Data-driven control of COVID-19 in buildings: a reinforcement-learning approach

Ashkan Haji Hosseinloo, Saleh Nabi, Anette Hosoi, and Munther A. Dahleh *Fellow, IEEE*

***Abstract*—In addition to its public health crisis, COVID-19 pandemic has led to the shutdown and closure of workplaces with an estimated total cost of more than $16 trillion. Given the long hours an average person spends in buildings and indoor environments, this research article proposes data-driven control strategies to design optimal indoor airflow to minimize the exposure of occupants to viral pathogens in built environments. A general control framework is put forward for designing an optimal velocity field and proximal policy optimization, a reinforcement learning algorithm is employed to solve the control problem in a data-driven fashion. The same framework is used for optimal placement of disinfectants to neutralize the viral pathogens as an alternative to the airflow design when the latter is practically infeasible or hard to implement. We show, via computational simulations, that the control agent learns the optimal policy in both scenarios within a reasonable time. The proposed data-driven control framework in this study will have significant societal and economic benefits by setting the foundation for an improved methodology in designing case-specific infection control guidelines that can be realized by affordable ventilation devices and disinfectants.**

***Note to Practitioners*—This paper is motivated by the problem of COVID-19 infection spread in enclosed spaces but it also applies to other airborne pathogens. Airborne disease contagion often takes place in indoor environments; however, ventilation systems are almost never designed to take this into account so as to contain the spread of the pathogens. This is mainly because airflow design requires solving high-dimensional nonlinear partial differential equations known as Navier Stokes equations in fluid dynamics. In this paper, we propose a data-driven approach for solving the control problem of pathogen containment without solving the fluid dynamics equations. To this end, we first mathematically formulate the problem as an optimal control problem and then cast it as a reinforcement learning (RL) task. Reinforcement learning is the data-driven science of sequential decision-making and control in which the controller finds an optimal solution by systematic trial and error and without access to the system dynamics, i.e. fluid and pathogen dynamics in this paper. We employ an state-of-the-art RL algorithm, called PPO, to solve for optimal airflow in a room so as to minimize the exposure risk of occupants. Once it is calculated, the optimal airflow could be realized, via reverse engineering, by proper placement of the ventilation equipment, e.g. inlets, outlets, and fans. As an alternative to the airflow design, we use the same proposed data-driven techniques to find an optimal placement for pathogen disinfectants if there exists one, such as, hydrogen peroxide for COVID-19. Our results show the efficacy of our data-driven approach in designing an steady-state controller with full access to the system states. In future research, we will address the controller design with sparse measurements of the system states.**

***Index Terms*—Disease control, COVID-19, reinforcement learning, data-driven control, HVAC system**

A. Haji Hosseinloo is with Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: ashkanhh@mit.edu).

S. Nabi is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139 USA (e-mail: nabi@merl.com).

A. Hosoi is with Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: peko@mit.edu).

M. A. Dahleh is with Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: dahleh@mit.edu).

I. Introduction

In addition to its public health crisis, COVID-19 pandemic led to the shutdown and closure of workplaces, retail and commercial spaces, schools, and restaurants among many others. The lockdown has severely impacted the US economy and caused millions of temporary and permanent job losses in the US alone. The US unemployment rose higher in the first three months of COVID-19 than it did in two years of the Great Recession: 14.4% in April 2020 versus 10.6% in January 2010. Safe reopening and containing the spread of COVID-19 in indoor spaces is an important step towards economy recovery without risking people's health. This requires a good understanding of the disease transmission and designing effective engineering controls that is the purpose of this research article.

Although WHO and CDC ignored, in the beginning, the importance of airborne transmission for the disease, observational studies and computational models [1]–[5] show that COVID-19 can remain aloft in air for a few hours and pose a risk of exposure at distances

beyond the commonly adopted 6-feet social distancing. Hence, indoor ventilation and airflow play a big role in containment or spread of COVID-19 pathogens, especially knowing that people in industrialized countries spend more than 90% of their lifetime indoors [6]. This is not specific to COVID-19 and the ventilation potential for preventing airborne disease transmission has been highlighted in the past [7]–[9]. Despite their importance, indoor ventilation and airflow are usually not designed for disease preventive purposes. Negative pressurized isolation rooms for patients with airborne diseases in hospitals are the only widely-adopted use of airflow design for preventive purposes. Personalized ventilation (PV) which delivers fresh air directly to the occupant's breathing zone is another design concept that can be leveraged for airborne infection control [10]–[14]. PVs are not well explored and can be costly to design and implement.

Designing an effective preventive airflow requires a good transmission model for the disease. Most infection control strategies are mainly based on overly-simplified models of disease transmission developed in the 1930s [15] which can limit the effectiveness of the resulting guidelines. The COVID-19 pandemic, however, gave rise to many studies exploring the host-to-host transmission of the disease with a wide spectrum of model complexity, from analytical well-mixed models to fully-blown 3D Navier-Stokes simulations. Burridge et. al. [16] and Luhar [17] used a well-mixed model to calculate the pathogen concentration and assess the infection risk in built environments. Balachandar et. al. [18] developed a simple model for the time evolution of droplet/aerosol concentration based on a theoretical analysis of the relevant physical processes. Their model ignores ambient mean flow and, hence, is not suitable for ventilation and airflow design. On the other end of the spectrum of the model complexity, computational fluid dynamics (CFD) simulations were adopted to solve Navier-Stokes equations coupled with pathogen transport equation to solve for spatiotemporal pathogen concentration in supermarkets [19], urban buses [20], [21], and a music classroom [22]. Such detailed models are computationally too expensive for optimization and controller design. This is even more problematic for online control and optimization where computational burden can easily make the real-time design impossible. In the middle range of the model complexity, Lau et. el. [23], [24] modeled pathogen concentration evolution as an advection-diffusion equation with uniform velocity field. Using the concentration, different risk measures, such as, time to infection were calculated.

As discussed above, transmission models of airborne diseases are either too simplistic for an effective controller design or too complex for a computationally-feasible model-based controller design. Furthermore, much is still unknown about bio- and fluid-physics of COVID-19 pathogens, and hence, physics-based models may ignore some important aspects of the transmission dynamics. Also, in addition to the complex and not-fully-understood transmission dynamics, airflow design depends very much on the interior layout of the space that is often subject to continuous change, e.g., by changing seating layout in a restaurant or classroom. This warrants building a new model for the space and redesigning its controller every time there is a change in the space layout. For the above-mentioned reasons, a model-free and data-driven approach for airflow design is a better alternative. That is why we take a data-driven approach, namely, reinforcement learning for the controller design in this article.

The efforts for applying reinforcement learning (RL), and deep reinforcement learning (DRL) to fluid mechanics started only a few years ago in 2016 [25], [26] and are still at an early stage, with only a handful of pioneering studies. In terms of the specific applications within the fluid mechanics, majority of these studies focus on drag reduction on a two-dimensional cylinder submerged in a fluid flow [27]–[31]. For instance in [27] DRL with proximal policy optimization (PPO) algorithm is employed to reduce the drag by 8% via controlling mass flow rates of two small jets on the sides of the cylinder. The second most-explored fluid mechanics application is fish swimming [26], [32]–[35]. For example in [33], the authors study the collective swimming of fishes and use DRL to find their optimal swimming strategy which turns out to be placing themselves in appropriate locations in the wake of other swimmers and intercepting judiciously their shed vortices.

In this study, we employ data-driven and RL techniques to design effective indoor airflow in order to reduce the disease exposure risk for the occupants. We also apply the said techniques to optimally place pathogen neutralizers (disinfectants) in a room for a given airflow. To the best of our knowledge, this is the first application of RL in airborne disease transmission control. To this end, we first formulate the control problem in section II after which the RL framework and methodology are discussed in section III. Then we present and discuss the results in section IV before we conclude the paper with some remarks and future directions in section V.

The main contribution of this paper is introducing the application of RL for controlling the airborne disease transmission in indoor environments for which we formulate the problem in a control set-up and

cast it into an RL framework for the very first time. Other contributions of this paper include making use of the domain knowledge as an inductive bias (e.g. parameterizing the airflow as a double-vortex flow) to reduce the amount of required data for training.

II. Control problem formulation

Let us consider, with no loss of generality, an exemplary case of a restaurant where airborne pathogens are released near a table with one or more infected customers at that table. We would like to design an airflow using available heating, ventilation, and air-conditioning (HVAC) system (e.g., fans and air-conditioners) that minimizes the exposure of the rest of the customers to the pathogens. This is the first of the two control problems we study in this paper and is schematically shown in Fig.1. Here, we consider the velocity field, $\boldsymbol{v}$, as the control variable and model transport dynamics of the virus by an advection-diffusion equation. We model the notion of exposure risk, that is also the control problem's performance metric J_e, as integral of the pathogen concentration, c, over a given time period, $[0, T]$, and a region of interest, Ω. We then formulate the control problem as:

$$\begin{aligned} \min_{\boldsymbol{v}} \quad & J_e = \int_\Omega \int_0^T c(\boldsymbol{x}, t)\, dt\, d\boldsymbol{x} \\ \text{s.t.} \quad & \nabla.\boldsymbol{v} = 0 \\ & \frac{\partial c}{\partial t} + \boldsymbol{v}.\nabla c - K\nabla^2 c = f(\boldsymbol{x}, t) - \lambda c, \end{aligned} \tag{1}$$

where, the first and second constraints are incompressibility condition and the pathogen transport dynamics, respectively. Diffusion coefficient is denoted by K. Similar to [23], the overall effect of air and virus removal via the HVAC system is modeled by the term, $-\lambda c$. The coefficient λ defines strength of the HVAC system and can be a function of time and space. Spatial coordinate and time are denoted by $\boldsymbol{x}$ and t, respectively, and $f(\boldsymbol{x}, t)$ is the virus source, the location of which is assumed to be known. This is a fair assumption for symptomatic infected occupants. We assume Neumann boundary condition for the concentration, i.e. $\partial c/\partial \boldsymbol{n} = 0$ where $\boldsymbol{n}$ is the normal to the boundary, $\partial\Omega$.

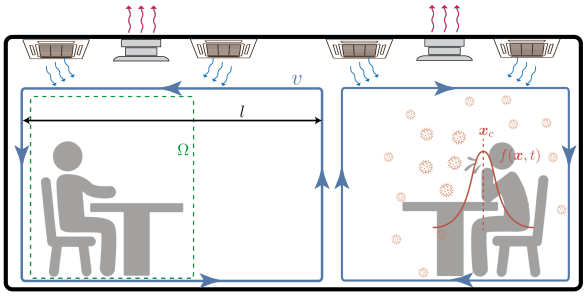

Figure 1. Control Problem 1: schematics of a room with the pathogen source, $f(\boldsymbol{x}, t)$ (with a Gaussian spatial distribution centered at $\boldsymbol{x}_c$) and the region of interest, Ω. A parameterized family of velocity field, namely, the double-vortex airflow is chosen in this study and is schematically shown by blue rectangles with arrows. The length of the left vortex is designated by l.

It is worth to mention that the actual control variables, in practice, are usually the location of the air inlet and outlet, as well as features of the inlet air, such as, temperature and velocity. Optimal values for these control variables can be found once we design the optimal velocity field, though it is not a trivial problem. Alternatively, the control problem could be formulated such that the above-mentioned variables are set as the control variables. In this case, airflow dynamics should be added to the problem constraints, e.g. in the form of Navier Stokes equations. This is beyond the scope of this study due to its computational complexity; however, the proposed control and RL framework in this study will directly apply to this formulation as well.

In addition to the airflow design, an alternative solution to control the virus spread is to neutralize them. Hydrogen Peroxide (HP), H_2O_2, particularly in its ionized state, has been shown to be an effective disinfectant for the COVID-19 virus [36]–[38]. As the second control problem, we consider here another set-up where HP is used to neutralize and disinfect the COVID-19 pathogens (see Fig.2). The general set-up is as the first one, with the difference that we assume a fixed uniform airflow and try to optimize the location of the HP source so as to minimize the same performance metric in Eq.1. The control problem is formulated as below:

$$\begin{aligned} \min_{\boldsymbol{x}_{hp}} \quad & J_e = \int_\Omega \int_0^T c(\boldsymbol{x}, t)\, dt\, d\Omega \\ \text{s.t.} \quad & \nabla.\boldsymbol{v} = 0 \\ & \frac{\partial c}{\partial t} + \boldsymbol{v}.\nabla c - K\nabla^2 c = f(\boldsymbol{x}, t) - \lambda c + g_1(c, c_{hp}) \\ & \frac{\partial c_{hp}}{\partial t} + \boldsymbol{v}.\nabla c_{hp} - K_{hp}\nabla^2 c_{hp} = f_{hp}(\boldsymbol{x}, t) - \lambda c_{hp} \\ & \qquad\qquad + g_2(c, c_{hp}), \end{aligned} \tag{2}$$

where, c_{hp}, K_{hp}, and $f_{hp}(\boldsymbol{x}, t)$ designate the con-

centration, diffusivity, and the source of HP, respectively. The chemical interaction between COVID-19 and HP particles are captured by the functions g_1 and g_2. Here, we consider a simple proportional model for these interactions as: $g_1(c, c_{hp}) = \alpha_1\, c\, c_{hp}$ and $g_2(c, c_{hp}) = \alpha_2\, c\, c_{hp}$, where α_1 and α_2 are constants. We model the HP source, as well as, the COVID-19 source in both control problems (Eqs. 1 and 2) as time-invariant, spatially Gaussian-distributed functions: $R_{(.)}/\pi\epsilon \exp(-(\boldsymbol{x} - \boldsymbol{x}_{(.)})^2/\epsilon)$, where, the subscript $(.) = c$ or hp, shows whether the parameter pertains to COVID or HP. The strength and spread of the source are decided by the parameters R and ϵ, respectively. We would like to emphasize that the decision variable in the second control problem defined in Eq.2 is the center location of the HP source, i.e., $\boldsymbol{x}_{hp}$.

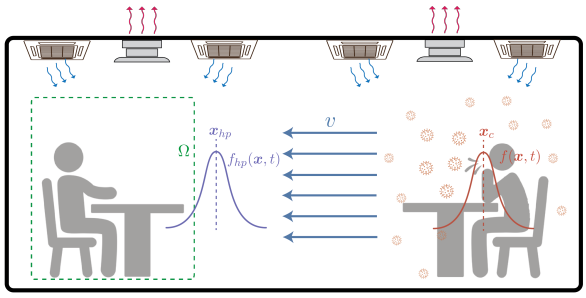

Figure 2. Control Problem 2: Schematics of a room with the pathogen and HP sources, $f(\boldsymbol{x}, t)$ and $f_{hp}(\boldsymbol{x}, t)$, both with Gaussian spatial distributions centered at $\boldsymbol{x}_c$ and $\boldsymbol{x}_{hp}$, respectively. The region of interest is denoted by Ω. For this control problem, a constant uniform velocity field is considered in this study. The control objective is to find an optimal center position ($\boldsymbol{x}_{hp}$) for the HP disinfectant source.

III. Reinforcement Learning Framework

Reinforcement Learning is the data-driven science of sequential decision making. It is about learning the optimal behavior/decisions in an environment to maximize a notion of cumulative or average reward. This optimal behavior is learned through interactions with the, often unknown, environment, similar to children exploring the world around them and learning the actions that help them achieve a goal.

In the RL framework, the agent, aka the controller, takes action a at state s and observes an immediate reward r after moving to next state s'. The goal of the agent is to find an optimal policy $\pi^*(s)$, i.e., optimal control law that maximizes the discounted cumulative rewards in expectation, usually referred to as the value function $V^\pi(s)$. In practice, we often maximize the value function weighted by the state distribution under the policy $\rho^\pi(s)$, i.e., $\int_s \rho(s)V(s)\, ds$. For the control problem defined by Eq.1, the state is the spatially-continuous concentration field and the action is the continuous velocity field. We define the immediate reward as:

$$r = \int_{t_1}^{t_2} \int_{\Omega} c(\boldsymbol{x}, t)\, d\boldsymbol{x}\, dt, \tag{3}$$

where, t_1 and t_2 are timestamps at states s_1 and s_2, respectively. In the second control problem defined by Eq.2, the state is the aggregation of both COVID-19 and HP concentration fields, as well as, the velocity field. The action is the center location of the HP source, i.e. $\boldsymbol{x}_{hp}$. The reward function remains the same as that in Eq.3. We would also like to point out the relationship between the objective functions in Eqs.1 and 2 and the reward function defined above: the objective functions are, in fact, sum of immediate rewards accumulated between $t = 0$ to $t = T$.

Reinforcement learning algorithms can, in general, be categorized into policy-based and value-based methods. In a value-based method, the algorithm learns an estimate of the optimal value function. Q-learning is probably the most well-known and one of the earliest value-based RL methods[1]. The policy here is implicit and can be derived directly from the value function.

In policy-based methods, however, we explicitly build a representation of a policy which we improve by, e.g., a policy-gradient (PG) technique. Policy-based methods have a few advantages over their value-based counterparts. They can solve for both deterministic and stochastic optimal policies. Furthermore, in many cases the optimal policy has a simpler form than the optimal value function, and hence, it is easier to directly learn the optimal policy [39], [40]. They also handle continuous action space better than their value-based counterparts [41].

Among different policy gradient algorithms, we use PPO [42], one of the state-of-the-art algorithms for training our RL agent in this study. Compared to many other policy gradient algorithms, such as, trust region policy optimization (TRPO), PPO is mathematically less complex, and hence, computationally faster [27]. It is also shown to be often more data-efficient. The most important aspect of the PPO algorithm is its clipped surrogate objective which prevents taking big steps in potentially wrong directions when updating the policy using the gradient decent. We would like to point out that it is not the main objective of this study to find the best RL algorithm among all the available off-the-shelf algorithms, nor it is to devise a

[1]Q-learning is also used extensively as part of many policy-based algorithms to learn the value or action-value function that serves as the critic for the actor.

Table I: Parameters description for the advection-diffusion dynamics

parameter description	symbol	numerical value
room length	l_x	8 m
room height	l_y	4 m
diffusion coeff. for COVID-19	K	0.022 m^2/s
diffusion coeff. for HP	K_{hp}	0.022 m^2/s
HVAC air-exchange coeff. (COVID-19)	λ	0.0085 $1/s$
HVAC air-exchange coeff. (HP)	λ_{hp}	0.0085 $1/s$
source intensity for COVID-19	R	2.5 particle$/s$
source intensity for HP	R_{hp}	2.5 particle$/s$
interaction coeff. between COVID-19 and HP	α_1	0.2 m^2/particle. s
interaction coeff. between HP and COVID-19	α_2	0.2 m^2/particle. s

new algorithm. The main objective of the simulations is to show the feasibility and proof-of-concept of the proposed RL framework for containing the indoor airborne pathogens. With that said, we will compare the PPO algorithm with two other widely-used algorithms, namely TRPO and A2C, before deciding to pick PPO as our main RL agent. In the next section, we delineate the computational simulations, discuss how the PPO algorithm is applied to the two control problems, and present and discuss the results.

IV. Results and Discussion

We use a Python library called FEniCSx to solve the advection-diffusion dynamics in Eqs.1 and 2. FEniCSx is a popular open-source computing platform for solving partial differential equations (PDEs) that is based on finite element (FE) methods [43]. As a FE-based solver, FEniCSx requires PDEs in variational form, aka the weak form. The variational reformulation of the boundary-value problems in Eqs.1 and 2 are presented in the appendix. A linear Lagrange element is used for both test and trial functions in the variational formulation. As discussed in section III, PPO, as well as TRPO and A2C are policy-gradient methods for which we need a parameterized policy. For the control problem in Eq.1, the policy to optimize is the velocity field $\boldsymbol{v}$ which we parameterize by the parameter vector $\boldsymbol{\theta}$ and denote it as $\boldsymbol{v}^{\boldsymbol{\theta}}$. For this control problem, we focus on a specific family of velocity field, often known as double-vortex velocity (schematically shown in Fig.1). We chose this particular family of velocity field for a number of reasons. First, it is relatively easy to realize this velocity field, e.g., by adjusting the location of inlets and outlets or the direction of the inflow air. Second, it can significantly reduce the dimension of the action space, or equivalently the size of the parameter vector $\boldsymbol{\theta}$, by expressing the vortex dynamics using its geometric center (or vortex length) and strength. In our case we consider the former only (fixing the strength) resulting in $\boldsymbol{\theta} = \{l\}$, where l is the length of the left vortex. And last but not least, stemming from domain knowledge, the double- and multiple-vortex fields can potentially create a notion of *air distancing* by isolating the infected region from a non-infected region in the room quite easily. The vortices in double-vortex flow are reminiscent of convection rolls encountered in forced convection in buildings [44], [45]. The two-dimensional double-vortex velocity field $\boldsymbol{v}^{\boldsymbol{\theta}} = \boldsymbol{v}^l$ is mathematically formulated as below:

$$\boldsymbol{v}^l = \begin{cases} \big(w_x^l \sin(\frac{\pi x}{l}) \cos(\frac{\pi y}{l_y}), \\ -w_y^l \cos(\frac{\pi x}{l}) \sin(\frac{\pi y}{l_y})\big) : & \text{if } x \leq l \\ \big(w_x^r \sin(\frac{\pi (x-l)}{l_x - l}) \cos(\frac{\pi y}{l_y}), \\ -w_y^r \cos(\frac{\pi (x-l)}{l_x - l}) \sin(\frac{\pi y}{l_y})\big) : & \text{if } x > l, \end{cases} \tag{4}$$

where, x and y are the spatial coordinates in the horizontal and vertical directions, respectively, with l_x and l_y denoting the room length in their respective directions. The parameters w's define the strength of the velocity field. These parameters cannot be chosen independently as the velocity field needs to satisfy the incompressibility condition ($\nabla . \boldsymbol{v}^l = 0$). Enforcing this condition yields:

$$\begin{aligned} w_x^l / l &= w_y^l / l_y \\ w_x^r / (l_x - l) &= w_y^r / l y. \end{aligned} \tag{5}$$

For all the simulations in this study we use $w_x^l = w_x^r = 1.0\, m/s$. For the RL algorithm implementation we use the open-source Stable Baselines3 (SB3) Python library. Stable Baselines3 is a set of reliable implementations of reinforcement learning algorithms in PyTorch [46]. For user-defined environments SB3 requires a gym-compatible environment which we create using the FEniCSx library in Python.

Before attempting to solve any of the control problems outlined in section II, we do a mesh study to find an adequately-fine mesh size for the simulations. We use

a uniform finite element mesh over the entire domain consisting of cells, which in 2D are triangles with straight sides. The mesh size parameter is the tuple (n_x, n_y), where, n_x and n_y specify the number of rectangles (each divided into a pair of triangles) in the x and y directions, respectively. The total number of triangles (cells) thus becomes $2 \times n_x \times n_y$. For the mesh study we run simulations with parameters in Table I, $l = 4.0\,m$, $T = 600\,s$, and a range of values for $n_x = 2n_y$. Figure 3 depicts the performance metric J_e calculated over the entire room (Ω = entire room) as a function of n_x. As shown in the figure, there is not much improvement in the results for mesh sizes finer than $n_x = 80$; thus, we set $n_x = 2n_y = 80$ for all the subsequent simulations in this study.

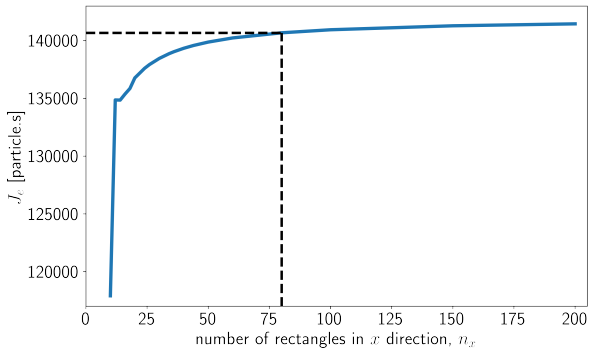

Figure 3. The performance metric, J_e as a function of the mesh size, n_x with parameters in Table I, left vortex length $l = 4.0\,m$, total simulation time $T = 600\,s$, and the region of interest Ω = entire room. $n_x = 2n_y = 80$ is chosen for all the subsequent simulations.

For both control problems we assume a center location for the COVID-19 source at $\boldsymbol{x}_c = (6, 2)\,m$, and the time period of interest as $T = 600\,s$. Also, parameters in Table I are used for all the simulations unless otherwise specified. For the first control problem, i.e. Eq.1, we first find the ground-truth optimal vortex length l^{opt} by brute-force simulations. For this problem we consider the left quarter of the room to be the region of interest; $\Omega = \{\boldsymbol{x}; x \leq 2\,m\}$. Figure 4 shows how the performance metric varies as the length of the left vortex changes for two different values of pathogen diffusivity. As shown in the figure, the optimal vortex length depends on the pathogen diffusivity. A naive guess for the optimal vortex length will be $l = 2\,m$ in the hope of isolating the left quarter from the rest of the room. However, because of the diffusion phenomenon some of the pathogens will still find their way to the left quarter of the room in spite of the vortices. For this very reason, the left vortex needs to be extended beyond the region of interest, i.e. $l^{\text{opt}} > 2$. In fact, the more diffusive the pathogens are the longer the left vortex should be. Simulation results in Fig.4 corroborates this; the optimal left vortex length for $K = 0.022\,m^2/s$ and $K = 5 \times 0.022\,m^2/s$ is $3\,m$ and $4\,m$, respectively. It is also worth to note that the abrupt change in the slopes at $x = 6\,m$ is because of the pathogen source being located at this very same $x-$coordinate.

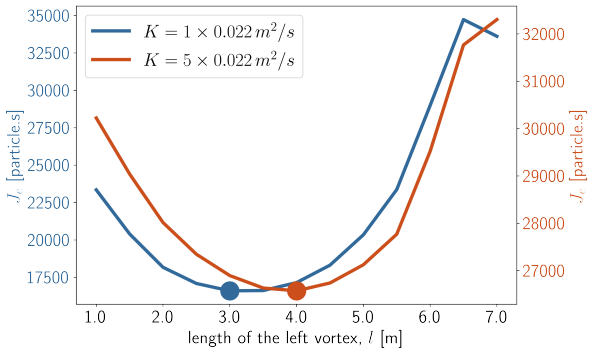

Figure 4. The performance metric, J_e as a function of the left vortex length for two different values of diffusion coefficients. The optimal values minimizing the performance metric are marked by solid circles.

We first train three RL algorithms (PPO, A2C, and TRPO) to learn the optimal length of the left vortex (l) of the double-vortex field. The same neural network with ReLU activation functions (for both policy and value functions) are used for the three agents. The main hyper-parameters of all the three agents were optimized via grid search (the optimized hyper-parameters for the PPO agent are shown in table II). The system was not sensitive to the remaining hyper-parameters, and hence, the default values were used for them. For this and all the other simulations pertaining to the first control problem, we train the control agents for a total time of $6 \times T = 6 \times 600\,s$. The environment is reset every T seconds. In all the simulations, the time increment is one second, that also means every step from the perspective of the RL agent takes one second. Figure 5 shows the training performance of the three RL agents in learning the length of the left vortex. The figure shows that both PPO and TRPO algorithms outperform the A2C in terms of both mean and variance; they learn a left-vortex length that is closer to the optimal value and they learn it more reliably. As discussed in section III, PPO is computationally much faster than TRPO; hence, given their comparable results, we pick PPO as our RL agent for the rest of the simulations.

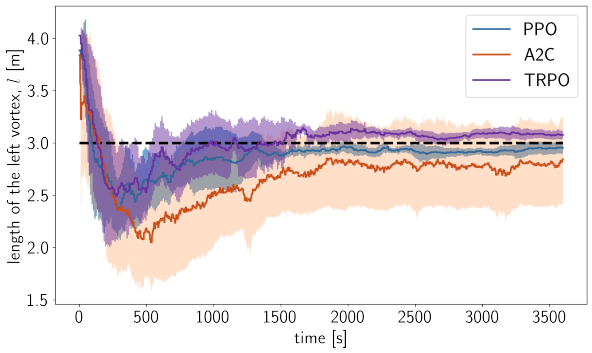

Figure 5. Training performance for learning optimal length of the left vortex (l) of the double-vortex field, averaged over 10 independent runs for three different RL agents: PPO, A2C, and TRPO. The solid lines and the shaded areas show the mean and the variance (one standard deviation) of the learned vortex length, respectively, over the 10 runs. The dashed line show the ground-truth optimal length for diffusion coefficient of $K = 1 \times 0.022\, m^2/s$, found via brute-force simulations.

We would also like to test our RL agent's performance for different values of diffusion coefficient. Figure 6 shows the training performance of the RL agent (PPO) in learning the length of the left vortex for two different values of the diffusion coefficient. The figure illustrates the agent's performance in terms of the mean and variance of the learned policy (left vortex length) over 10 independent runs. As shown in the figure, the agent learns a reliable approximate of the optimal policy in roughly $1000\, s \approx 17$ mins for both values of the diffusion coefficient.

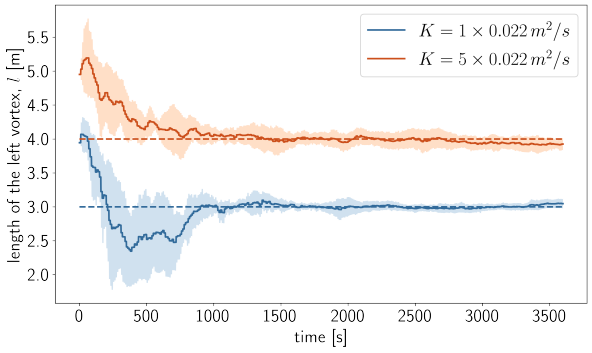

Figure 6. Training performance for learning optimal length of the left vortex (l) of the double-vortex field, averaged over 10 independent runs for two values of the diffusion coefficient. The solid lines and the shaded areas show the mean and the variance (one standard deviation) of the learned vortex length, respectively, over the 10 runs. The dashed lines show the respective ground-truth optimal lengths found via brute-force simulations.

Next, we would like our RL agent to learn the optimal policy for the second control problem as stated in Eq.2. Just like the first control problem, we first find the ground-truth optimal policy, i.e. the optimal center position of the HP source, by brute-force simulations. For the sake of computational simplicity, we fix the center position in the y-direction and optimize for the position in the x-direction; $\boldsymbol{x}_{hp} = (x_{hp}, y_{hp}) = (x_{hp}, 3\, m)$. For this problem, we assume a uniform velocity field of $\boldsymbol{v}(x, y) = (-0.015, 0)\, m/s$. We also consider the left half of the room to be the region of interest, i.e., $\Omega = \{\boldsymbol{x}; x \leq 4\, m\}$.

Table II: Hyper-parameters for the PPO algorithm

hyper-parameter	numerical value
learning rate	0.005
number of steps to run per update	10
mini-batch size	10
number of epochs	10
discount factor	0.99

Figure 7 depicts how the performance measure varies by the x position of the HP source, and that it is minimized at $x_{hp}^{\text{opt}} = 4.5\, m$. As a first guess, one may go for a position as close to the pathogen source as possible to maximally neutralize the pathogens. However, because of the diffusion, and more importantly, advection in this case, the optimal position for the HP source is not the closest to the COVID-19 source. Figure 8 shows the performance of the RL agent in learning this optimal position over 10 independent runs. All the hyper-parameters of the RL algorithm remain the same, except the total simulation time that is extended to $8 \times T = 8 \times 600\, s$. The simulation results show that the agent learns a reliable approximate of the optimal policy in roughly $3000\, s = 50$ mins.

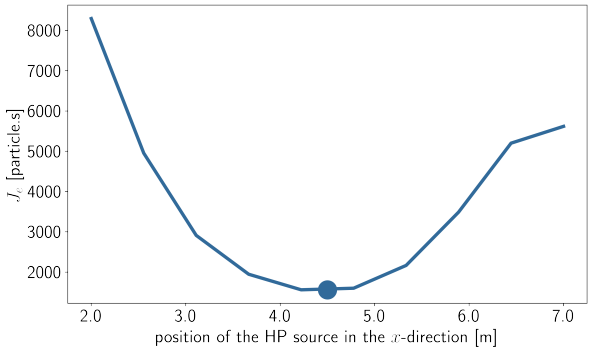

Figure 7. The performance metric, J_e as a function of the HP source location in the x direction. The optimal value minimizing the performance metric is marked by a solid circle.

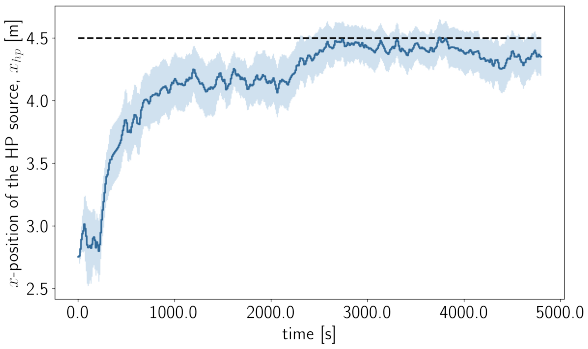

Figure 8. Training performance for learning optimal center position of the HP source in the x direction (x_{hp}^{opt}), averaged over 10 independent runs. The solid line and the shaded area show the mean and the variance (one standard deviation) of the learned x-position of the HP source, respectively, over the 10 runs. The dashed line show the ground-truth optimal x-position of the HP source.

V. Conclusion and Future Work

In this study we investigated data-driven control of COVID-19 in indoor environments. We put forward the idea of designing indoor airflow to contain spread of viral pathogens. The control problem is formulated in a general set-up and the PPO RL algorithm is employed to learn the optimal control law, i.e. the optimal airflow. Transport dynamics of the pathogens are modeled by advection-diffusion equations and a parameterized double-vortex field is chosen as a class of velocity fields to be optimized by the control agent. By using such a compactly-parameterized velocity field, we significantly reduce the dimension of the action space, which subsequently reduces the amount of required data for the RL agent to learn a good policy. Simulation results show that the agent can learn the optimal lengths of the vortices in less than 17 mins.

As a secondary control problem, we also studied the feasibility of optimal placement of disinfectants in a room in order to minimize the infection risk of occupants in a sub-space of the room. We showed that our learning-based controller can learn the optimal location of the disinfectant in less than 50 mins. Given the computational complexity of the CFD simulations, lack of knowledge about the fluid-physics of pathogens transport, and frequent change of interior layout of enclosed spaces, the data-driven nature of the proposed ideas makes them particularly advantageous over their model-based counterparts.

Despite its simplicity, the double-vortex velocity field may not be a good choice for designing an effective airflow in more complex built environments, such as, large offices or theaters with many cubicles and seats. In this case, one can employ more complex velocity fields, e.g., multiple-vortex or a number of point vortices, and optimize for their geometric centers and intensities. Also, for the second control problem, more than one disinfectant could be used and optimized for.

Another limitation of the current study is that we looked for time-invariant optimal solutions in both control problems, i.e., a time-invariant optimal double-vortex and a time-invariant optimal location for the disinfectant. This was a good solution mainly because we assumed a stationary source of virus, as well as, a long enough period of interest (T) for the pathogens to diffuse after the initial transient. Despite referred to as a limitation, the time-invariant nature of the solution helps considerably with reducing the amount of required data for training. However, if very short transient time is of interest or the pathogen source is moving, a time-varying solution (double-vortex with time-varying length or disinfectant with time-varying location) will probably be much more effective. This will be the future work and an extension to the current study.

In general, the proposed data-driven control framework in this study can have significant societal and economic benefits by setting the foundation for an improved methodology in designing case-specific infection control guidelines that can be realized by affordable HVAC devices and disinfectants. Implementing the proposed design and control guidelines helps mitigate the spread of airborne diseases, such as COVID-19, and hence, can save tens of thousands of lives worldwide. In the case of COVID-19 or other potential pandemic-causing viruses, containing the indoor virus spread will also facilitate reopening of schools, universities, offices, and restaurants, to name a few, which in turn, speeds up the recovery of the US and the world economy. This will help low-income communities in particular, by allowing small businesses to open up their operation to avoid major income loss in these vulnerable communities.

References

[1] Li, Y. *et al.* Evidence for probable aerosol transmission of sars-cov-2 in a poorly ventilated restaurant. *medRxiv* (2020).

[2] Hamner, L. High sars-cov-2 attack rate following exposure at a choir practice—skagit county, washington, march 2020. *MMWR. Morbidity and Mortality Weekly Report* **69** (2020).

[3] Zhang, R., Li, Y., Zhang, A. L., Wang, Y. & Molina, M. J. Identifying airborne transmission as the dominant route for the spread of covid-19. *Proceedings of the National Academy of Sciences* (2020).

[4] Kohanski, M. A., Lo, L. J. & Waring, M. S. Review of indoor aerosol generation, transport, and control in the context of covid-19. In *International forum of allergy & rhinology*, vol. 10, 1173–1179 (Wiley Online Library, 2020).

[5] Morawska, L. & Milton, D. K. It is time to address airborne transmission of coronavirus disease 2019 (covid-19). *Clinical*

Infectious Diseases **71**, 2311–2313 (2020).

[6] Zhang, Y. *Indoor air quality engineering* (CRC press Boca Raton, FL, 2005).

[7] Li, Y. *et al.* Role of ventilation in airborne transmission of infectious agents in the built environment-a multidisciplinary systematic review. *Indoor air* **17**, 2–18 (2007).

[8] Aliabadi, A. A., Rogak, S. N., Bartlett, K. H. & Green, S. I. Preventing airborne disease transmission: review of methods for ventilation design in health care facilities. *Advances in preventive medicine* **2011** (2011).

[9] Qian, H. & Zheng, X. Ventilation control for airborne transmission of human exhaled bio-aerosols in buildings. *Journal of thoracic disease* **10**, S2295 (2018).

[10] Nielsen, P. V. Control of airborne infectious diseases in ventilated spaces. *Journal of the Royal Society Interface* **6**, S747–S755 (2009).

[11] Pantelic, J., Sze-To, G. N., Tham, K. W., Chao, C. Y. & Khoo, Y. C. M. Personalized ventilation as a control measure for airborne transmissible disease spread. *Journal of the Royal Society Interface* **6**, S715–S726 (2009).

[12] Habchi, C., Ghali, K., Ghaddar, N., Chakroun, W. & Alotaibi, S. Ceiling personalized ventilation combined with desk fans for reduced direct and indirect cross-contamination and efficient use of office space. *Energy Conversion and Management* **111**, 158–173 (2016).

[13] Xu, C. & Liu, L. Personalized ventilation: one possible solution for airborne infection control in highly occupied space? (2018).

[14] Ding, J., Yu, C. W. & Cao, S.-J. Hvac systems for environmental control to minimize the covid-19 infection. *Indoor and Built Environment* **29**, 1195–1201 (2020).

[15] Bourouiba, L. Turbulent gas clouds and respiratory pathogen emissions: potential implications for reducing transmission of covid-19. *Jama* **323**, 1837–1838 (2020).

[16] Burridge, H., Fan, S., Jones, R., Noakes, C. & Linden, P. Predictive and retrospective modelling of airborne infection risk using monitored carbon dioxide (2021). URL https://europepmc.org/article/PPR/PPR310871.

[17] Luhar, M. Airborne viral emission and risk assessment in enclosed rooms (2020).

[18] Balachandar, S., Zaleski, S., Soldati, A., Ahmadi, G. & Bourouiba, L. Host-to-host airborne transmission as a multiphase flow problem for science-based social distance guidelines. *International Journal of Multiphase Flow* **132**, 103439 (2020).

[19] Vuorinen, V. *et al.* Modelling aerosol transport and virus exposure with numerical simulations in relation to sars-cov-2 transmission by inhalation indoors. *Safety Science* **130**, 104866 (2020).

[20] Mesgarpour, M. *et al.* Prediction of the spread of coronavirus carrying droplets in a bus-a computational based artificial intelligence approach. *Journal of Hazardous Materials* **413**, 125358 (2021).

[21] Zhang, Z. *et al.* Disease transmission through expiratory aerosols on an urban bus. *Physics of Fluids* **33**, 015116 (2021).

[22] Narayanan, S. R. & Yang, S. Airborne transmission of virus-laden aerosols inside a music classroom: Effects of portable purifiers and aerosol injection rates. *Physics of Fluids* **33**, 033307 (2021).

[23] Lau, Z., Kaouri, K. & Griffiths, I. M. Modelling airborne transmission of covid-19 in indoor spaces using an advection-diffusion-reaction equation. *arXiv e-prints* arXiv–2012 (2020).

[24] Lau, Z., Griffiths, I. M., English, A. & Kaouri, K. Predicting the spatially varying infection risk in indoor spaces using an efficient airborne transmission model. *arXiv preprint arXiv:2012.12267* (2020).

[25] Reddy, G., Celani, A., Sejnowski, T. J. & Vergassola, M. Learning to soar in turbulent environments. *Proceedings of the National Academy of Sciences* **113**, E4877–E4884 (2016).

[26] Gazzola, M., Tchieu, A. A., Alexeev, D., de Brauer, A. & Koumoutsakos, P. Learning to school in the presence of hydrodynamic interactions. *Journal of Fluid Mechanics* **789**, 726–749 (2016).

[27] Rabault, J., Kuchta, M., Jensen, A., Réglade, U. & Cerardi, N. Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control. *Journal of fluid mechanics* **865**, 281–302 (2019).

[28] Fan, D., Yang, L., Wang, Z., Triantafyllou, M. S. & Karniadakis, G. E. Reinforcement learning for bluff body active flow control in experiments and simulations. *Proceedings of the National Academy of Sciences* **117**, 26091–26098 (2020).

[29] Tang, H., Rabault, J., Kuhnle, A., Wang, Y. & Wang, T. Robust active flow control over a range of reynolds numbers using an artificial neural network trained through deep reinforcement learning. *Physics of Fluids* **32**, 053605 (2020).

[30] Paris, R., Beneddine, S. & Dandois, J. Robust flow control and optimal sensor placement using deep reinforcement learning. *Journal of Fluid Mechanics* **913** (2021).

[31] Ren, F., Rabault, J. & Tang, H. Applying deep reinforcement learning to active flow control in weakly turbulent conditions. *Physics of Fluids* **33**, 037121 (2021).

[32] Novati, G. *et al.* Synchronisation through learning for two self-propelled swimmers. *Bioinspiration & biomimetics* **12**, 036001 (2017).

[33] Verma, S., Novati, G. & Koumoutsakos, P. Efficient collective swimming by harnessing vortices through deep reinforcement learning. *Proceedings of the National Academy of Sciences* **115**, 5849–5854 (2018).

[34] Yan, L. *et al.* A numerical simulation method for bionic fish self-propelled swimming under control based on deep reinforcement learning. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* **234**, 3397–3415 (2020).

[35] Zhu, Y., Tian, F.-B., Young, J., Liao, J. C. & Lai, J. A numerical study of fish adaption behaviors in complex environments with a deep reinforcement learning and immersed boundary–lattice boltzmann method. *Scientific Reports* **11**, 1–20 (2021).

[36] Cheng, V., Wong, S., Kwan, G., Hui, W. & Yuen, K. Disinfection of n95 respirators by ionized hydrogen peroxide during pandemic coronavirus disease 2019 (covid-19) due to sars-cov-2. *The Journal of hospital infection* **105**, 358 (2020).

[37] Schwartz, A. *et al.* Decontamination and reuse of n95 respirators with hydrogen peroxide vapor to address worldwide personal protective equipment shortages during the sars-cov-2 (covid-19) pandemic. *Applied Biosafety* **25**, 67–70 (2020).

[38] Kenney, P. A. *et al.* Hydrogen peroxide vapor decontamination of n95 respirators for reuse. *Infection Control & Hospital Epidemiology* **43**, 45–47 (2022).

[39] Haji Hosseinloo, A. *et al.* Data-driven control of micro-climate in buildings: An event-triggered reinforcement learning approach. *Applied Energy* **277**, 115451 (2020). URL http://www.sciencedirect.com/science/article/pii/S0306261920309636.

[40] Hosseinloo, A. H. & Dahleh, M. A. Event-triggered reinforcement learning; an application to buildings' micro-climate control. In *AAAI Spring Symposium: MLPS* (2020).

[41] Hosseinloo, A. H. & Dahleh, M. A. Deterministic policy gradient algorithms for semi-markov decision processes. *International Journal of Intelligent Systems* **37**, 4008–4019 (2022).

[42] Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[43] Alnæs, M. *et al.* The fenics project version 1.5. *Archive of Numerical Software* **3** (2015).

[44] Chen, Q. & Xu, W. A zero-equation turbulence model for indoor airflow simulation. *Energy and buildings* **28**, 137–144 (1998).

[45] Farahmand, A.-m., Nabi, S. & Nikovski, D. N. Deep reinforcement learning for partial differential equation control. In *2017 American Control Conference (ACC)*, 3120–3127 (IEEE, 2017).

[46] Raffin, A. *et al.* Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research* **22**, 1–8 (2021). URL http://jmlr.org/papers/v22/20-1364.html.

APPENDIX

In this section we present the variational form of the PDEs in Eqs.1 and 2 required for the FEniCSx finite-element solver. A straightforward approach to solving time-dependent PDEs by the finite element method is to first discretize the time derivative by a finite difference approximation, which yields a sequence of stationary problems, and then turn each stationary problem into a variational formulation. We use backward Euler, aka implicit Euler discretization:

$$\left(\frac{\partial c}{\partial t}\right)^{n+1} = \frac{c^{n+1} - c^n}{\Delta t}, \tag{6}$$

where, superscript n denotes the quantity at time t_n and Δt is the time discretization parameter.

The basic recipe for turning a PDE into a variational problem is to multiply the PDE by a function w, integrate the resulting equation over the domain D, and perform integration by parts of terms with second-order derivatives. The function w which multiplies the PDE is called a *test* function. The unknown function c to be approximated is referred to as a *trial* function. The trial and test functions belong to certain so-called *function spaces* that specify the properties of the functions. An important feature of variational formulations is that the test function w is required to vanish on the parts of the boundary where the solution c is known. Now multiplying the advection-diffusion equation in Eq.1 by the test function w, integrating it over the domain D, performing integration by parts, and applying the test function properties and boundary conditions, we arrive at the variational form:

$$\int_D \left(\frac{1}{\Delta t}\left(c^{n+1} - c^n\right) w + \left(\boldsymbol{v}.\nabla c^{n+1}\right) w + K\nabla c^{n+1}.\nabla w\right) d\boldsymbol{x} - \int_D f^{n+1} w \, d\boldsymbol{x} + \int_D \lambda c^{n+1} w \, d\boldsymbol{x} = 0. \tag{7}$$

The variational problem now is to find c from the trial space such that Eq.7 holds true for all the test functions, w, in the test space. This variational problem is a *continuous problem*: it defines the solution c in the infinite-dimensional function space (the trial space). The finite element method finds an approximate solution of the continuous variational problem by replacing the infinite-dimensional function spaces by discrete (finite-dimensional) trial and test spaces. FEniCS automatically solves the discrete variational problem.

With the same procedure, we derive the variational form of the coupled advection-diffusion equations in Eq.2 as:

$$\begin{aligned}
&\int_D \left(\frac{1}{\Delta t}\left(c^{n+1} - c^n\right) w_1 + \left(\boldsymbol{v}.\nabla c^{n+1}\right) w_1 + K\nabla c^{n+1}.\nabla w_1\right) d\boldsymbol{x} \\
&+ \int_D \left(\frac{1}{\Delta t}\left(c_{hp}^{n+1} - c_{hp}^{n}\right) w_2 + \left(\boldsymbol{v}.\nabla c_{hp}^{n+1}\right) w_2 + K_{hp}\nabla c_{hp}^{n+1}.\nabla w_2\right) d\boldsymbol{x} \\
&- \int_D \left(f^{n+1} w_1 + f_{hp}^{n+1} w_2\right) d\boldsymbol{x} \\
&+ \int_D \left(\lambda c^{n+1} w_1 + \lambda_{hp} c_{hp}^{n+1} w_2\right) d\boldsymbol{x} \\
&+ \int_D \left(\alpha_1 c^{n+1} c_{hp}^{n+1} w_1 + \alpha_2 c^{n+1} c_{hp}^{n+1} w_2\right) d\boldsymbol{x} = 0,
\end{aligned} \tag{8}$$

where, w_1 and w_2 are test functions for the unknown variables c and c_{hp}, respectively.

Ashkan Haji Hosseinloo received the B.Sc. from Amirkabir University of Technology, Tehran, Iran, in 2009, the M.Eng. from Nanyang Technological University, Singapore, in 2013, and the Ph.D. from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2018, all in Mechanical Engineering. Ashkan is a postdoctoral scholar at MIT Laboratory for Information and Decision Systems (LIDS) and MIT Institute for Data, Systems, and Society (IDSS). His research lies at the intersection of machine learning and system dynamics & control, and is motivated by the urgent need to address the pressing issues of energy and environmental sustainability and social equity. Ashkan's work has addressed fundamental challenges in the control of complex dynamical systems with applications in structural dynamics, energy harvesting, and smart cities.

Saleh Nabi received the B.Sc. from K. N. Toosi University of Technology, Tehran, Iran, in 2005, the M.Sc. from Isfahan University of Technology, Isfahan, Iran, in 2008, and the Ph.D. from University of Alberta, Canada, in 2013, all in Mechanical Engineering. Saleh is a Principal Research Scientist at Mitsubishi Electric Research Labs (MERL). His research interests are at the intersection of fluid mechanics, scientific machine learning, dynamical systems, and optimal control in complex systems. His current research involves hybrid methods using traditional tools along with deep learning-based methods for efficient and robust control and estimation of PDEs with applications to HVACs and atmospheric LiDARs.

Anette (Peko) Hosoi received the B.A. degree from Princeton University, Princeton, NJ, USA, in 1992, and the M.Sc. and the Ph.D. degrees from the University of Chicago, Chicago, IL, USA, in 1994 and 1997, respectively, and all three degrees in physics. She is the Neil and Jane Pappalardo Professor of Mechanical Engineering and associate dean of engineering at the Massachusetts Institute of Technology, Cambridge, MA, USA. She is the co-founder of the MIT Sports Lab which connects the MIT community with pro-teams and industry partners to address data and engineering challenges in the sports domain. Hosoi's research interests include fluid dynamics, unconventional robotics, and bio-inspired design. She has received numerous awards including the APS Stanley Corrsin Award, the Bose Award for Excellence in Teaching, and the Jacob P. Den Hartog Distinguished Educator Award.

Munther A. Dahleh received the B.S. degree from Texas A&M University, College Station, TX, USA, in 1983, and the Ph.D. degree from Rice University, Houston, TX, USA, in 1987, both in electrical engineering. He is the William A. Coolidge Professor with the Massachusetts Institute of Technology, Cambridge, MA, USA, where he is also the Director of the Institute for Data, Systems and Society. Dahleh is a co-recipient of four George S. Axelby Outstanding Paper Awards. He is internationally known for his fundamental contributions to robust control theory, computational methods for controller design, the interplay between information and control, the fundamental limits of learning and decision in networked systems, and the detection and mitigation of systemic risk in interconnected and networked systems.