



A Digital-Twin and Machine-Learning Framework for Ventilation System Optimization for Capturing Infectious Disease Respiratory Emissions

T. I. Zohdi¹

Received: 3 March 2021 / Accepted: 22 May 2021
© CIMNE, Barcelona, Spain 2021

Abstract

The pandemic of 2019 has led to an enormous interest in all aspects of modeling and simulation of infectious diseases. One central issue is the redesign and deployment of ventilation systems to mitigate the transmission of infectious diseases, produced by respiratory emissions such as coughs. This work seeks to develop a combined Digital-Twin and Machine-Learning framework to optimize ventilation systems by building on rapidly computable respiratory emission models developed in Zohdi (Comput Mech 64:1025–1034, 2020). This framework ascertains the placement and flow rates of multiple ventilation units, in order to optimally sequester particles released from respiratory emissions such as coughs, sneezes, etc. Numerical examples are provided to illustrate the framework.

1 Introduction

The pandemic of 2019, due to SARS-CoV-2, named COVID-19 and referred to as coronavirus, has been responsible for nearly two million of deaths worldwide in 2020 alone. It is well-established that this virus primarily spreads from person-to-person contact by respiratory droplets produced when an infected person coughs or sneezes. Subsequently, the droplets come into contact with the eyes, nose or mouth of a nearby person or when a person touches an infected surface, then makes contact with their eyes, nose or mouth. Since the virus is small, 0.06–0.14 microns in diameter, it can be contained in or attached to such emitted droplets. Droplets as small as one micron can carry enough viral load to cause an infection. A particular concern is the interaction of droplets with ventilation systems, which can capture emitted pathogens, but can also potentially could enhance their propagation. This has implications on situation-specific safe distancing and the design of building filtration systems, air distribution, heating, air-conditioning and decontamination systems, for example using UV-c and related technologies.

In order to facilitate such system redesigns, fundamental analysis tools are needed that are easy to use. Accordingly, this work develops a combined Digital-Twin and Machine-Learning framework to optimize ventilation systems by building on rapidly computable respiratory emission models developed in Zohdi [1]. This framework ascertains the placement and flow rates of multiple mobile ventilation units, in order to optimally sequester particles released from respiratory emissions such as coughs, sneezes, etc. There have been dramatic advances in technologies associated with automation across many industries, which have the potential to drastically improve system efficiency, quality and safety. Some approaches are methodical and systematic, while some are ad-hoc and haphazard. In the world of systems engineering, increasingly sophisticated and integrated approaches for digital systems are appearing at a rapid rate. One key component to advancements in this areas are Digital-Twins, which are digital replicas of complex physical systems that can be safely manipulated and optimized in a virtual world and deployed in the physical world afterwards, reducing costs of experiments and accelerating development of new technologies. Digital-Twins blend AI, machine learning, software analytics and data to create living digital computer models that can update and change in tandem with their physical counterparts-and ultimately control them in real time. Updates to the Digital-Twin are made continuously in real-time, which necessitates technologies like rapid wireless communication, hyperspectral cameras, sensor fusion

✉ T. I. Zohdi
zohdi@me.berkeley.edu

¹ Department of Mechanical Engineering 6195 Etcheverry Hall, University of California, Berkeley, CA 94720-1740, USA

and fast simulations of process behavior. Today, there is no shortage of general simulation software; however, the fundamental limitations are ease-of-use and integration of models with in-field data and easy deployment for researchers. A core issue across all domains is the ability of a system to adapt to rapid changes in the environment and system capabilities by autonomously modifying operation-with humans in the loop. By developing these tools for food systems researchers, we believe that we can enhance the ability of industry to conduct research that inevitably involve Digital Twins, as well as benefiting society and industry.

The presented work also involves blending Machine-learning with Digital Twins, whereby simulations iteratively learn from their mistakes and constantly evolve to improve and optimize the model, safely in a virtual environment. This rapid simulation should be designed to run rapidly in tandem with a real system. Accordingly, the Digital-Twin and Machine-Learning in this framework combines these rapidly computable models with genomic Machine-Learning methods to ascertain the placement and flow rates of multiple mobile ventilation units, in order to optimally sequester particles released from respiratory emissions such as coughs, sneezes, etc. Specifically:

- A model is developed for the propagation of emitted particles that is rapidly computable,
- A qualitative analysis of the model is discussed,
- A numerical discretization for the flow of the distribution of particles is constructed,
- A genomic-based Machine-Learning algorithm to optimize the mobile ventilation systems is developed and
- Numerical examples are provided.

Remark 1 It is now widely accepted that masks are an integral part of controlling the spread of infectious diseases, such as COVID19. There are three general categories of face masks: (1) Cloth Masks (for the general public), which provides some protection to the wearer, but mainly serves to stop the spread of viruses (2) Surgical Masks (for healthcare workers and the general public), which provides partial protection and (3) N95 Respirators (primarily for healthcare workers), which provides high level of protection and use charged fibers (electrets) to attract particles of all sizes. N95 masks are excellent at trapping very small particles (below 0.1 micron), since their motion is random (Brownian), due to collisions with other particles, air molecules, etc., which will force them to collide with an electrically charged “sticky” fiber. It is also excellent in trapping large particles (above 0.3 microns), since their motion is effectively linear across the mask (because of their inertia) and will collide with a fiber. It has more difficulty trapping medium-sized particles (0.1 microns-0.3 microns), which travel with airflow

patterns. N95 means that it traps at least 95% of the particles. However, there are problems, due to mask ill-fitting, re-use due to shortages, use of alcohols and solvents for mask cleaning, etc., which may damage the mask or neutralize the fiber charge. Furthermore, the proper donning and doffing (removal from face) of a mask is nontrivial, due to strap elongation, facial hair, etc. Therefore, one must assume that some particles get through, regardless of the mask type. Accordingly, in workplaces, some type ventilation is needed, generally involving HEPA (High Efficiency Particulate Absorbing) grade filters, which remove 99.95% of the particles equal to 0.3 microns in size, with efficiency increasing for particles larger or smaller than this size. Such filters were started by the German military in WWII for gas masks, whereby increases in mask efficiency were noted by simply adding a piece of paper (cellulose) to the mask. Such systems started commercially in the 1950s for decontamination control in hospitals, nuclear facilities, homes vehicles, etc. They are comprised of randomly arranged fiberglass fibers between 0.5 and 2.0 microns. As with N95 masks, they are designed to capture a range of particles by similar mechanisms. Another common term is ‘MERV’ Minimum Efficiency Reporting Value (1987 ASHRAE), which is designed to report the worst case performance of a filter, ranging 0.3–10 microns, with a scale rating of from 1–16 (highest at 95%). It is essential, that ventilation systems be given much more attention than ever before.

Remark 2 We remark that some systems also employ UV-c light for further decontamination, in combination with the ventilation system. A dosage of 1 J/cm^2 is considered adequate to destroy COVID19. Many commercial mobile UV-c light systems now exist. UV-c light 254–265 nm is considered optimal. Wavelengths above this range are not effective, while wavelengths below produce ozone, which is dangerous (Zohdi [2]). Heating contaminated at 70 °C for 60 min is effective, but impractical in a large workplace setting with sensitive equipment and people. For example, decontamination based on UV technology has become ubiquitous, with many variants now being proposed. UV light varies in wavelength from 10 to 400 nm, thus making it shorter than visible wavelengths and larger than X-rays. Short wave UV light (UV-c) can damage DNA and sterilize surfaces making it useful in the medical industry. This was first noted in 1878 (Downes and Blunt [3]) when the effect of short-wavelength light killing bacteria was discovered. By 1903 it was known the most effective wavelengths were around 250 nm (UV-c), for which Niels Finsen won a Nobel Prize (for skin-based tuberculosis eradication using UV light). Contaminants in the indoor environment are almost entirely organic carbon-based compounds, which break down when exposed to high-intensity UV at 240–280 nm. Despite the attractiveness of using UV-c light, the literature has shown

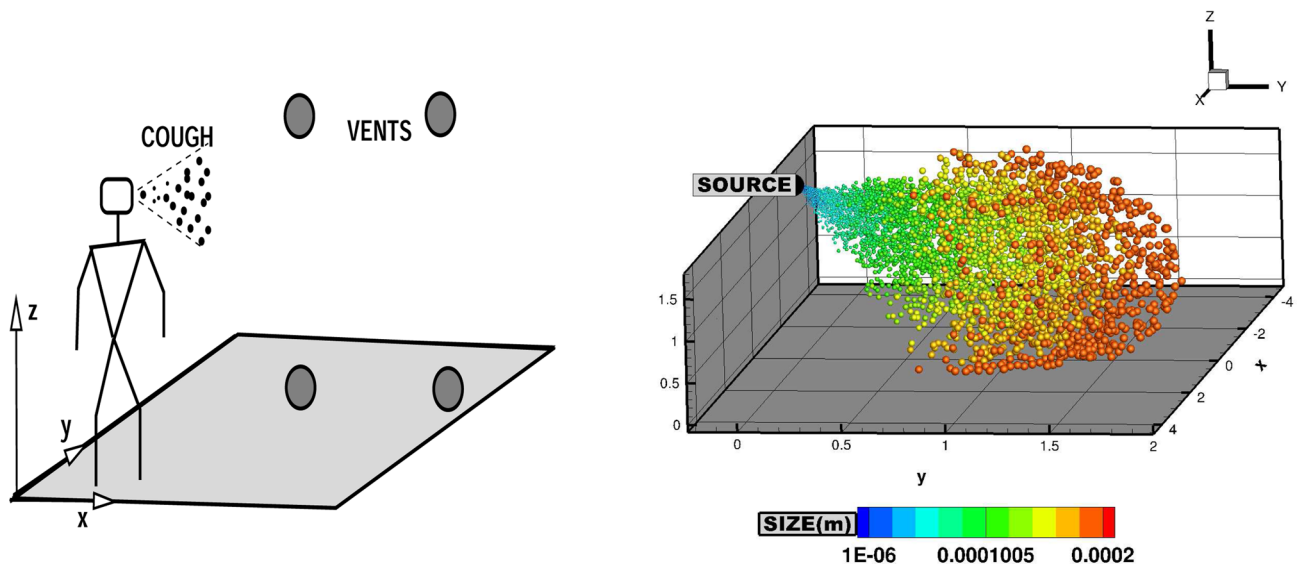


Fig. 1 Left: The model problem studied in this work with hypothetical ventilation unit locations (which will be optimized in this work). Right: Zoom: The release of cough particles-color coded by size (Zohdi [1])

that it is difficult to ensure that all surfaces are completely decontaminated due to shadowing effects. Thus, the use of ultraviolet germicidal irradiation (UVGI) is effective only as a *component* in a multistage process—it alone carries the risk of residual contamination. Thus, purely UV-c protocols should be adopted if there is no other choice. However, they can be an integral part of a multistage process involving a combination of (a) gas vapors and (b) heat and humidity. The topic of decontamination technologies is of paramount interest (see references Anderson et al. [4], Battelle [5], Boyce et al. [6], Card et al. [7], Heimbuch and Harish [8], Heimbuch et al. [9], Ito and Ito [10], Lin et al. [11], Kanemitsu [12], Lindsley et al. [13], Lore et al. [14], Marra et al. [15], Mills et al. [16] and Nerandzic et al. [17]), and the corresponding simulation of such processes has recently been undertaken in Zohdi [1] and is a topic of ongoing research.

2 Reduced Order Particle-Fluid Interaction Model

In its most basic form, a cough can be considered as a high-velocity release of a random distribution of particles of various sizes, into an ambient atmosphere (Fig. 1). We refer the reader to Wei and Li [18], Duguid [19], Papineni and Rosenthal [20], Wei and Li [21], Zhu et al. [22], Chao et al. [23], Morawska et al. [24], VanSciver et al. [25], Kwon et al. [26], Tang et al. [27], Xie et al. [28], Gupta et al. [29], Wan et al. [30], Villafruela et al. [31], Nielson [32] Zhang and Li [33], Lindsley et al. [34] and Burridge

et al. [35] for extensive reviews of coughs and other respiratory emissions. We will consider the model problem in Fig. 1.

2.1 Assumptions

Following formulations for physically similar problems associated with particulate dynamics from the fields of blasts, explosions and fire embers (Zohdi [36–39]), we make the following assumptions (Zohdi [1]):

- We assume the same initial velocity magnitude for all particles under consideration, with a random distribution of outward directions away from the source of the cough. This implies that a particle non-interaction approximation is appropriate. Thus, the inter-particle collisions are negligible. This has been repeatedly verified by “brute-force” collision calculations using formulations found in Zohdi [40–43].
- We assume that the particles are spherical with a random distribution of radii R_i , $i = 1, 2, 3 \dots P_n$, where P_n is the total number of particles. The masses are given by $m_i = \rho_i \frac{4}{3} \pi R_i^3$, where ρ_i is the density of the particles.
- We assume that the cough particles are *quite small* and that the amount of rotation, if any, contributes negligibly to the overall trajectory of the particles. The equation of motion for the i th particle in the system is

$$m_i \dot{\mathbf{v}}_i = \Psi_i^{\text{grav}} + \Psi_i^{\text{drag}} + \Psi_i^{\text{mask}}, \quad (2.1)$$

with initial velocity $\mathbf{v}_i(0)$ and initial position $\mathbf{r}_i(0)$. The gravitational force is $\Psi_i^{grav} = m_i \mathbf{g}$, where $\mathbf{g} = (g_x, g_y, g_z) = (0, 0, -9.81) \text{ m/s}^2$.

- For the drag, we will employ a general phenomenological model

$$\Psi_i^{drag} = \frac{1}{2} \rho_a C_D ||\mathbf{v}^f - \mathbf{v}_i|| (\mathbf{v}^f - \mathbf{v}_i) A_i, \quad (2.2)$$

where C_D is the drag coefficient, A_i is the reference area, which for a sphere is $A_i = \pi R_i^2$, ρ_a is the density of the ambient fluid environment and \mathbf{v}^f is the velocity of the surrounding medium which, in the case of interest, is air. We will assume that the velocity of the surrounding fluid medium (\mathbf{v}^f) is given, implicitly assuming that the dynamics of the surrounding medium are unaffected by the particles. We will discuss these assumptions further, later in the work.

2.2 Qualitative Behavior of the Simplified Model

A simplified Stokesian model, which can be solved analytically provides some physical insight, is discussed in Zohdi [1]. A summary of that analysis is as follows. For a (low Reynolds number) Stokesian model, the differential equation for each particle is

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \mathbf{g} + c_i (\mathbf{v}^f - \mathbf{v}_i) \quad (2.3)$$

where $c_i = \mu_f 6\pi R_i$, where μ_f is the viscosity of the surrounding fluid (air) and the local Reynolds number for a particle is $Re = \frac{\rho_a ||\mathbf{v}^f - \mathbf{v}_i||}{\mu_f}$ and μ_f is the fluid viscosity. The trends are

- As $R_i \rightarrow 0$, then $\mathbf{v}_i(\infty) \rightarrow \mathbf{v}^f$,
- As $\mathbf{v}_i^f \rightarrow \mathbf{0}$, then $\mathbf{v}_i(\infty) \rightarrow \frac{2\rho R_i^2}{9\mu_f} \mathbf{g}$.

In summary

- Large particles travel far and settle quickly and
- Small particles do not travel far and settle slowly.

Remark 3 The ratio of the Stokesian drag force to gravity is

$$\frac{||\Psi_i^{drag, Stokesian}||}{||\Psi_i^{grav}||} = \frac{9\mu_f ||\mathbf{v}^f - \mathbf{v}_i||}{2\rho_i R_i^2 g}, \quad (2.4)$$

which indicates that for very small particles, drag will dominate the settling process and for larger particles, gravity will dominate.

3 Simulation of the Model: Cough Particle Generation and Propagation

For the general model, in order to more accurately model the effects of drag, one can take into account that the empirical drag coefficient varies with Reynolds number. For example, consider the following piecewise relation (Chow [23]):

- For $0 < Re \leq 1$, $C_D = \frac{24}{Re}$,
- For $1 < Re \leq 400$, $C_D = \frac{24}{Re^{0.646}}$,
- For $400 < Re \leq 3 \times 10^5$, $C_D = 0.5$,
- For $3 \times 10^5 < Re \leq 2 \times 10^6$, $C_D = 0.000366 Re^{0.4275}$ and
- For $2 \times 10^6 < Re < \infty$, $C_D = 0.18$,

where, as in the previous section, the local Reynolds number for a particle is $Re = \frac{\rho_a ||\mathbf{v}^f - \mathbf{v}_i||}{\mu_f}$ and μ_f is the fluid viscosity.¹

We note that in the zero Reynolds number limit, the drag is Stokesian. In order to solve the governing equation,

$$\begin{aligned} m_i \dot{\mathbf{v}}_i &= \Psi_i^{grav} + \Psi_i^{drag} + \Psi_i^{mask} \\ &= m_i \mathbf{g} + \frac{1}{2} \rho_a C_D ||\mathbf{v}^f - \mathbf{v}_i|| (\mathbf{v}^f - \mathbf{v}_i) A_i + \Psi_i^{mask}. \end{aligned} \quad (3.1)$$

3.1 An Example

To illustrate the model, we utilize an example from Zohdi [1], without a mask.

As an example, a mean particle radius was chosen to be $\bar{R} = 0.0001 \text{ m}$ with variations according to

$$R_i = \bar{R} \times (1 + A \times \zeta_i), \quad (3.2)$$

where $A = 0.9975$ and a random variable $-1 \leq \zeta_i \leq 1$. The algorithm used for particle generation was:

- $M = 0$
- Start loop: $i = 1, P_n$
- $R_i = \bar{R} \times (1 + A \times \zeta_i)$,
- $M = M + m_i = M + \rho_i \frac{4}{3} \pi R_i^3$
- If $M \geq M^{Total}$ then stop (determines $P_n = \text{particles}$)
- End loop

The initial trajectories were determined from the following algorithm

- Specify relative direction 'cone' parameters: $\mathbf{N}^c = (N_x^c, N_y^c, N_z^c)$,
- For each particle, $i = 1, 2, 3, \dots, P_n$, construct a (perturbed) trajectory vector:

¹ The viscosity coefficient for air is $\mu_f = 0.000018 \text{ Pa-s}$.

$$\begin{aligned} N_i &= (N_x^c + A_x^c \times \eta_{ix}, N_y^c + A_y^c \times \eta_{iy}, N_z^c + A_z^c \times \eta_{iz}) \\ &= (N_{ix}, N_{iy}, N_{iz}), \end{aligned} \quad (3.3)$$

where $-1 \leq \eta_{ix} \leq 1$, $0 \leq \eta_{iy} \leq 1$ and $-1 \leq \eta_{iz} \leq 1$.

- For each particle, normalize the trajectory vector:

$$\mathbf{n}_i = \frac{1}{\|\mathbf{N}_i\|} (N_{ix}, N_{iy}, N_{iz}). \quad (3.4)$$

- For each particle, the velocity vector is constructed by a projection onto the normal vector:

$$\mathbf{v}_i = V^c \mathbf{n}_i. \quad (3.5)$$

Following Zohdi [1], we integrate the velocity numerically

$$\begin{aligned} \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \frac{1}{m_i} \int_t^{t+\Delta t} (\Psi_i^{\text{grav}} + \Psi_i^{\text{drag}}) dt \approx \mathbf{v}_i(t) \\ &+ \frac{\Delta t}{m_i} (\Psi_i^{\text{grav}}(t) + \Psi_i^{\text{drag}}(t)). \end{aligned} \quad (3.6)$$

The position is the obtained by integrating again:

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \int_t^{t+\Delta t} \mathbf{v}_i(t) dt \approx \mathbf{r}_i(t) + \Delta t \mathbf{v}_i(t). \quad (3.7)$$

This approach has been used repeatedly for a variety of physically similar drift-type problems in Zohdi [36–39]. In order to illustrate the model, the following simulation parameters were chosen:

- Starting height of 2 meters,
- Total simulation duration, 4 s,
- The time step size, $\Delta t = 10^{-6}$ s,
- The cough velocity, $V^c(t = 0) = 30 \text{ m/s}$ (taken from the literature which indicates $10 \text{ m/s} \leq V^c \leq 50 \text{ m/s}$),
- Density of particles, $\rho_i = 1000 \text{ kg/m}^3$,
- Density of air, $\rho_a = 1.225 \text{ kg/m}^3$ and
- Total mass, $M^{\text{Total}} = \sum_{i=1}^{P_n} m_i = 0.0005 \text{ kg}$.

An extremely small (relative to the total simulation time) time-step size of $\Delta t = 10^{-6}$ seconds was used. Further reductions of the time-step size produced no noticeable changes in the results, thus the solutions generated can be considered to have negligible numerical error. The simulations took under 10 seconds on a standard laptop. The algorithm generated 59941 particles ranging from $2.5 \times 10^{-7} \text{ m} \leq R_i \leq 2 \times 10^{-4} \text{ m}$ (i.e. $0.25 \text{ microns} \leq R_i \leq 200 \text{ microns}$). We used a trajectory cone of $N^c = (0, 1, 0)$ and $A^c = (1, 0.5, 1)$ in the example given. Figure 2 illustrates the results for the parameters above (for $v_y^f = 0$). We refer the reader to Zohdi [1] for more details.

3.2 Model Enhancements

The mask model will include an extra term in the drag resistance for particles in the domain of the mask ahead of the face:

$$\Psi_i^{\text{mask}} = -\mathcal{R} \mathbf{v}_i A_i^s, \quad (3.8)$$

where $A^s = 4\pi R^2$. To account for multiple interacting vents assume a radial distance decay model for flow field from each vent, $j = 1, 2, \dots, N$ of the form:

$$\mathbf{v}_j^f = v_j^o e^{-d\|\mathbf{r}_j^v - \mathbf{r}_i\|} \frac{(\mathbf{r}_i^v - \mathbf{r}_i)}{\|\mathbf{r}_i^v - \mathbf{r}_i\|}, \quad (3.9)$$

where \mathbf{r}_j^v is the position of the j th vent and d is a decay factor. Figure 3 illustrates the flow patterns with this model for 4 randomly placed vents. These will be optimized later in the work. In summary, the overall model is:

$$\begin{aligned} m_i \dot{\mathbf{v}}_i &= \Psi_i^{\text{grav}} + \Psi_i^{\text{drag}} + \Psi_i^{\text{mask}} \\ &= m_i \mathbf{g} + \frac{1}{2} \rho_a C_D \|\mathbf{v}^f - \mathbf{v}_i\| (\mathbf{v}^f - \mathbf{v}_i) A_i - \mathcal{R} \mathbf{v}_i A_i^s. \end{aligned} \quad (3.10)$$

Remark 4 A fully resolved two-way coupled system of particles and surrounding fluid would require a fine spatio-temporal discretization involving Finite Element, Finite Difference or Finite Volume methods of the Navier-Stokes equations for the surrounding fluid

$$\text{Balance of mass : } \frac{\partial \rho}{\partial t} = -\nabla_x \rho \cdot \mathbf{v} - \rho \nabla_x \cdot \mathbf{v},$$

$$\text{Balance of momentum : } \rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\nabla_x \mathbf{v}) \cdot \mathbf{v} \right) = \nabla_x \cdot \boldsymbol{\sigma} + \mathbf{f},$$

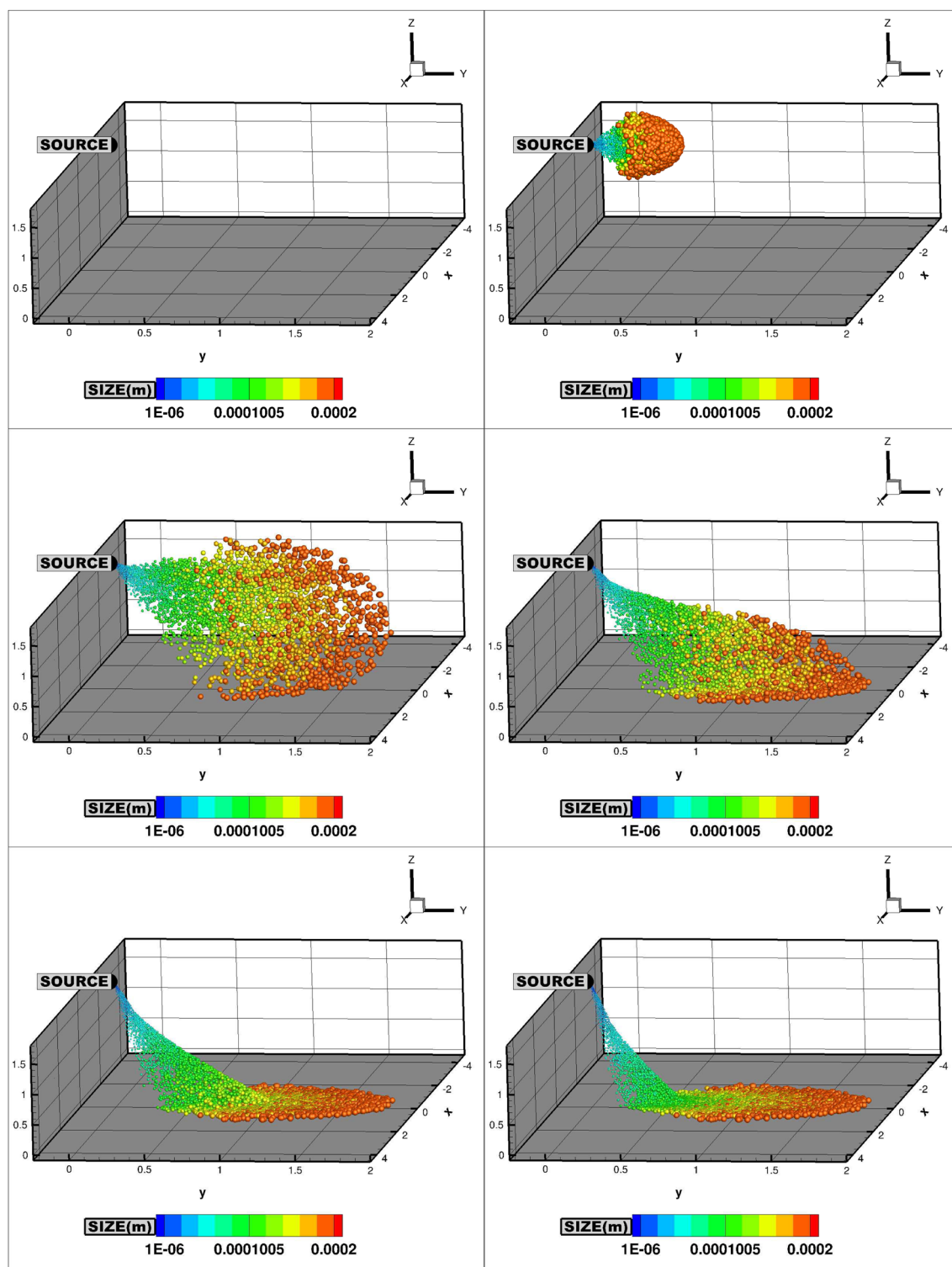
$$\text{Constitutive Law : } \boldsymbol{\sigma} = -P \mathbf{1} + \lambda \text{tr} \mathbf{D} \mathbf{1} + 2\mu \mathbf{D}$$

$$= -P \mathbf{1} + 3\kappa \frac{\text{tr} \mathbf{D}}{3} \mathbf{1} + 2\mu \mathbf{D}', \quad (3.11)$$

where $\rho(\mathbf{x})$ is the density field of the fluid, $\mathbf{v}(\mathbf{x})$ is the fluid velocity field, $\boldsymbol{\sigma}(\mathbf{x})$ is the fluid stress field, $\mathbf{D}(\mathbf{x})$ is the fluid velocity gradient field, $\mathbf{f}(\mathbf{x})$ is the body force field, $P(\mathbf{x})$ is the fluid pressure field, $\lambda(\mathbf{x})$ and $\mu(\mathbf{x})$ are fluid material property fields.² Additionally, the first law of thermodynamics could be included (along with equations for various chemical reactions), which reads as

$$\rho \dot{w} - \boldsymbol{\sigma} : \nabla_x \mathbf{v} + \nabla_x \cdot \mathbf{q} - \rho z = 0, \quad (3.12)$$

² It is customary to specify \mathbf{v} and P on the boundary, and to determine ρ on the boundary through the Equation of State. P is given by an Equation of State.



Article for Special Issue

Fig. 2 Cough simulation (from a starting height of 2 meters, for $\mathbf{v}^f = (0, 0, 0)$): successive frames indicating the spread of particles. **a** Large particles travel far and settle quickly and **b** Small particles do not travel far and settle slowly (Zohdi [1])

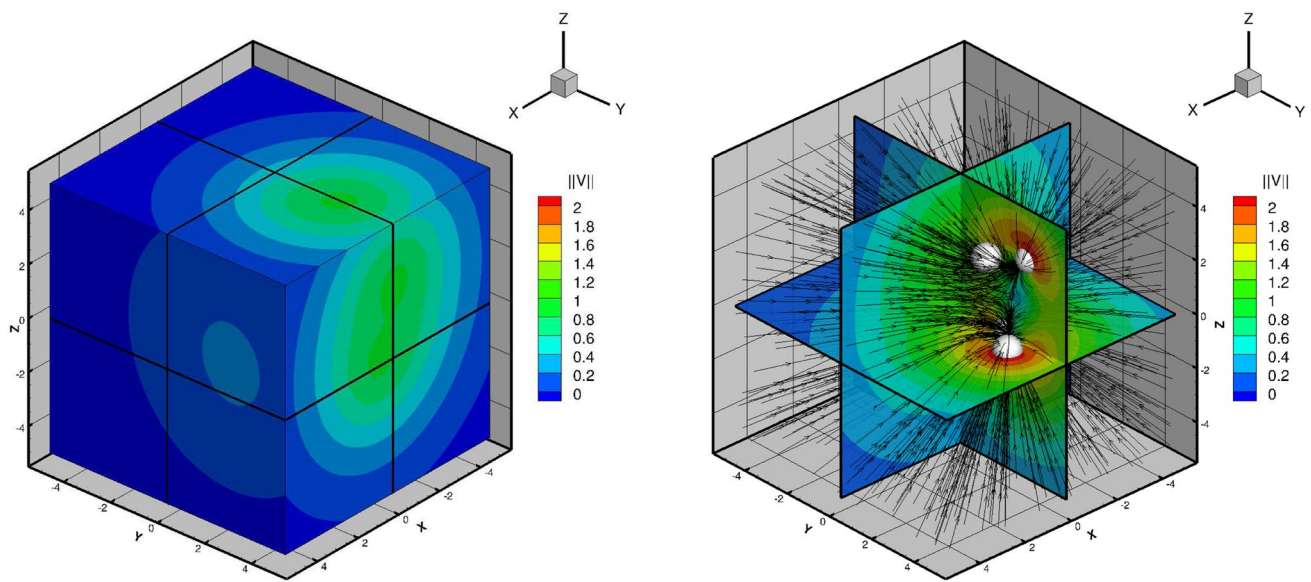
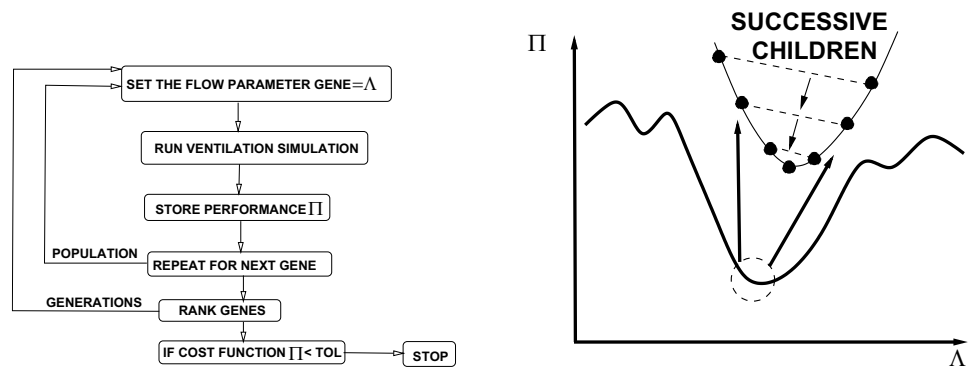


Fig. 3 Flow patterns with the model. Left: flow velocity magnitude. Right: flow velocity magnitude with streamlines

Fig. 4 The basic action of a MLA/GA-Machine Learning Algorithm/Genetic Algorithm. Zohdi [45–47, 39]



where $w(x)$ is the stored energy in the fluid, $q(x)$ is the heat flux field, z is the heat source field per unit mass. This is discussed further at the end of this work.

4 Genomic Machine-Learning Ventilation Optimization

The rapid rate at which these simulations can be completed enables the ability to explore inverse problems seeking to determine what parameter combinations can deliver a desired result (Fig. 4). In order to cast the objective mathematically, we set the problem up as a Machine Learning Algorithm (MLA); specifically a Genetic Algorithm (GA) variant, which is well-suited for nonconvex optimization. Following Zohdi [44–47, 39], we formulate the objective as a cost function minimization problem that seeks system parameters that match a desired response

$$\Pi(\Lambda_1, \dots, \Lambda_{16}) = \frac{\text{PARTICLES REMAINING}}{\text{TOTAL PARTICLES RELEASED}}. \quad (4.1)$$

We systematically minimize Eq. 4.1, $\min_{\Lambda} \Pi$, by varying the design parameters: $\Lambda_{\text{def}}^i = \{\Lambda_1^i, \Lambda_2^i, \Lambda_3^i, \dots, \Lambda_N^i\} = \{\text{panel size, spacing, angles} \dots\}$. The system parameter search is conducted within the constrained ranges of $\Lambda_1^{(-)} \leq \Lambda_1 \leq \Lambda_1^{(+)}$, $\Lambda_2^{(-)} \leq \Lambda_2 \leq \Lambda_2^{(+)}$ and $\Lambda_3^{(-)} \leq \Lambda_3 \leq \Lambda_3^{(+)}$, etc. These upper and lower limits would, in general, be dictated by what is physically feasible.

4.1 System Parameter Search: Machine Learning Algorithm (MLA)

Here we follow Zohdi [45–47, 39] in order to minimize Eq. 4.1, which we will refer to as a “cost function”. Cost functions such as Eq. 4.1 are nonconvex in design parameter space and often nonsmooth. Their minimization is usually difficult with direct application of gradient methods. This

motivates derivative-free search methods, for example those found in Machine Learning Algorithms (MLA's). One of the most basic subsets of MLA's are so-called Genetic Algorithms (GA's). Typically, one will use a GA first in order to isolate multiple local minima, and then use a gradient-based algorithm in these locally convex regions or reset the GA to concentrate its search over these more constrained regions. GA's are typically the simplest scheme to start the analysis, and one can, of course, use more sophisticated methods if warranted. For a review of GA's, see the pioneering work of John Holland ([48, 49]), as well as Goldberg [50], Davis [51], Onwubiko [52] and Goldberg and Deb [53].

4.2 Generalities

The MLA/GA approach is extremely well-suited for nonconvex, nonsmooth, multicomponent, multistage systems and, broadly speaking, involves the following essential concepts:

1. **Population generation:** Generate a parameter population of genetic strings: Λ^i
2. **Performance evaluation:** Compute performance of each genetic string: $\Pi(\Lambda^i)$
3. **Rank strings:** Rank them $\Lambda^i, i = 1, \dots, S$
4. **Mating process:** Mate pairs/produce offspring
5. **Gene elimination:** Eliminate poorly performing genetic strings
6. **Population regeneration:** Repeat process with updated gene pool and new *random* genetic strings
7. **Solution post-processing:** Employ gradient-based methods afterwards in local "valleys"-if *smooth enough*

4.2.1 Specifics

Following Zohdi [54, 45–47], the algorithm is as follows:

- **Step 1:** Randomly generate a population of S starting genetic strings, $\Lambda^i, (i = 1, 2, 3, \dots, S)$:

$$\Lambda^i \stackrel{\text{def}}{=} \begin{Bmatrix} \Lambda_1^i \\ \Lambda_2^i \\ \Lambda_3^i \\ \dots \\ \Lambda_N^i \end{Bmatrix} \quad (4.2)$$

- **Step 2:** Compute fitness of each string $\Pi(\Lambda^i), (i = 1, \dots, S)$
- **Step 3:** Rank genetic strings: $\Lambda^i, (i = 1, \dots, S)$
- **Step 4:** Mate nearest pairs and produce two offspring, $(i=1, \dots, S)$:

$$\lambda^i \stackrel{\text{def}}{=} \Phi \circ \Lambda^i + (1 - \Phi) \circ \Lambda^{i+1} \stackrel{\text{def}}{=} \begin{Bmatrix} \phi_1 \Lambda_1^i \\ \phi_2 \Lambda_2^i \\ \phi_3 \Lambda_3^i \\ \dots \\ \phi_N \Lambda_N^i \end{Bmatrix} + \begin{Bmatrix} (1 - \phi_1) \Lambda_1^{i+1} \\ (1 - \phi_2) \Lambda_2^{i+1} \\ (1 - \phi_3) \Lambda_3^{i+1} \\ \dots \\ (1 - \phi_N) \Lambda_N^{i+1} \end{Bmatrix} \quad (4.3)$$

and

$$\lambda^{i+1} \stackrel{\text{def}}{=} \Gamma \circ \Lambda^i + (1 - \Gamma) \circ \Lambda^{i+1} \stackrel{\text{def}}{=} \begin{Bmatrix} \gamma_1 \Lambda_1^i \\ \gamma_2 \Lambda_2^i \\ \gamma_3 \Lambda_3^i \\ \dots \\ \gamma_N \Lambda_N^i \end{Bmatrix} + \begin{Bmatrix} (1 - \gamma_1) \Lambda_1^{i+1} \\ (1 - \gamma_2) \Lambda_2^{i+1} \\ (1 - \gamma_3) \Lambda_3^{i+1} \\ \dots \\ (1 - \gamma_N) \Lambda_N^{i+1} \end{Bmatrix} \quad (4.4)$$

where for this operation, the ϕ_i and γ_i are random numbers, such that $0 \leq \phi_i \leq 1, 0 \leq \gamma_i \leq 1$, which are different for each component of each genetic string

- **Step 5:** Eliminate the bottom M strings and keep top K parents and their K offspring (K offspring+ K parents+ $M=S$)
- **Step 6:** Repeat STEPS 1-6 with top gene pool (K offspring and K parents), plus M new, randomly generated, strings
- **Important option:** Rescale and restart search around best performing parameter set every few generations

Remark 5 If one selects the mating parameter Φ to be greater than one and/or less than zero, one can induce "mutations", i.e. characteristics that neither parent possesses. However, this is somewhat redundant with introduction of new random members of the population in the current algorithm.

Remark 6 If one does not retain the parents in the algorithm above, it is possible that inferior performing offspring may replace superior parents. Thus, top parents should be kept for the next generation. This guarantees a monotone reduction in the cost function. Furthermore, retained parents do not need to be reevaluated, making the algorithm less computationally expensive, since these parameter sets do not have to be reevaluated (or ranked) in the next generation. Numerous studies of the author have shown that the advantages of

Table 1 The top system parameter performers ($\Lambda_1 - \Lambda_{16}$) and the corresponding cost function for various overall time limits

T^{lim}	Λ_1	Λ_2	Λ_3	Λ_4	Λ_5	Λ_6	Λ_7	Λ_8	Λ_9	Λ_{10}	Λ_{11}	Λ_{12}	Λ_{13}	Λ_{14}	Λ_{15}	Λ_{16}	Π
0.175	0.514	2.194	2.661	0.781	-0.286	3.419	2.411	1.369	-0.740	2.368	0.245	0.671	-3.543	2.057	0.689	1.485	0.365
2.000	0.728	2.184	2.712	0.746	-0.199	3.426	2.383	1.275	-0.772	2.434	0.228	0.731	-1.047	2.018	0.657	1.485	0.125
2.225	0.514	2.343	2.931	0.765	-0.263	3.307	2.429	1.205	-0.752	2.466	0.228	0.736	-3.903	2.057	0.669	1.494	0.025
2.500	0.491	2.219	2.800	0.832	-0.313	3.296	2.488	1.345	-0.718	2.343	0.233	0.722	-2.824	2.039	0.656	1.482	0.000

parent retention outweighs inbreeding, for sufficiently large population sizes. Finally, we remark that this algorithm is easily parallelizable.

Remark 7 After application of such a global search algorithm, one can apply a gradient-based method, if the cost function is sufficiently smooth in that region of the parameter space. In other words, if one has located a convex portion of the parameter space with a global genetic search, one can employ gradient-based procedures locally to minimize the objective function further, since they are generally much more efficient for convex optimization of smooth functions. An exhaustive review of these methods can be found in the texts of Luenberger [55] and Gill, Murray and Wright [56].

5 Model Problem

The system was optimized, Eq. 4.1, varying the following (16) parameters:

- Ventilation locations: 4 vents, each with 3 (x_1, x_2, x_3) locations = 12 variables and
- Ventilation flow rates : 4 vents, each with 1 flow rate = 4 variables.

Specifically, the system optimized varying 16 parameters: 4 vents each with 3 (x_1, x_2, x_3) locations, each with 1 flow rate:

- Flow rate of vent 1: $v_1^- = 0 \leq v_1 \leq v_1^+ = 1.0$,
- Flow rate of vent 2: $v_2^- = 0.5 \leq v_2 \leq v_2^+ = 1.5$,
- Flow rate of vent 3: $v_3^- = 0.0 \leq v_3 \leq v_3^+ = 1.0$,
- Flow rate of vent 4: $v_4^- = 0.5 \leq v_4 \leq v_4^+ = 1.5$,
- Position of vent 1: $(x_1, x_2, x_3)^- = (-0.5, 1.0, 2.0) \leq (x_1, x_2, x_3) \leq (x_1, x_2, x_3)^+ = (1.5, 3.0, 4.0)$,
- Position of vent 2: $(x_1, x_2, x_3)^- = (-1.5, 2.0, 2.0) \leq (x_1, x_2, x_3) \leq (x_1, x_2, x_3)^+ = (0.5, 4.0, 4.0)$,
- Position of vent 3: $(x_1, x_2, x_3)^- = (-1.5, 1.0, -1.0) \leq (x_1, x_2, x_3) \leq (x_1, x_2, x_3)^+ = (1.0, 3.0, 1.0)$ and
- Position of vent 4: $(x_1, x_2, x_3)^- = (-0.5, 2.0, -1.0) \leq (x_1, x_2, x_3) \leq (x_1, x_2, x_3)^+ = (1.5, 4.0, 1.0)$.

Figure 5 shows the reduction of the cost function for the 16 parameter set. This cost function Π represents the fraction of uncaptured particles. Shown are the best performing gene (design parameter set, in *red*) as a function of successive generations, as well as the average performance of the entire population of the genes (designs, in *green*). The design parameters $\Lambda = \{\Lambda_1, \Lambda_2, \dots, \Lambda_N\}$ are optimized over the search intervals (16 variables): $\Lambda_i^- \leq \Lambda_i \leq \Lambda_i^+, i = 1, 2, \dots, 16$. We used the following MLA settings:

- Number of design variables: 16,
- Population size per generation: 24,
- Number of parents to keep in each generation: 6,
- Number of children created in each generation: 6,
- Number of completely new genes created in each generation: 12
- Number of generations for re-adaptation around a new search interval: 10 and
- Number of generations: 50.

Table 1 and Fig. 5 illustrate the results. The algorithm was automatically reset around the best gene every 10 generations. The entire 50 generation simulation, with 24 genes per evaluation (1200 total designs) took a few minutes on a laptop, *making it ideal as a design tool*. Shown are the optimization results for successively longer time limits of $T = 1.75, 2.0, 2.25$ and 2.5 s for the 16 parameter set. Shown are the best performing gene (design parameter set, in *red*) as a function of successive generations, as well as the average performance of the entire population of genes (design parameter set, in *green*). Successively allowing longer simulations times allows the vents to trap more particles. We note that, for a given set of parameters, each complete simulation takes on the order of 0.25 s, several thousand parameters sets can be evaluated in well under an hour, *without even exploiting the inherent parallelism of the MLA/GA*.

Remark 8 There are other possible Machine-Learning paradigms such as Artificial Neural Networks (ANNs) that could potentially be useful in such an analysis. ANNs are based on constructing simple input-output type models are essentially adaptive nonlinear regression fits of the form $OUTPUT = \mathcal{B}(INPUT, w_1, w_2, \dots, w_N)$, where \mathcal{B} (the Artificial Neural Network) is constructed from (a) **synapses**,

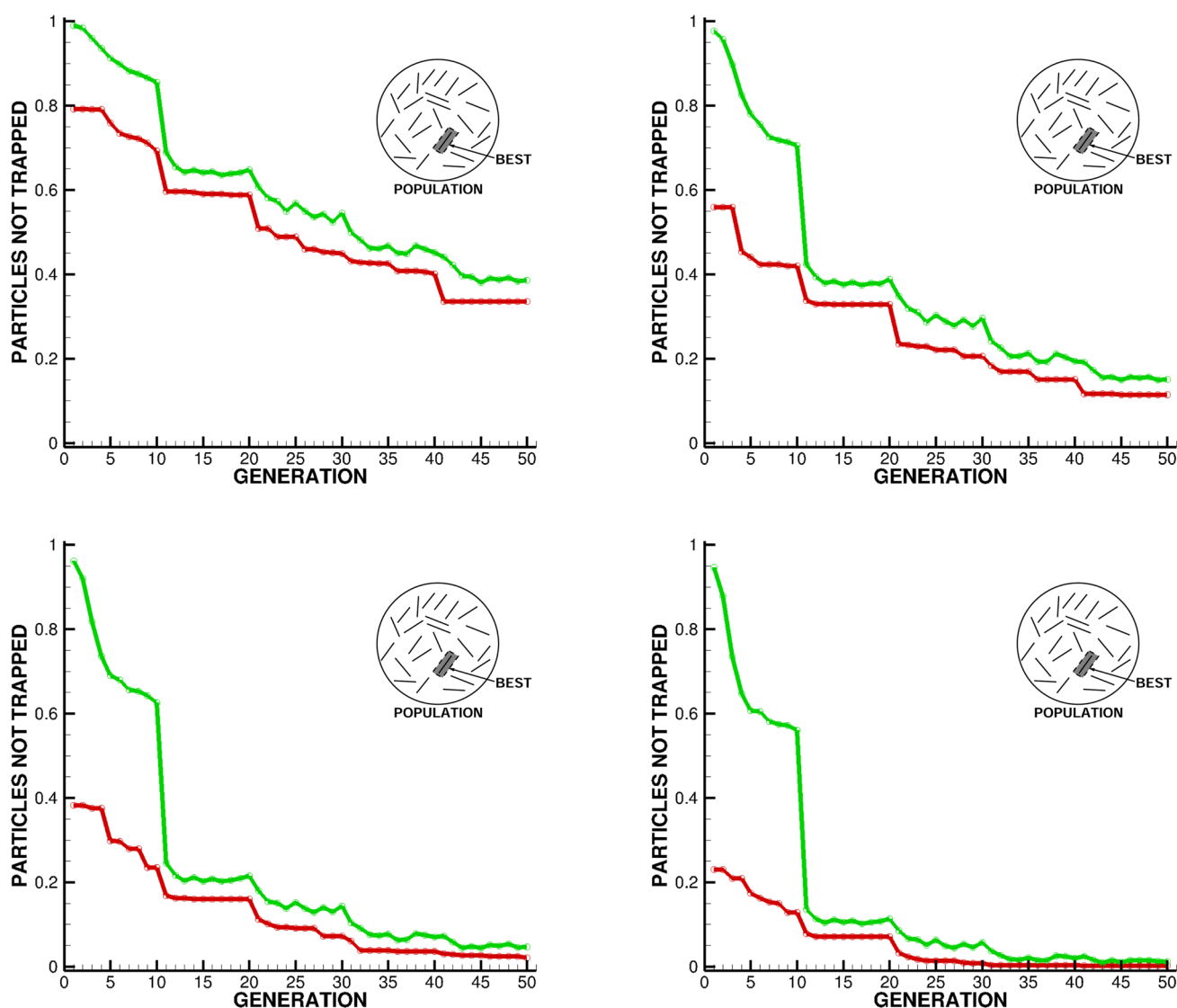


Fig. 5 Optimization for successively longer time limits of $T = 1.75, 2.0, 2.25$ and 2.5 s. Shown is the cost function values (the fraction of particles not trapped) at the end of each generation. The red plot is the cost function associated with the best performing gene

and the green function is the average cost function of the entire population. Successively allowing longer simulation times allows the vents to trap more particles

which multiply inputs by weights that represent the inputs' relevance to the desired output, (b) **neurons**, which add outputs from all incoming synapses and applies activation functions and (c) **training**, which recalibrates the weights ($w_i, i = 1, 2, \dots, N$) to achieve a desired overall output. Ultimately, one constructs a system with optimized weights to mimic an artificial "input-output" brain. For extremely physically-complex systems, these techniques remain unproven, but are actively being investigated in a number of scientific fields.

6 Summary and Future Work

This work developed a combined Digital-Twin and Machine-Learning framework to optimize ventilation systems by building on rapidly computable respiratory emission models developed in Zohdi [1]. This framework ascertains the placement and flow rates of multiple ventilation units, in order to optimally sequester particles released from respiratory emissions such as coughs, sneezes, etc. Numerical examples are provided to illustrate the framework. An extension of the analysis is to consider cases where the change in the

surrounding fluid's behavior, due to the motion of the particles and cough, may be important. The result is a system of coupled equations between the particles and the fluid, requiring spatio-temporal discretization, such as high-fidelity Finite Element or Finite Difference methods, of the classical equations governing the surrounding fluid mechanics (Navier Stokes). Generally such models are computationally quite expensive and ineffective for rapid real-time use, but are useful for detailed offline background analyses, where a rapid response is a nonissue. The continuum discretization is usually combined with a Discrete Element Method for the particle dynamics. There are a variety of such approaches, for example, see Avci and Wriggers [57], Onate et al. [58, 59], Leonardi et al. [60], Onate et al. [61], Bolintineanu et al. [62] and Zohdi [40, 43]. Such models are significantly more complex than the models used in the current work. Along these lines, in Zohdi [40, 43], more detailed, computationally intensive models were developed to characterize the motion of small-scale particles embedded in a flowing fluid where the dynamics of the particles affects the dynamics of the fluid and vice-versa. In such a framework, a fully implicit Finite-Difference discretization of the Navier-Stokes equations was used for the fluid and a direct particle-dynamics discretization is performed for the particles. Because of the large computational difficulty and expense of a conforming spatial discretization needed for large numbers of embedded particles, simplifying assumptions are made for the coupling, based on semi-analytical computation of drag-coefficients, which allows for the use of coarser meshes. Even after these simplifications, the particle-fluid system was strongly-coupled. The approach taken in that work was to construct a sub-model for each primary physical process. In order to resolve the coupling, a recursive staggering scheme was constructed, which was built on works found in Zohdi [40–43]. The procedure was as follows (at a given time increment): (1) each submodel equation (fluid or particle-system) is solved individually, “freezing” the other (coupled) fields in the system, allowing only the primary field to be active, (2) after the solution of each submodel, the associated field variable was updated, and the next submodel was solved and (3) the process is then repeated, until convergence. The time-steps were adjusted to control the rates of convergence, which is dictated by changes in the overall physics. Specifically, the approach was a staggered implicit time-stepping scheme, with an internal recursion that automatically adapted the time-step sizes to control the rates of convergence within a time-step. If the process did not converge (below an error tolerance) within a preset number of iterations, the time-step was adapted (reduced) by utilizing an estimate of the spectral radius of the coupled system. The developed approach can be incorporated within any standard computational fluid mechanics code based on finite difference, finite element, finite volume or discrete/

particle element discretization (see Labra and Onate [63], Onate et al. [58, 59], Rojek et al. [64] and Avci and Wriggers [57]).

Appendix: Websites of General Interest

- Item 1: UV Light Uses:
 - <https://en.wikipedia.org/wiki/Ultraviolet>
 - <https://xenex.com/>
 - <https://aitheon.com/medical-robots>
- Item 2: Masks and decontamination:
 - <https://www.n95decon.org/>
 - <https://www.n95decon.org/bibliography>
 - https://en.wikipedia.org/wiki/Face_masks_during_the_COVID-19_pandemic
 - <https://www.epa.gov/pesticide-registration/list-n-disinfectants-coronavirus-covid-19>
- Item 3: Coughs:
 - <https://en.wikipedia.org/wiki/Cough>
 - <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0169235>
 - <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4676262/>
- Item 4: Filters:
 - <https://en.wikipedia.org/wiki/HEPA>
 - https://en.wikipedia.org/wiki/Minimum_efficiency_reporting_value
- Item 5: Macroscale disease propagation
 - https://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology
 - https://en.wikipedia.org/wiki/Agent-based_model
 - <https://en.wikipedia.org/wiki/Pandemic>
 - <https://www.r-ccs.riken.jp/en/fugaku/research/covid-19/msg-en/>

Acknowledgements The author states that there is no conflict of interest. This work has been partially supported by the UC Berkeley College of Engineering, Lam Research and the USDA AI Institute for Next Generation Food Systems (AIFS), USDA award number 2020-67021-32855.

References

1. Zohdi TI (2020) Modeling and simulation of the infection zone from a cough. *Comput Mech* 64: 1025–1034. <https://doi.org/10.1007/s00466-020-01875-5>
2. Zohdi TI (2020) Rapid simulation of viral decontamination efficacy with UV irradiation. *Eng Comput Methods Appl Mech*. <https://doi.org/10.1016/j.cma.2020.113216>
3. Downes A, Blunt TP (1878) On the influence of light upon protoplasm. In: *Proceedings of the Royal Society of London*. 28

- (190–195): 199–212. Bibcode:1878RSPS...28..199D. <https://doi.org/10.1098/rspl.1878.0109>
4. Anderson JG, Rowan NJ, MacGregor SJ, Fouracre RA, Farish O (2000) Inactivation of food-borne enteropathogenic bacteria and spoilage fungi using pulsed-light. *IEEE Trans Plasma Sci* 28(1):83–88
 5. Battelle (2020) Instructions for Healthcare personnel: preparation of compatible N95 respirators for decontamination by the battelle memorial institute using the battelle decontamination system. <https://www.fda.gov/media/137032/download>
 6. Boyce JM (2016) Modern technologies for improving cleaning and disinfection of environmental surfaces in hospitals. *Antimicrob Resist Infect Control* 5:10. <https://doi.org/10.1186/s13756-016-0111-x>
 7. Card KJ, Crozier D, Dhawan A, Dinh M, Dolson E, Farrokhan N, Gopalakrishnan V, Ho E, King ES, Krishnan N, Kuzmin G, Maltas J, Pelesko J, Scarborough JA, Scott JG, Sedor G, Weaver DT (2020) UV sterilization of personal protective equipment with idle laboratory biosafety cabinets during the Covid-19 pandemic preprint. *Occup Environ Health*. <https://doi.org/10.1101/2020.03.25.20043489>
 8. Heimbuch B, Harnish D (2019) Research to mitigate a shortage of respiratory protection devices during public health emergencies (Report to the FDA No. HHSF223201400158C). Applied Research Associate, Inc
 9. Heimbuch BK, Wallace WH, Kinney K, Lumley AE, Wu C-Y, Woo M-H, Wander JD (2011) A pandemic influenza preparedness study: Use of energetic methods to decontaminate filtering facepiece respirators contaminated with H1N1 aerosols and droplets. *Am J Infect Control* 39(1):e1–e9
 10. Ito A, Ito T (1986) Absorption spectra of deoxyribose, ribosephosphate, ATP and DNA by direct transmission measurements in the vacuum-UV (150–190 nm) and far-UV (190–260 nm) regions using synchrotron radiation as a light source. *Photochem Photobiol* 44(3):355–358
 11. Lin T-H, Tang F-C, Hung P-C, Hua Z-C, Lai C-Y (2018) Relative survival of *Bacillus subtilis* spores loaded on filtering facepiece respirators after five decontamination methods. *Indoor Air* 28(5):754–762
 12. Kanemitsu K et al (2005) Does incineration turn infectious waste aseptic? *J Hosp Infect* 60(4):304–306
 13. Lindsley WG, Martin SB, Thewlis RE, Sarkisian K, Nwoko JO, Mead KR, Noti JD (2015) Effects of Ultraviolet germicidal irradiation (UVGI) on N95 respirator filtration performance and structural integrity. *J Occup Environ Hyg* 12(8):509–517. <https://doi.org/10.1080/15459624.2015.1018518>
 14. Lore MB, Heimbuch BK, Brown TL, Wander JD, Hinrichs SH (2011) Effectiveness of three decontamination treatments against influenza virus applied to filtering facepiece respirators. *Ann Occup Hyg* 56(1):92–101
 15. Marra AR, Schweizer ML, Edmond MB (2018) No-touch disinfection methods to decrease multidrug-resistant organism infections: a systematic review and meta-analysis. *Infect Control Hosp Epidemiol* 39(1):20–31
 16. Mills D, Harnish DA, Lawrence C, Sandoval-Powers M, Heimbuch BK (2018) Ultraviolet germicidal irradiation of influenza-contaminated N95 filtering facepiece respirators. *Am J Infect Control* 46(7):e49–e55
 17. Nerandzic MM, Cadnum JL, Pultz MJ, Donskey CJ (2010) Evaluation of an automated ultraviolet radiation device for decontamination of *Clostridium difficile* and other healthcare-associated pathogens in hospital rooms. *BMC Infect Dis* 10(1):197
 18. Wei J, Li Y (2016) Airborne spread of infectious agents in the indoor environment. *Am J Infect Control* 44(9):S102–S108
 19. Duguid JP (1946) The size and the duration of air-carriage of expiratory droplets and droplet-nuclei. *J Hyg* 44:471–479
 20. Papineni RS, Rosenthal FS (1997) The size distribution of droplets in the exhaled breath of healthy human subjects. *J Aerosol Med* 10(2):105–116
 21. Jianjian Wei, Yuguo Li (2017) Human cough as a two-stage jet and its role in particle transport Published: January 3, 2017 <https://doi.org/10.1371/journal.pone.0169235>
 22. Zhu S, Kato S, Yang J-H (2006) Study on transport characteristics of saliva droplets produced by coughing in a calm indoor environment. *Build Environ* 41(12):1691–1702
 23. Chow CY (1980) An introduction to computational fluid dynamics. Wiley, New York
 24. Morawska L, Johnson GR, Ristovski ZD, Hargreaves M, Mengersen K, Corbett S et al (2009) Size distribution and sites of origin of droplets expelled from the human respiratory tract during expiratory activities. *J Aerosol Sci* 40(3):256–269
 25. VanSciver M, Miller S, Hertzberg J (2011) Particle image velocimetry of human cough. *Aerosol Sci Technol* 45(3):415–422
 26. Kwon S-B, Park J, Jang J, Cho Y, Park D-S, Kim C et al (2012) Study on the initial velocity distribution of exhaled air from coughing and speaking. *Chemosphere* 87(11):1260–1264
 27. Tang JW, Settles GS (2008) Coughing and aerosols. *New Engl J Med* 359(15):e19
 28. 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(3):211–225
 29. Gupta JK, Lin CH, Chen Q (2009) Flow dynamics and characterization of a cough. *Indoor Air* 19(6):517–525
 30. Wan MP, Sze To GN, Chao CYH, Fang L, Melikov A (2009) Modeling the fate of expiratory aerosols and the associated infection risk in an aircraft cabin environment. *Aerosol Sci Technol* 43(4):322–343
 31. Villafruela JM, Olmedo I, Ruiz De Adana M, Mendez C, Nielsen PV (2013) CFD analysis of the human exhalation flow using different boundary conditions and ventilation strategies. *Build Environ* 62:191–200
 32. Nielsen PV (2009) Control of airborne infectious diseases in ventilated spaces. *J R Soc Interface* 6(Suppl 6):S747–S755
 33. Zhang L, Li Y (2012) Dispersion of coughed droplets in a fully-occupied high-speed rail cabin. *Build Environ* 47:58–66
 34. Lindsley William G, Pearce Terri A, Hudnall Judith B, Davis Kristina A, Davis Stephen M, Fisher Melanie A, Khakoo R, Palmer Jan E, Clark Karen E, Celik I, Coffey Christopher C, Blachere Francoise M, Beezhold DH (2012) Quantity and size distribution of cough-generated aerosol particles produced by influenza patients during and after illness. *J Occup Environ Hyg* 9(7):443–449. <https://doi.org/10.1080/15459624.2012.684582>
 35. Burrige HC et al (2021) The ventilation of buildings and other mitigating measures for COVID-19: a focus on wintertime. *Proc R Soc A* 477:20200855. <https://doi.org/10.1098/rspa.2020.0855>
 36. Zohdi TI (2016) A note on firework blasts and qualitative parameter dependency. *Proc R Soc*. <https://doi.org/10.1098/rspa.2015.0720>
 37. Zohdi TI, Cabalo J (2017) On the thermomechanics and footprint of fragmenting blasts. *Int J Eng Sci* 118:28–39
 38. Zohdi TI (2018) Modeling the spatio-thermal fire hazard distribution of incandescent material ejecta in manufacturing. *Comput Mech*. <https://doi.org/10.1007/s00466-018-1617-2>
 39. Zohdi T (2020) A machine-learning framework for rapid adaptive digital-twin based fire-propagation simulation in complex environments. *Comput Methods Appl Mech Eng*. <https://doi.org/10.1016/j.cma.2020.112907>
 40. Zohdi TI (2007) Computation of strongly coupled multifield interaction in particle-fluid systems. *Comput Methods Appl Mech Eng* 196:3927–3950
 41. Zohdi TI (2010) On the dynamics of charged electromagnetic particulate jets. *Arch Comput Methods Eng* 17(2):109–135

42. Zohdi TI (2013) Numerical simulation of charged particulate cluster-droplet impact on electrified surfaces. *J Comput Phys* 233:509–526
43. Zohdi TI (2014) Embedded electromagnetically sensitive particle motion in functionalized fluids. *Comput Particle Mech* 1:27–45
44. Zohdi TI (2020) An agent-based computational framework for simulation of global pandemic and social response on planet X. *Comput Mech*. <https://doi.org/10.1007/s00466-020-01886-2>
45. Zohdi TI (2009) Mechanistic modeling of swarms. *Comput Methods Appl Mech Eng* 198(21–26):2039–2051
46. Zohdi TI (2018) Multiple UAVs for Mapping: a review of basic modeling, simulation and applications. *Annu Rev Environ Resour*. <https://doi.org/10.1146/annurev-environ-102017-025912>
47. Zohdi TI (2019) The Game of Drones: rapid agent-based machine-learning models for multi-UAV path planning. *Comput Mech*. <https://doi.org/10.1007/s00466-019-01761-9>
48. Holland JH (1975) *Adaptation in natural & artificial systems*. University of Michigan Press, Ann Arbor, Mich
49. Holland JH, Miller JH (1991) *Artificial Adaptive Agents in Economic Theory* (PDF). *American Economic Review*. 81 (2): 365–71. Archived from the original (PDF) on October 27, 2005
50. Goldberg DE (1989) *Genetic algorithms in search, optimization & machine learning*. Addison-Wesley, Boston
51. Davis L (1991) *Handbook of genetic algorithms*. Thompson Computer Press, Haryana
52. Onwubiko C (2000) *Introduction to engineering design optimization*. Prentice Hall, Hoboken
53. Goldberg DE, Deb K (2000) Special issue on genetic algorithms. *Comput Methods Appl Mech Eng* 186(2–4):121–124
54. Zohdi TI (2020) A machine-learning framework for rapid adaptive digital-twin based fire-propagation simulation in complex environments. *Comput Methods Appl Mech Eng*. <https://doi.org/10.1016/j.cma.2020.112907>
55. Luenberger D (1974) *Introduction to linear & nonlinear programming*. Addison-Wesley, Menlo Park
56. Gill P, Murray W, Wright M (1995) *Practical optimization*. Academic Press, Cambridge
57. Avci B, Wriggers P (2012) A DEM-FEM coupling approach for the direct numerical simulation of 3D particulate flows. *J Appl Mech* 79:010901-1–7
58. Onate E, Idelsohn SR, Celigueta MA, Rossi R (2008) Advances in the particle finite element method for the analysis of fluid-multi-body interaction and bed erosion in free surface flows. *Comput Methods Appl Mech Eng* 197(19–20):1777–1800
59. Onate E, Celigueta MA, Idelsohn SR, Salazar F, Suárez B (2011) Possibilities of the particle finite element method for fluid-soil-structure interaction problems. *Comput Mech* 48:307–318
60. Leonardi A, Wittel FK, Mendoza M, Herrmann HJ (2014) Coupled DEM-LBM method for the free-surface simulation of heterogeneous suspensions. *Comput Particle Mech* 1(1):3–13
61. Onate E, Celigueta MA, Latorre S, Casas G, Rossi R, Rojek J (2014) Lagrangian analysis of multiscale particulate flows with the particle finite element method. *Comput Particle Mech* 1(1):85–102
62. Bolintineanu DS, Grest GS, Lechman JB, Pierce F, Plimpton SJ, Schunk PR (2014) Particle dynamics modeling methods for colloid suspensions. *Comput Particle Mech* 1(3):321–356
63. Labra C, Onate E (2009) High-density sphere packing for discrete element method simulations. *Commun Numer Methods Eng* 25(7):837–849
64. Rojek J, Labra C, Su O, Onate E (2012) Comparative study of different discrete element models and evaluation of equivalent micromechanical parameters. *Int J Solids Struct* 49:1497–1517. <https://doi.org/10.1016/j.ijsolstr.2012.02.032>

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