modeling of shock interaction with a truck. AIAA-96-0795
- Baum JD, Luo H, Löhner R, Goldberg E, Feldhun A (1997) Application of unstructured adaptive moving body methodology to the simulation of fuel tank separation from an F-16 c/d fighter. AIAA-97-0166
- Baum JD, Löhner R, Marquette TJ, Luo H (1997) Numerical simulation of aircraft canopy trajectory. AIAA-97-1885
- Baum JD, Luo H, Mestreau E, Löhner R, Pelessone D, Charman C (1999) Coupled CFD/CSD methodology for modeling weapon detonation and fragmentation. AIAA-99-0794
- Baum JD, Mestreau E, Luo H, Löhner R, Pelessone D, Giltrud ME, Gran JK (2006) Modeling of near-field blast wave evolution. AIAA-06-0191
- Boris JP, Grinstein FF, Oran ES, Kolbe RJ (1992) New insights into large eddy simulation. *Fluid Dyn Res* 10:199–228
- Burns JA, Rautenberg CN (2015) The infinite-dimensional optimal filtering problem with mobile and stationary sensor networks. *Numer Funct Anal Optim* 36(2):181–224. <https://doi.org/10.1080/01630563.2014.970647>
- Camelli F, Löhner R (2004) Assessing maximum possible damage for contaminant release events. *Eng Comput* 21(7):748–760
- Camelli F, Löhner R, Sandberg WC, Ramamurti R (2004) VLES study of ship stack gas dynamics. AIAA-04-0072
- Camelli F, Löhner R (2006) VLES study of flow and dispersion patterns in heterogeneous urban areas. AIAA-06-1419
- Cebal JR, Löhner R (2001) From medical images to anatomically accurate finite element grids. *Int J Numer Methods Eng* 51:985–1008
- Cebal JR, Löhner R (2005) Efficient simulation of blood flow past complex endovascular devices using an adaptive embedding technique. *IEEE Trans Med Imaging* 24(4):468–476
- Chao C, Wan MP, Morawska L, Johnson G, Graham R, Ristovski Z, Hargreaves M, Mengersen K, Kerrie L, Steve C, Li Y, Xie X, Katoheviski S (2009) Characterization of expiration air jets and droplet size distributions immediately at the mouth opening. *J Aerosol Sci* 40(2):122–133
- Clift R, Grace JR, Weber ME (1978) Bubbles, drops and particles. Academic Press, New York
- Corrigan A, Camelli F, Löhner R (2010) Porting Of an edge-based CFD solver to GPUs. AIAA-10-0523
- Corrigan A, Camelli F, Löhner R, Mut F (2010) Porting of FEFLO to GPUs. In: *Proceedings of ECCOMAS CFD 2010 conference*, June 14–17, Lisbon, Portugal
- Corrigan A, Camelli FF, Löhner R, Wallin J (2011) Running unstructured grid based CFD solvers on modern graphics hardware. *Int J Numer Methods Fluids* 66:221–229
- Corrigan A, Löhner R (2011) Porting of FEFLO to multi-GPU clusters. AIAA-11-0948
- Dietz L, Horve PF, Coil DA, Fretz M, Eisen JA, van den Wymelenberg L (2020) 2019 novel coronavirus (COVID-19) pandemic: built environment considerations to reduce transmission. *mSystems* 5:e00245-20. <https://doi.org/10.1128/mSystems.00245-20>
- Fabian P, McDevitt JJ, Dehaan WH, Fung ROP, Cowling BJ, Chan KH et al (2008) Influenza virus in human exhaled breath: an observational study. *PLoS One* 3:e2691
- Franz DR, Jahrling PB, Friedlander AM et al (1997) Clinical recognition and management of patients exposed to biological warfare agents. *JAMA* 278:399–411
- Fureby C, Grinstein F (1999) Monotonically integrated large eddy simulation of free shear flows. *AIAA J* 37(5):544–556
- Grinstein FF, Fureby C (2002) Recent progress on MILES for high-Reynolds-number flows. *J Fluids Eng* 124:848–861
- Gupta JK, Lin C-H, Chen Q (2009) Flow dynamics and characterization of a cough. *Indoor Air* 19:517–525

37. Gupta JK, Lin C-H, Chen Q (2010) Characterizing exhaled airflow from breathing and talking. *Indoor Air* 20:31–39
38. Gupta JK, Lin C-H, Chen Q (2011) Inhalation of expiratory droplets in aircraft cabins. *Indoor Air* 21:341–350. <https://doi.org/10.1111/j.1600-0668.2011.00709.x>
39. Gupta JK, Lin C-H, Chen Q (2011) Transport of expiratory droplets in an aircraft cabin. *Indoor Air* 21:3–11
40. Halloran SK, Wexler AS, Ristenpart WD (2012) A Comprehensive breath plume model for disease transmission via expiratory aerosols. *PLoS One* 7(5):e37088. <https://doi.org/10.1371/journal.pone.0037088>
41. Hintermüller M, Rautenberg CN, Mohammadi M, Kanitsar M (2017) Optimal sensor placement: a robust approach. *SIAM J Control Optim* 55(6):3609–3639. <https://doi.org/10.1137/16M1088867>
42. Idelsohn SR, Nigro N, Larreteguy A, Gimenez JM, Ryshakov P (2019) A pseudo-DNS method for the simulation of incompressible fluid flows with instabilities at different scales. *Int J Comput Part Mech*. <https://doi.org/10.1007/s40571-019-00264-x>
43. Jameson A, Schmidt W, Turkel E (1981) Numerical solution of the Euler equations by finite volume methods using Runge–Kutta time-stepping schemes. *AIAA-81-1259*
44. Johnson GR, Morawska L, Ristovski ZD, Hargreaves M, Mengersen K, Chao CYH, Wan MP, Li Y, Xie X, Katoshevski D, Corbette S (2011) Modality of human expired aerosol size distributions. *J Aerosol Sci* 42:839–851
45. Kampf G, Todt D, Pfaender S, Steinmann E (2020) Persistence of coronaviruses on inanimate surfaces and their inactivation with biocidal agents. *J Hosp Infect* 104(3):246–251. <https://doi.org/10.1016/j.jhin.2020.01.022>
46. Kim Y-I et al (2020) Infection and rapid transmission of SARS-CoV-2 in ferrets. *Cell Host Microbe* 27:1–6. <https://doi.org/10.1016/j.chom.2020.03.023>
47. Li Y et al (2007) Role of ventilation in airborne transmission of infectious agents in the built environment—a multidisciplinary systematic review. *Indoor Air* 17:2–18
48. Lindsley WG, Blachere FM, Thewlis RE, Vishnu A, Davis KA, Cao G et al (2010) Measurements of airborne influenza virus in aerosol particles from human coughs. *PLoS One* 5:e15100
49. Lindsley WG, Pearce TA, Hudnall JB, Davis KA, Davis SM, Fisher MA et al (2012) Quantity and size distribution of cough-generated aerosol particles produced by influenza patients during and after illness. *J Occup Environ Hyg* 9:443–9
50. Liu J, Kailasanath K, Ramamurti R, Munday D, Gutmark E, Löhner R (2009) Large-eddy simulations of a supersonic jet and its near-field acoustic properties. *AIAA J* 47(8):1849–1864
51. Löhner R, Ambrosiano J (1990) A vectorized particle tracer for unstructured grids. *J Comput Phys* 91(1):22–31
52. Löhner R, Ramamurti R (1995) A load balancing algorithm for unstructured grids. *Comput Fluid Dyn* 5:39–58
53. Löhner R (1998) Renumbering strategies for unstructured-grid solvers operating on shared-memory, cache-based parallel machines. *Comput Methods Appl Mech Eng* 163:95–109
54. Löhner R, Yang C, Oñate E (1998) Viscous free surface hydrodynamics using unstructured grids. In: *Proceedings of 22nd symposium on naval hydrodynamics*, Washington, DC
55. Löhner R, Yang C, Cebal J, Soto O, Camelli F, Baum JD, Luo H, Mestreau E, Sharov D, Ramamurti R, Sandberg W, Oh C (2001) Advances in FEFLO. *AIAA-01-0592*
56. Löhner R, Galle M (2002) Minimization of indirect addressing for edge-based field solvers. *Commun Numer Methods Eng* 18:335–343
57. Löhner R (2004) Multistage explicit advective prediction for projection-type incompressible flow solvers. *J Comput Phys* 195:143–152
58. Löhner R, Baum JD, Rice D (2004) Comparison of coarse and fine mesh 3-D Euler predictions for blast loads on generic building configurations. In: *Proceedings of MABS-18 conference*, September, Bad Reichenhall, Germany
59. Löhner R, Camelli F (2005) Optimal placement of sensors for contaminant detection based on detailed 3-D CFD simulations. *Eng Comput* 22(3):260–273
60. Löhner R, Chi Yang JR, Cebal F, Soto Camelli O, Waltz J (2006) Improving the speed and accuracy of projection-type incompressible flow solvers. *Comput Methods Appl Mech Eng* 195(23–24):3087–3109
61. Löhner R, Chi Yang, Oñate E (2006) On the simulation of flows with violent free surface motion. *Comput Methods Appl Mech Eng* 195:5597–5620
62. Löhner R, Chi Yang, Oñate E (2007) Simulation of flows with violent free surface motion and moving objects using unstructured grids. *Int J Numer Methods Fluids* 53:1315–1338
63. Löhner R (2008) *Applied CFD techniques*, 2nd edn. Wiley, Hoboken
64. Löhner R, Luo H, Baum JD, Rice D (2008) Improvements in speed for explicit, transient compressible flow solvers. *Int J Numer Methods Fluids* 56(12):2229–2244
65. Löhner R, Cebal JR, Camelli FF, Appanaboyina S, Baum JD, Mestreau EL, Soto O (2008) Adaptive embedded and immersed unstructured grid techniques. *Comput Methods Appl Mech Eng* 197:2173–2197
66. Löhner R, Mut F, Camelli FF (2011) Timings of FEFLO on the SGI-ICE machines. *AIAA-11-1064*
67. Löhner R, Corrigan A (2011) Semi-automatic porting of a general fortran CFD code to GPUs: the difficult modules. *AIAA-11-3219*
68. Löhner R, Mut F, Cebal JR, Aubry R, Houzeaux G (2011) Deflated preconditioned conjugate gradient solvers for the pressure-Poisson equation: extensions and improvements. *Int J Numer Methods Eng* 87(1–5):2–14
69. Löhner R (2012) F2GPU—a general fortran to GPU translator. In: *Proceedings of NVIDIA GTC conference*, San Jose, CA, May
70. Löhner R, Camelli F, Baum JD, Togashi F, Soto O (2014) On mesh-particle techniques. *Comput Part Mech* 1:199–209
71. Löhner R, Camelli F (2016) Tightly coupled computational fluid and crowd dynamics. In: Song W, Ma J, Fu L (eds) *Proceedings of pedestrian and evacuation dynamics 2016 (PED 2016)*, University of Science and Technology Press, Hefei, China, pp 505–509
72. Loudon RG, Roberts RM (1967) Droplet expulsion from the respiratory tract. *Am Rev Respir Dis* 95(3):435–442
73. Luo H, Baum JD, Löhner R (1994) Edge-based finite element scheme for the Euler equations. *AIAA J* 32(6):1183–1190
74. Luo H, Baum JD, Löhner R, Cabello J (1994) Implicit finite element schemes and boundary conditions for compressible flows on unstructured grids. *AIAA-94-0816*
75. Luo H, Baum JD, Löhner R (1999) An accurate, fast, matrix-free implicit method for computing unsteady flows on unstructured grids. *AIAA-99-0937*
76. Luo H, Sharov D, Baum JD, Löhner R (2000) A class of matrix-free implicit methods for compressible flows on unstructured grids. In: *First international conference on computational fluid dynamics*, July, Kyoto, Japan, pp 10–14
77. Luo H, Baum JD, Löhner R (2001) A fast, matrix-free implicit method for computing low mach number flows on unstructured grids. *Int J Comput Fluid Dyn* 14:133–157
78. Milton DK, Fabian MP, Cowling BJ, Grantham ML, McDevitt JJ (2013) Influenza virus aerosols in human exhaled breath: particle size, culturability, and effect of surgical masks. *PLoS Pathog* 9:e1003205
79. Ramamurti R, Löhner R (1993) Simulation of flow past complex geometries using a parallel implicit incompressible flow solver. In:

- Proceedings of 11th AIAA CFD Conference, July, Orlando, FL, pp 1049–1050
80. Ramamurti R, Löhner R (1996) A parallel implicit incompressible flow solver using unstructured meshes. *Comput Fluids* 5:119–132
 81. Ramamurti R, Sandberg WC, Löhner R (1999) Computation of unsteady flow past deforming geometries. *Int J Comput Fluid Dyn* 13:83–99
 82. Rice DL, Baum JD, Togashi F, Löhner R, Amini A (2008) First-principles blast diffraction simulations on a notebook: accuracy, resolution and turn-around issues. In: Proceedings of MABS-20 conference, September, Oslo, Norway
 83. Schlichting H (1979) *Boundary layer theory*. McGraw-Hill, New York
 84. Sharov D, Luo H, Baum JD, Löhner R (2000) Implementation of unstructured grid GMRES+LU-SGS method on shared-memory, cache-based parallel computers. AIAA-00-0927
 85. Stück A, Camelli F, Löhner R (2010) Adjoint-based design of shock mitigation devices. *Int J Numer Methods Fluids* 64:443–472
 86. Tang JW, Li Y, Eames I, Chan PKS, Ridgway GL (2006) Factors involved in the aerosol transmission of infection and control of ventilation in healthcare premises. *J Hosp Infect* 64:100–114
 87. Tang JW, Noakes CJ, Nielsen PV, Eames I, Nicolle A, Li Y, Settles GS (2011) Observing and quantifying airflows in the infection control of aerosol- and airborne-transmitted diseases: an overview of approaches. *J Hosp Infect* 77:213–222
 88. Tang JW, Nicolle AD, Pantelic J, Koh GC, Wang L, Amin M, Klettner CA, Cheong DKW, Sekhar C, Tham KW (2012) Airflow dynamics of coughing in healthy human volunteers by shadow-graph imaging: an aid to aerosol infection control. *PLoS One* 7(4):e34818. <https://doi.org/10.1371/journal.pone.0034818>
 89. Tang JW, Nicolle AD, Klettner CA, Pantelic J, Wang L, Bin Suhaimi A, Tan AYL, Ong GWX, Su R, Sekhar C, Cheong DKW, Tham KW (2013) Airflow dynamics of human jets: sneezing and breathing-potential sources of infectious aerosols. *PLoS One* 8(4):e59970. <https://doi.org/10.1371/journal.pone.0059970>
 90. Teunis PFM, Brienens N, Kretzschmar MEE (2010) High infectivity and pathogenicity of influenza A virus via aerosol and droplet transmission. *Epidemics* 2:215–222
 91. Tilch R, Tabbal A, Zhu M, Decker F, Löhner R (2008) Combination of body-fitted and embedded grids for external vehicle aerodynamics. *Eng Comput* 25(1):28–41
 92. To KK-W et al (2020) Temporal profiles of viral load in posterior oropharyngeal saliva samples and serum antibody responses during infection by SARS-CoV-2: an observational cohort study. *Lancet Infect Dis*. [https://doi.org/10.1016/S1473-3099\(20\)30196-1](https://doi.org/10.1016/S1473-3099(20)30196-1)
 93. Togashi F, Baum JD, Mestreau E, Löhner R, Sunshine D (2009) Numerical modeling of long-duration blast wave evolution in confined facilities. AIAA-09-1531
 94. van Doremalen N, Bushmaker T, Morris DH, Holbrook MG, Gamble A, Williamson BN, Tamin A, Harcourt JL, Thornburg NJ, Gerber SI, Lloyd-Smith JO, de Wit E, Munster VJ (2020) Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1. *N Engl J Med*. <https://doi.org/10.1056/NEJMc2004973>
 95. Wei J, Li Y (2016) Airborne spread of infectious agents in the indoor environment. *Am J Infect Control* 44:S102–S108. <https://doi.org/10.1016/j.ajic.2016.06.003>
 96. Xie X, Li Y, Chwang ATY, Ho PL, Seto WH (2007) How far droplets can move in indoor environments—revisiting the wells evaporation-falling curve. *Indoor Air* 17:211–225. <https://doi.org/10.1111/j.1600-0668.2006.00469.x>
 97. Zhang T, Chen Q, Lin C-H (2007) Optimal sensor placement for airborne contaminant detection in an aircraft cabin. *HVAC&R Res* 13(5):683–696
 98. Zhang Y, Feng G, Kang Z, Bi Y, Cai Y (2017) Numerical simulation of coughed droplets in conference room. In: 10th international symposium on heating, ventilation and air conditioning, ISHVAC2017, October, 19–22 Jinan, China. *Procedia Eng* 205:302–308

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.