Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Constructing Markov matrices for real-time transient contaminant transport analysis for indoor environments

Anthony D. Fontaninia, Umesh Vaidyab, Baskar Ganapathysubramaniana,*

a Department of Mechanical Engineering, 2100 Black Engineering, Iowa State University, Ames, IA 50010, USA
b Department of Electrical and Computer Engineering, 2215 Coover, Iowa State University, Ames, IA 50010, USA

Abstract

Predicting the movement of contaminants in the indoor environment has applications in tracking airborne infectious disease, ventilation of gaseous contaminants, and the isolation of spaces during biological attacks. Markov matrices provide a convenient way to perform contaminant transport analysis. However, no standardized method exists for calculating these matrices. A methodology based on set theory is developed for calculating contaminant transport in real-time utilizing Markov matrices from CFD flow data (or discrete flow field data). The methodology provides a rigorous yet simple strategy for determining the number and size of the Markov states, the time step associated with the Markov matrix, and calculation of individual entries of the Markov matrix. The procedure is benchmarked against scalar transport of validated air flow fields in enclosed and ventilated spaces. The approach can be applied to any general airflow field, and is shown to calculate contaminant transport over 3000 times faster than solving the corresponding scalar transport partial differential equation. This near real-time methodology allows for the development of more robust sensing and control procedures of critical care environments (clean rooms and hospital wards), small enclosed spaces (like airplane cabins) and high traffic public areas (train stations and airports).

1. Introduction

High indoor air quality (IAQ) is essential for safety and comfort within the indoor built environment. Poor IAQ may lead to a loss in productive work time in commercial environments, expensive mechanical systems maintenance, and even litigation [2]. Gasses (carbon dioxide, volatile organic compounds, carbon monoxide, and radon) and airborne particulates (mold, atmospheric particulate matter, and microbial contaminants) are generally known to negatively affect IAQ. Control of these contaminants is critical in healthcare environments, clean rooms, and confined spaces.

For instance, incidence of burn wound infections remains a high-risk proposition even today when the burn area covers a substantial portion of the patient’s body. Controlling airborne contamination is of major importance in these cases. The susceptibility of patients is partially due to infections (from the large amount of exposed area) and the unique environmental conditions (the required elevated temperatures and humidity in the patient room) that such rooms operate under. Furthermore, the biological quality of air in hospital environments is of particular concern as patients may serve as a source of pathogenic microorganism to staff and hospital visitors in addition to other admitted patients. A classic example of this problem is the tuberculosis outbreak in 2000 [25].

Outside of healthcare facilities, confined spaces are also susceptible to the transmission of infectious diseases. Public transportation (especially aircraft and underground subway systems), schools classrooms, and close office quarters provide an environment where a single infected individual can potentially infect a large number of people. Examples include the spread of influenza [26] and severe acute respiratory syndrome (SARS) [30] that have been reported in aircraft travel. Other examples include the gas attack on the Tokyo subway system in 1995 [31], and the measles outbreak in office spaces in 1985 [7]. In US schools, students miss approximately 38 million school days each year due to the influenza virus [8].

Motivated by these issues, it is clear that methods for prediction and control of these contaminants in indoor environments are...
needed. In this context, Computational Fluid Dynamics (CFD) simulations have been used extensively for determining airflow in indoor environments. The earliest reference to CFD being used for this purpose was the study of the velocity characteristics of ventilated rooms by P.V. Nielsen et al., in 1978 [27]. Since then the popularity of running CFD simulations for predicting air flow in indoor spaces has increased rapidly. ASHRAE has taken efforts to standardize CFD simulation procedures, i.e. to setup, simulate, and perform both verification and validation [1]. CFD simulations have also been integrated into building simulation programs like EnerGyPlus [37] and ESP-r [16] for establishing boundary conditions of buoyancy based fluid flow. CFD flow fields have also been used for developing methods for scalar-transport in indoor environments. Validation for multi-zone CFD simulations with contaminant transport has been performed by building researchers [35,40]. Other validations of scalar contaminant transport have been performed for ventilation efficiency [3], personal ventilation [22], occupant exposure [23] and dispersion [21] of toxic contaminants, and dispersion of contaminants in hospital wards [15]. The application of contaminant dispersion in the indoor environment applied to emergency situations was investigated by L. Wang et al. [41]. Many other application of contaminant transport in the indoor environment can also be found in scientific literature. A few examples include identification of sources using an inverse CFD approach [44], removal of contaminants using different ventilation strategies [17], and contaminant transport in an airliner by moving bodies [32].

In the previous discussed work, contaminant transport simulations by solving the scalar transport equations have been shown to produce accurate results, but many limitations exist to utilizing these methods for real-time prediction and control of indoor environments. Solving the scalar transport equations involves loading a potentially large CFD data and mesh into a computer’s memory, and then solving a partial differential equation (PDE) for the transient contaminant transport. Buildings with a complex geometry could require specialized hardware simply to load the CFD data and mesh into memory. Specialized software packages or commercial software may also be needed to perform the transient contaminant transport simulation. The computational time and memory requirements are too much for wireless sensors designed for a long battery life to manage for real-time decisions.

There have been some promising advances that enable simulating fluid behavior in real time. These include Fast fluid dynamics, lattice Boltzmann methods, and Markov methods. Fast fluid dynamics (FFD) [36] uses a set of low order schemes and a semi-lagrangian approach for advection to reduce computing cost. Comparisons of FFD and CFD along with implementation of FFD on graphical hardware with promising results have been recently accomplished by Zuo et al. [45,46]. The lattice Boltzmann method uses kinetic theory to incorporate physics of micro/mesoscopic processes on the behavior macroscopic modules [9]. Lattice Boltzmann methods have been popular due to their ability for parallelization and implementation on graphical processing units (GPUs) [4,42]. The third alternative is Markov matrices methods that are the focus of the current work.

Markov matrices provide an efficient and elegant approach to modeling the contaminant transport problem. This was explored in the recent seminal work by Chen and co-workers for CFD data [10,11] and Nicas for a multi-zone model [20]. These methodologies use either Lagrangian particle tracking or a contaminant flux to calculate entries of the Markov matrix. The Markov matrix is used to propagate concentrations of contaminants through time. The simulation results using the Markov method compared well with experiments and CFD simulations [10,11]. The Markov method is extremely fast compared to Eulerian transport, as only a single matrix-vector multiplication is needed to perform the transient contaminant transport. Due to the speed of this method, real-time inference of the contaminant field can be performed. Furthermore, by utilizing sparse matrix storage this method drastically reduces the memory requirements needed to perform the analysis.

Analyzing the transport of contaminants using Markov matrices has some additional benefits over other fast simulation techniques. Not only can Markov matrices be used for forward propagation of contaminants, but they can also be used for inferring the positions of the contaminants at previous times. The Markov matrix can be used to develop systematic optimization-based procedure for the optimal locations of actuators and sensors for the sensing and control of contaminants [34,39]. Furthermore, the linear property of a Markov matrix can also be used to develop a systematic procedure using linear system theory for the optimal control and estimation of contaminant [33,38]. Besides control and estimation applications, the spectral properties of the Markov matrix provides information about the long-term concentration profiles of contaminants in the space and the amount of time contaminants stay at a given position. Thus, utilization of the Markov matrix for flow field analysis and contaminant transport can be used to get more insight than the standard scalar transport methodology. An approach dual to the Markov approach using finite dimension approximation of Koopman operator is proposed by M. Georgescu et al. [14] for reducing order modeling in building systems.

While the Markov method is very promising, some aspects of the procedure to calculate the Markov matrix are still unclear. The Markov matrix is based on a time step that allows scalar density evolution in a flow field. Fig. 1. While previous work Chen et al. [10] emphasizes the importance of the time step, $\Delta t$, associated with the Markov matrix, there are no rigorous rules available for determining this time step. Other open parameters are the size of the Markov states, $h$ in Fig. 1, and the number of streamlines needed to construct an accurate Markov matrix. Establishing guidelines for determining these parameters can help bring the utility of Markov matrices closer to practice. A standard methodology will provide engineers and researchers an enabling framework for real-time prediction of transient contaminant transport in indoor environments.

In this paper, we develop a rigorous framework for estimating the parameters involved in the construction of the Markov matrix from CFD data. We present a data driven approach to determine the size of the Markov states. We then give simple formulas that provide bounds for the time step associated with the Markov matrix,

![Fig. 1. Motion of a particle and a discrete volume in an airflow field.](image-url)
and the ideal time step based on the flow field and Markov state discretization. A set theory approach is utilized to calculate the entries of the Markov matrix that satisfies the contaminant flux from one state to another. We also provide a method for determining the number of particle traces used in the set theory approach to calculate accurate Markov matrices. Finally we present a flow chart that clearly describes the inputs, the calculation procedure of the Markov matrix, and steps for performing contaminant transport. The results of this paper better define the methodology for using Markov matrices for real-time prediction of transient scalar transport. This methodology opens the door for real-time control, sensing of contaminants, and source identification effecting air quality of indoor environments.

2. Example flow fields

While the developed methodology is generic and can be applied to any arbitrary flow field, we utilize two example flow fields to illustrate the methodology. The two flow fields are the Rayleigh–Bénard convective instability problem and the IEA annex 20 isothermal problem [19] initially examined by P.V. Nielsen et al., in 1978 [27]. The two flow fields were chosen because one (the Rayleigh-Bernard problem) describes a fluid flow that is driven predominately by buoyancy forces, and the other (the IEA-annex-20 problem) is predominately driven by inertial forces. These two choices represent the relative extremes of flow fields seen in indoor environments, from laminar buoyancy driven flow to turbulent inertial driven flow. Details of each flow field are discussed separately in the next sub-sections.

2.1. Rayleigh–Bénard convective instability problem

The Rayleigh–Bénard convective instability problem consists of an enclosed cavity. Fig. 2. The bottom boundary is heated, the top boundary is cooled, and the two side boundaries are perfectly insulated. The boundaries cause the fluid near the top boundary to fall and the fluid near the bottom boundary to rise. The resulting steady flow field is two counter rotating convective cells, Fig. 2. The flow field was simulated using a CFD simulation with a domain discretization of 100 uniform cells in the x-direction, and uniform 50 cells in the y-direction. Governing equations solved on the domain were the laminar incompressible form of the Navier–Stokes and energy equations. Buoyancy was introduced using the Boussinesq approximation for density. The Rayleigh number, $Ra$, for this flow field is 4840. The governing equations were solved to a residual tolerance of 1e-6 for all the solution variables. Although not shown here, validation of the fluid flow model was performed for the Nusselt number on the boundaries for a cavity with an aspect ratio $1 \times 1$ and $1 \times 2$. The Nusselt number for laminar flow ($10^2 \leq Ra \leq 10^5$) agreed with the empirical correlation developed by G. Barakos et al. [5].

2.2. IEA annex 20-2D isothermal benchmark

The IEA-annex-20 problem is a ventilated cavity with the inlet on the top section of the left wall and an outlet on the lower right wall. The dimensions of the inlet and outlet relative to the height are $l_{inlet}/H = 0.056$ and $l_{outlet}/H = 0.16$ respectively. The air at the inlet stays along the ceiling until the air reaches the far right wall. The air in the ventilated space creates one large recirculation zone. The flow field was also simulated using a CFD simulation. The domain was discretized into 112 non-uniform cells in the x-direction, and 40 non-uniform cells in the y-direction, Fig. 3. The governing equations solved on the domain were the incompressible form of the Reynolds averaged Navier–Stokes (RANS) equations. The turbulence model chosen was the two-equation RNG k-ε model. The Reynolds number is 5000 based on the inlet dimension for the characteristic length. The governing equations were solved to a residual tolerance of 1e-6 for all the solution variables. Solutions of the flow field were compared according to the quantities specified by the benchmark (U-velocity at $x/H = 1$ and 2, and $y/H = H/2$ and $H – l_{inlet}/2$). Although not shown here, data produced

![Fig. 2.](a) Contour of the air speed for the Rayleigh–Bénard problem with streamlines showing the counter rotating convection cells. (b) The temperature $(\theta = |T – T_{cold}|)/|T_{max} – T_{cold}|$ distribution in the heated cavity.

![Fig. 3.](a) Contour of the normalized air speed in the domain with streamlines showing the large recirculation zone. (b) The nonuniform discretization (112 x 40 elements) used to solve the governing equations.
by the simulations agreed well with the published values [19].

2.3. Discussion of the example flow fields

The flow fields used are based on benchmark CFD problems with a lot of supporting literature. The flow fields are steady state flow fields calculated on structured (uniform for case 1, non-uniform for case 2) CFD meshes. Although the flow fields are steady state, the methodology developed in this paper is trivially extended to transient flow fields. This paper explains how to construct a single Markov matrix from a single flow field. In a transient flow field, the same process is used to construct a Markov matrix corresponding to each snapshot of the transient flow field.

We emphasize that the methodology is very general and can be used on both structured and unstructured CFD meshes for arbitrarily complicated geometries.

3. Methods

3.1. Mathematics of Markov matrices and scalar transport

In order to predict the motion of contaminants in indoor environments, the movement of a particle or a discrete volume of the contaminant is tracked through time, Fig. 1. Given the current position of a particle in an airflow field, the particle will move according to the air velocity at its current position. The underlying airflow field governs the motion of a contaminant at each time instant.

Any indoor environment can be discretized into a set of cells \( \{ \omega_1, ..., \omega_n \} \), Fig. 4. We also refer to these cells as states. If only one particle is placed at the center of the state the particle represents the entire volume of the state (state 3, 6, and 7 in Fig. 4). Then by tracking each particle for a period of time, \( \Delta t \), both the starting state and destination state are easily determined. Since there is only one particle per cell, the transition is binary. Depending on whether the particle has left the state, the state either experiences a transition or remains in the initial state. In this case state 3 transitioned to state 6, state 6 stayed in the initial state, and state 7 transitioned to state 9. The result of these transitions are expressed in a transition matrix, \( P_3 \), Equation (1), where \( P_{ij} \) denotes the probability of a particle placed in state \( i \) to transition to state \( j \). In this example the entries of the transition matrix would be \( P(3, 6) = 1, P(6, 6) = 1, \) and \( P(7, 9) = 1 \) for initial states 3, 6, and 7 respectively.

If a set of particles are uniformly placed in a state, each particle represents a fraction of the volume of the state, see state 1 in Fig. 4. If multiple particles are used then a fraction of those particles after some time \( \Delta t \) may end up in different cells, see state 1 in Fig. 4. Thus, the transitions for state 1 become probabilistic. In this example state 1 transitions with a probability of \( P(1,:) = [2/9, 1/9, 0, 2/9, 2/9, 0, 1/9, 1/9, 0] \) for each state 1 through 9. This process can be repeated for each state in the indoor environment. Since the transitions for a given state are probabilistic, the matrix in Equation (1) also has the following properties. All entries of the matrix, \( P \), are positive (Equation (2)) and the row sum of the matrix is equal to one (Equation (3)), i.e. row stochastic. A matrix with these properties is called a Markov matrix. The entries of the Markov matrix are often fairly sparse or limited to a relatively small bandwidth. In order to minimize the memory requirements for the Markov matrix, a sparse matrix format is used. In sparse matrix format only the nonzero entries are stored, thus minimizing the memory used to store the Markov matrix.

\[
P = \begin{bmatrix}
P_{11} & P_{12} & \cdots & P_{1n} \\
P_{21} & P_{22} & \cdots & P_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
P_{n1} & P_{n2} & \cdots & P_{nn}
\end{bmatrix}
\]  

(1)

\[
P_{ij} \geq 0 \quad \text{for all } i,j
\]  

(2)

\[
\sum_{j=1}^{n} P_{ij} = 1 \quad \text{for all } i
\]  

(3)

The underlying airflow field that governs the motion of the particles or contaminants is based on the passive scalar equations for contaminants less than 3 \( \mu \) m. For larger airborne particles more complicated particle transport models can be utilized [10]. Traditionally, scalar transport in an Eulerian frame is performed by solving an advection diffusion partial differential equation (PDE), Equation (4). The solution process involves iteratively solving a linear system of equations at each time step. The solution of Equation (4) is often fairly computationally expensive in the context of performing the calculations in real-time.

\[
\frac{\partial \varphi}{\partial t} + \nabla \cdot (U \varphi) + \nabla^2 (D \varphi) = S_e
\]  

(4)

In contrast to solving Equation (4), transient scalar transport utilizing Markov matrices is fairly easy to calculate. The Markov matrix, \( P \), serves as a finite dimensional approximation of the advection-diffusion partial differential equation on the finite partition of the Markov state. Let \( \mu \) be \( \{ \mu_1, \ldots, \mu_n \} \in \mathbb{R}^n \) be the discretization of the contaminant density function \( \varphi(x, y, z, t) \), Equation (5), \( \mu^k \) is the averaged cell value of the contaminant in the Markov cell \( \omega_k \) at time \( t \), and is mathematically written as:

\[
\mu^k_t = \frac{1}{V_{sk}} \int \int \int_{\omega_k} \varphi(x, y, z) dV, \quad k = 1, \ldots, n
\]  

(5)

The evolution of \( \mu_t \) is governed by following linear system, Equation (6). The process of evolving the contaminant only consists of a single matrix vector multiplication operation.

\[
\mu_{t+\Delta t} = \mu_t P
\]  

(6)

We compare the difference between the solving the advection diffusion equation and the Markov method. The difference between the methods is quantified by the \( L_2 \) Euclidean error, Equation (7),
Based on these mathematical preliminaries, the construction of Markov matrices requires insight into three main questions: How to choose the number of Markov states (or Markov state size, \( h \))? How to choose the time step associated with the Markov matrix, \( \Delta t \)? And, given these two answers, how to accurately calculate entries of the Markov matrix by discretizing each state into a finite number of particles, \( n_{pts} \)? In the following sections, each question is addressed individually, and we develop easy to use formulae to estimate these for a general flow field.

3.2. Determining the number of Markov states

Choosing number of states is a balance between accuracy and computational efficiency. If the number of states is too small, the coarse Markov matrix will smooth out characteristics of the airflow field. If the number of states is too large, the computational time to compute (and use) the Markov matrix increases. As the number of states increase, the computational effort to propagate the contaminant a single time step becomes larger. As the simulation time increases, the ability to run real-time prediction of the contaminant transport may be compromised. Therefore, it is necessary to choose an optimal number of states that balances both speed and accuracy.

For very coarse approximations of the underlying air flow field, the distribution of air speeds in the space is very poor. See, for example, the 04 \( \times \) 08 distribution compared to the 64 \( \times \) 32 distribution in Fig. 5. As the number of cells increases different velocity scales of the air flow field become resolved. As this occurs, the distribution of the air speeds in the space starts to converge. See, for example, the 32 \( \times \) 16 distribution and the 64 \( \times \) 32 distribution in Fig. 5. Using this concept of the standard deviation, Equation (8), of the air speeds resolved by the Markov matrix should provide insight to when the spatial scales of the problem have been resolved. The change in the standard deviation as the size of the Markov states decreases can be quantified by a simple expression, Equation (9). The refining of the Markov cells continues until the \( \varepsilon_{std} \) drops below some tolerance. Based on the experience of the authors, a value of 1% seems to be sufficient for these problems. Larger or smaller values of the air speed standard deviation may be sufficient for other problems. We observe more numerical diffusion with larger thresholds of air speed standard deviations and lower numerical diffusion with smaller thresholds of air speed standard deviation.

\[
\varepsilon_{error}(t) = \left( \frac{\sum_{k=1}^{n} (\psi^k_P - \psi^k_f)^2}{\sum_{k=1}^{n} (\psi^k_f)^2} \right) \times 100
\]  

(7)

\[
\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left( \|U\|_{i} - \langle \|U\| \rangle \right)^2}
\]  

(8)

Fig. 5. Left) average air vector in each state for three different Markov matrix resolutions for the Rayleigh–Bénard problem. Right) the corresponding probability density functions of the air speeds represented by the Markov matrix.
The maximum number of states is bounded by the number of cells in the CFD data or the experimental flow field (for uniformly discretized data). For unstructured meshes the size of the Markov states is bounded by the smallest element in the CFD mesh. As the size of the Markov states reaches the size of the data of the underlying flow field, the Markov matrix resolves all the same scales of the data. If the size of the Markov states is decreased further, the air speeds produced for the Markov cells are interpolated from the underlying data.

### 3.3. Determining the time step for the Markov transition matrix

The time associated with the Markov matrix is a crucial parameter for contaminant transport. If the time step is too small, there may not have enough time for a particle to leave the state, see state 6 in Fig. 4. The states that have not transitioned are called absorbed states; because once the contaminant enters the state the contaminant cannot leave that state. If the time step is too long, then the particle may jump over a neighboring state to another, see state 7 in Fig. 4. These cases result in information loss when calculating the Markov matrix. Alternatively, for accurate contaminant transport the Markov time step should be chosen such that the number of particles that have transitioned to neighboring cells is maximized.

These concepts are explored for a given flow field by placing a particle at the center of the Markov state like state 3, 6, and 7 in Fig. 4. These particles are tracked to see when they transition from their initial state to their nearest neighbors, and from their nearest neighbors to other cells. The nearest neighbors are cells that are directly connected to the initial state. For example the nearest neighbors of cell 3 in Fig. 4 are cells 2, 6, and 5. The nearest neighbors of state 2 in Fig. 4 are states 1, 5, 3, 4, and 6. The nearest neighbors of state 5 in Fig. 4 are all the state 1 through 9.

If the time step is extremely small, all the states are absorbed states. At some minimum time, one particle will leave the starting cell. This time, \( \Delta t_{\text{min}} \), is the minimum time step that can be used for the Markov matrix. The minimum time is related to half the size of the Markov state divided by the largest air speed, Equation (10). This departure from a fully absorbed Markov matrix is displayed in Fig. 6. As the time step is increased from \( \Delta t_{\text{min}} \), more and more particles transition into their nearest neighbor and the number of absorbed states continue to drop, Fig. 6.

\[
\Delta t_{\text{min}} = \frac{1}{2} \frac{h}{\|U\|_{\text{max}}}
\]  

Initially as the number of absorbed states drop, the number of cells that have transitioned to its nearest neighbor starts to increase as more particles leave their initial state, Fig. 6. At some choice of ideal time step, \( \Delta t_{\text{ideal}} \), the number of particles in their nearest neighbors peaks before the particles transition to states that are further away (like state 7 in Fig. 4). The total distance a particle needs to move to leave the neighboring state is \( l_{\text{max}} = h/2+h = 3/2h \). But considering the spread in air speeds possible in the domain, one half of this maximum distance is used as an average \( l_{\text{ave}} = l_{\text{max}}/2 = 3/4h \).

\[
\Delta t_{\text{ideal}} = \frac{l_{\text{ave}}}{\|U\|} = \frac{3}{4} \frac{h}{\|U\|_s}
\]

As the choice of the time step is further increased, particles start to transition out of their nearest neighbors and into the rest of the domain. The percentage of particles in their nearest neighbors may not decrease to 0 but to some small tolerance \( \epsilon_{\text{tol}} \), Fig. 6.² The time at which \( \eta_{\text{abs}}(t) = \Delta t_{\text{ideal}}/n \) drops below \( \epsilon_{\text{tol}} \) is the maximum time step that can be used for the Markov matrix. In order to calculate the maximum time step a characteristic air speed and length are needed. The characteristic air speed is the \( \epsilon_{\text{tol}} \times 100 \) percentile of the average air speeds in the Markov states, Equations (12)–(14) [29]. The \( \epsilon_{\text{tol}} \times 100 \) percentile air speed is calculated by determining the indices \( a \) and \( a+1 \) in an ordered list of average air speeds of the Markov states with an interpolation weight \( b \) between the two indices.

\[
a = \lfloor \epsilon_{\text{tol}}(n-1) + 1 \rfloor
\]

\[
b = a - \epsilon_{\text{tol}}(n-1) + 1
\]

\[
\|U\|_s = \|U\|_a + b(\|U\|_{a+1} - \|U\|_a)
\]

The characteristic length is the maximum distance the particles need to travel, \( l_{\text{max}} \). The maximum time step for the Markov matrix is expressed in Equation (15).

\[
\Delta t_{\text{max}} = \frac{3}{2} \frac{h}{\|U\|_s}
\]

Equation (10) and Equation (15) determined bounds for the time step associated with the Markov matrix, while Equation (11) describes the ideal time step for the Markov matrix. The upper bound, lower bound, and ideal time step are all based on the size of the Markov states and the underlying airflow field.

### 3.4. Set theory approach for calculation of Markov matrices

Set theory based numerical methods have been developed for the construction of Markov matrices or more generally for the finite dimensional approximation of linear transfer Perron–Frobenius operators [12,13]. We leverage these ideas to develop a systematic data-driven method for the construction of the Markov matrix from CFD data. Each state of the Markov matrix represents a discrete element area in 2D or volume in 3D. Fig. 7 displays the

² The tolerance may never go to zero, because of recirculation patterns in the domain.

---

Fig. 6. Graphical explanation of the evolution of the number of absorbed states and the number of states that have transitioned to their nearest neighbor as the time step is increased. The points where the minimum, ideal, and maximum time steps are calculated are also displayed.
transformation of state 11 for a time step $\Delta t$. For this example, each boundary of state 11 is discretized into 4 segments (5 particles). These particles on the edges of the states are evolved through time according to the underlying flow field. After the transformation of the element, the entries of the Markov matrix are calculated by taking the intersection between each state and the transformed area or volume. Fig. 7. Fig. 7 shows how the area of transformed state 11 is intersected with the states 11, 12, 19, and 20. The percent of the area or volume found in each state with respect to new transformed area or volume provides the information necessary to assign values to the Markov matrix. Since the transformed element initiated from state 11, Fig. 7, the row of the Markov matrix being filled is row 11. Approximately 43% of the area of the transformed state ended up in state 19. Therefore the Markov entry row 11 and column 19 is 0.43 (P(11,19) state ended up in state 19. Therefore the Markov entry row 11 and column 19 is 0.43 (P(11,19)

Fig. 7. The transformation of state 11 for a timestep $\Delta t$ into some of the nearest neighbors of state 11. The percentages represent the percent of the transformed area in each state.

Fig. 7.

The number of particles that start to leave their initial Markov state, Fig. 11. The major steps in the procedure are numbered 1 to 6. Step 1 provides the inputs necessary for the analysis and where the inputs are used to construct the Markov matrix. Steps 2, 3, and 4 describe how to calculate the parameters for the Markov matrix. Step 5 is responsible for the calculation of the Markov matrix. Step 6 details the transient contaminant transport calculation.

5. Results

5.1. Choosing the number of states

The number of states in the Markov matrix is based on the convergence of the standard deviation of the air speed in each Markov state. For six different sizes of the Markov states the (non-dimensional) air speed standard deviation is plotted for both the Rayleigh-Bénard and the IEA-annex 20 problems, Fig. 9(a). The standard deviation of the non-dimensional air speed increases as the cell size decreases for both example problems. Once the airflow structures are reasonably resolved, the rate of increase of the non-dimensional air speed standard deviation asymptotically converges.

The convergence of the non-dimensional air speed standard deviation can be seen in Fig. 9(b). As size of the Markov states decrease, the absolute difference ($L_1$ error) between the previous resolution (i-1) and the current resolution (i) decreases. Fig. 9(b) shows that for values of $H/h = 32$ for both the Rayleigh-Bénard and IEA annex 20 problem, the non-dimensional air speed standard deviation converges below the desired tolerance of 1%. This value of $H/h$ corresponds to a Markov state discretization of 64 x 32 cells for the Rayleigh–Bénard problem, and 96 x 32 cells for the IEA annex 20 problem. The Markov state discretizations of both domains can be seen in Fig. 10. Total number of states for the Rayleigh–Bénard problem is 2,048, and for the IEA annex 20 problem the number of states is 3072.

5.2. Choosing the Markov time step

Now that the discretization of the Markov states has been evaluated, the time step associated with the Markov matrix can be calculated. Initially a particle is generated at the center of each Markov state. The paths of the particles are tracked through time. For each discrete time instant the number of absorbed states, Fig. 11, and the number of states that have transitioned to their nearest neighbors, Fig. 12, are counted. Initially, the results show that the Markov states are absorbed states, but as the time step is increased the particles start to leave their initial Markov state, Fig. 11. The noisy section at the end of the Rayleigh-Bénard absorbed state plot...
is due to eventual recirculation of the particles back into their initial state, Fig. 11(a).

Once the particles start to leave their initial states, the particles enter to one of their nearest neighbors, Fig. 12. The peak of the curves (Fig. 12) maximizes the number of the particles in their neighboring states, which corresponds to the ideal time step for the Markov matrix. After the peak, a number of particles start to leave the neighboring states into other states in the domain. Once again for the Rayleigh–Bénard problem the noise is due to the states returning to their nearest neighbors after a long time step. The absorbed state curves and the nearest neighbor curves in Figs. 11 and 12 are clearly a function of the Markov state size. As the
Markov state size is decreased each of the cures are shifted to the left. Therefore as the Markov state size is decreased, the time steps chosen should also decrease. Also notice that for very large (coarse discretization) Markov state sizes the curves seem to have discontinuous jumps. As the number of states are increased by reducing the size of the states, the curves become smoother. This relationship shows that very coarse Markov states may have an unsatisfactory performance.

Comparison of the data-driven choice of time step with the proposed heuristic equations: Based on the curves in Figs. 11 and 12, the minimum, ideal, and maximum time step for the Markov matrices were computed for each Markov size. These data are plotted in Fig. 13, and labeled as “Data.” The tolerance chosen for the maximum time step was 0.2 and 0.07 for the Rayleigh-Bénard and the IEA annex 20 problems respectively. The result of Equations (10), (11) and (15) for are plotted in Fig. 13, and are labeled “Mesh.” The equations compare very well for the minimum, ideal, and maximum time steps calculated by particle tracking. Therefore, the equations provide fairly accurate approximations to the Markov time step. Since the equations based strictly on the Markov state size and the flow field, the time step chosen for the Markov matrix can be calculated as a pre-processing step. Using the discretization recommendations in the previous section, the time steps, $\Delta t U_{\text{max}}/H$, for the Rayleigh-Bénard and the IEA annex 20 problem are 0.0736 and 0.1199 respectively.

5.3. Choosing the number of points along the edges of each state

The deformation metric, Equation (17), was calculated for the Rayleigh–Bénard and IEA annex 20 problems based on the recommended Markov state discretization in the Section 5.1. The worst cell for the Rayleigh–Bénard problem for the chosen discretization was found to be the cell 2,004, which is the [20,32] cell in the x and y direction. The worst cell for the IEA annex 20 problem for the chosen discretization was found to be the cell 2,983, which is the [7,32] cell in the x and y direction. The number of points per edge of the cell that allowed the area error, Equation (18), to converge to 1% was 5 and 23 for the Rayleigh-Bénard and IEA annex 20 problems, respectively. Fig. 14 shows the worst cells transformed through the chosen timestep, and the number of points needed around the edge to allow the error, Equation (18), converge to 1%. The number of points needed for the IEA annex problem is higher because the shear at the boundary for turbulent flows is much higher than the laminar flow of the Rayleigh–Bénard problem.
5.4. Comparisons and error analysis

In the previous sections, the discretization of the Markov states, the time step of the Markov matrix, and the number of points per edge were calculated. Based on this information the Markov matrix was calculated for both the Rayleigh–Bénard and IEA annex 20 problems. To test the quality of the Markov matrices that have been calculated, a transient contaminant analysis comparison is performed between Eulerian transport, Equation (4), and the Markov based contaminant transport, Equation (6).

A qualitative comparison for both problems can be seen in Fig. 15 for the Rayleigh–Bénard problem and Fig. 16 for the IEA annex 20 problem. Qualitatively, the comparisons between the Eulerian transport method and the Markov method is very good for both problems. A quantitative comparison between the two methods was performed using the Euclidean error, Equation (7), Fig. 17, and Fig. 18. The quantitative comparison was performed for the chosen Markov discretizations for each problem (H/h = 32). The concentration from scalar transport and the Markov method for the Rayleigh–Bénard problem was compared at x/H = 0.25 and y/H = 0.5 for nondimensional time 2.01 and 4.02, Fig. 17. For the IEA annex 20 problem the concentration was for scalar transport and the Markov method was compared at x/H = 1.0 and x/H = 2.0 for nondimensional time 6.69 and 13.38, Fig. 18. For both Figs. 17 and 18, the Markov method compares very well with scalar transport. The largest differences are found at the boundary of the IEA annex 20 problem, which is most likely from the large amount of shear at the boundary.

The relative distance between the Eulerian transport method and the Markov method was also compared at each time instant produced by the Markov method. The number of contaminant propagations taken for the Rayleigh Bénard and the IEA annex 20 problems by the Markov method was 300 and 115 respectively. Overall the difference between the two methods stayed relatively small, Fig. 19. The maximum difference between the two methods for the Rayleigh–Bénard problem was about 16%, Fig. 19. The maximum difference between the two methods for the IEA annex 20 problem was about 11%, Fig. 19.

5.5. Real-time evaluation of the method

In order to ensure the Markov method for contaminant transport established in the paper produces real-time or faster than real-time estimates of the contaminant transport, the computational time to compute the contaminant transport was evaluated. Each problem was simulated for 1000 Markov time steps. The Rayleigh–Bénard problem took a simulation time of t U_{max}/H = 0.00864 to evolve the contaminant a total time of t U_{max}/H = 54.7. The IEA annex 20 problem took a simulation time of t U_{max}/H = 0.0315 to evolve the contaminant a total time of t U_{max}/H = 119.8. These tests used a 1.7 GHz Intel Core i7 processor. The Markov method was able to compute the contaminant transport 6665 times faster than an optimized PDE solver for the Rayleigh–Bénard problem and 3723 times faster than an optimized PDE solver for the IEA annex problem.

6. Discussion: implications of using the Markov matrices for contaminant transport

As the results shown in the previous section, Markov matrices provide a fast and relatively accurate method for calculating...
contaminant transport in indoor environments. The methodology can be applied to complex geometries and boundaries, as well as to transient flow fields. For transient simulations, each CFD time snapshot is converted into a Markov matrix using the developed methodology and by choosing the CFD timestep as the Markov timestep. This section is dedicated to briefly discuss other indoor air applications of Markov matrices for contaminant transport analysis. The focus is specifically on the benefits of spectral analysis,
placement of sensors, and performing state estimation using Markov matrices.

When designing an indoor airflow pattern for high IAQ, contaminants should not spend a lot of time in the occupied zone. Spectral analysis of the Markov matrix is useful to determine the long-term distribution of the contaminants and the time contaminants spend in the occupied space. In particular, the eigenvector with eigenvalue one of the Markov matrix carries information about the steady state or long-term asymptotic distribution of contaminant in the space. Similarly the other part of the spectrum of Markov matrix carry information about almost invariant region in the space where the contaminant spend most of the time before finally exiting these region [12].

Once the design of the airflow patterns is complete, sensors can be placed in the space to aid the HVAC system controls. The rows of the Markov matrix describe the probability of where the contaminants will go given a starting position, while the columns describe the origins of the contaminants at the previous time step. By finding the column with the maximum column support, a sensor

![Fig. 17. Comparison of the concentration between the scalar transport and the Markov methods along the a) x/H = 0.25 and the b) x/H = 2.0 lines in the domain for the Rayleigh–Bénard problem for two different times.](image)

![Fig. 18. Comparison of the concentration between the scalar transport and the Markov methods at vertical lines a) x/H = 1.0 and b) x/H = 2.0 in the domain for the IEA annex 20 problem for two different times.](image)

![Fig. 19. Quantitative comparison between the Eulerian transport method and the Markov method.](image)
placed in this column can sense the largest volume of the space [34].

Given a set of sensors that have been placed in the enclosed space, the next natural step is estimation of the contaminant field in the space at any given time. Here the linear property of the Markov matrices can be exploited to develop estimation techniques using linear system theory to determine the location or release of contaminant.

All these future directions are being explored and will be the topic of subsequent publications. We are also currently working on an open source platform to distribute to the community that will enable fast calculation of the Markov matrix, perform contaminant transport, and place the sensors.

7. Conclusions

This paper establishes a robust procedure for constructing accurate Markov matrices for real-time prediction of contaminant transport in enclosed spaces. This work formulates methods to determine the number and size of the Markov states, provides simple equations for the choosing the time step associated with the Markov matrix, and a procedure based on flow physics to determine the number of particles to represent the Markov elements. The methods provided in this work showed excellent agreement with the scalar transport methods well established in the field. An explicit procedure was articulated for application to other indoor airflow problems. For the two example problems the Markov method was able to predict contaminant transport at least 3000 times faster than the corresponding PDE based approach. Although the boundaries of the example problems are geometrically relatively simple, the method is naturally extended to complex boundaries, as well as to transient flow fields. The established methodology created in the paper has been applied to buoyant and inertial driven flow field. This procedure opens the door for real-time sensing and control of airborne infectious diseases, ventilation of indoor environments, and demand response and isolation of indoor environments, and demand response and isolation of airborne infectious diseases, ventilation of indoor environments, and demand response and isolation during biological attacks.

Acknowledgments

We thank the National Science Foundation for partial support (NSF CAREER 1149365) and computing support via XSEDE (CTS110007). We also thank the Iowa Energy Center for partial support (CTS110007). We also thank the Iowa Energy Center for partial support (CTS110007). We also thank the Iowa Energy Center for partial support (CTS110007). We also thank the Iowa Energy Center for partial support (CTS110007).
fast fluid dynamics model on graphics processing unit, Build. Environ. 45 (2010) 747–757.

Nomenclature

\( a \): number of sorted air speed values below the \( t_{\text{aol}} \times 100 \) percentile.
\( b \): A weighting factor to calculate the \( t_{\text{aol}} \times 100 \) air speed percentile
\( D \): diffusivity of the contaminant in the medium
\( H \): height of the domain
\( h \): the length and height of a square Markov state
\( h_{\text{inlet}} \): the length of the inlet for the IEA annex 20 problem
\( l_{\text{min}} \): minimum distance that a particle initially placed at the center of a Markov state needs to travel to leave the initial state
\( l_{\text{max}} \): maximum distance that a particle initially placed at the center of a Markov state needs to travel to leave the nearest neighbor of the initial state
\( l_{\text{ave}} \): average distance between \( l_{\text{min}} \) and \( l_{\text{max}} \)
\( n \): number of Markov states
\( n_{\text{det}} \): the number of Markov states
\( n_{\text{num}} \): the number of particles that have not transitioned from their initial state to one of their nearest neighbors
\( n_{\text{tot}} \): the number of points per edge to discretize the Markov states
\( n_{\text{steps}} \): the number of time steps taken for a simulation
\( N_{\text{Ra}} \): the Rayleigh number
\( t \): time
\( \Delta t \): time step associated with the Markov matrix
\( \Delta t_{\text{id}} \): the minimum time step associated with the Markov matrix
\( \Delta t_{\text{ideal}} \): the ideal time step associated with the Markov matrix
\( \Delta t_{\text{max}} \): the maximum time step associated with the Markov matrix
\( P \): Markov matrix
\( P_{ij} \): the \( i \)-th row and \( j \)-th column entry in the Markov matrix
\( S_{\epsilon} \): contaminant sources in the indoor environment
\( T \): temperature in the Rayleigh–Bénard problem
\( T_{\text{top}} \): Top surface temperature in the Rayleigh–Bénard problem
\( T_{\text{bot}} \): Bottom surface temperature in the Rayleigh–Bénard problem
\( \mathbf{U} \): air velocity vector
\( | \mathbf{U} | \): air speed
\( | \mathbf{U} |_{2} \): the \( i \)-th entry in the sorted (low to high) air speed from the average air speeds in each Markov cell
\( | \mathbf{U} |_{i} \): air speed in Markov state \( i \)
\( | \mathbf{U} |_{100} \): the \( t_{\text{aol}} \times 100 \) air speed percentile
\( | \mathbf{U} |_{\text{max}} \): maximum air speed in the domain
\( \langle | \mathbf{U} | \rangle \): average air speed for all the Markov states
\( u \): the \( x \)-component of the air velocity vector
\( \mathbf{V} \): volume of the transformed cell using \( i \) number of points per cell
\( V_{\text{air}} \): volume of the \( x_{k} \) state
\( \nu \): the \( y \)-component of the air velocity vector
\( \mathbf{w} \): the \( z \)-component of the air velocity vector
\( x \): the \( x \)-coordinate direction
\( y \): the \( y \)-coordinate direction
\( z \): the \( z \)-coordinate direction
\( \epsilon \): deformation measure for the velocity gradient tensor
\( \epsilon_{\text{def}} \): the \( L_{2} \) Euclidean error between the Markov method and the CFD solution
\( \epsilon_{\text{tol}} \): error in the Markov matrix due to discretization of the boundaries of each transformed state
\( \epsilon_{\text{std}} \): the error of the air speed standard deviation between two successive refinements of the Markov state size
\( \epsilon_{\text{tol}} \): tolerance to limit the maximum time step associated with the Markov matrix
\( \epsilon_{\text{desired}} \): the desired tolerance in the volume calculation of the transformed Markov state
\( \theta \): nondimensional temperature for the Rayleigh–Bénard Problem
\( \mu_{k} \): contaminant concentration in Markov cell \( k \) at time \( t \)
\( \mu_{k} \): contaminant concentration field at the current time step
\( \mu_{k+1} \): contaminant concentration field at the next time step
\( \sigma \): standard deviation of the air speed in the Markov states
\( \sigma(h_{i}/h_{j}) \): standard deviation of the air speed in the Markov states with Markov state size \( (h_{i}/h_{j}) \), \( i \) corresponds to the current refinement level
\( \sigma(h_{i}/h_{j+1}) \): standard deviation of the air speed in the Markov states with Markov state size \( (h_{i}/h_{j}) \), \( i \) corresponds to the previous refinement level
\( \psi \): the contaminant concentration from Eulerian scalar transport
\( \Omega_{k} \): the \( k \)-th Markov cell in the domain \( \Omega \)
\( \| \cdot \| \): magnitude of a vector
\( \langle \cdot \rangle \): Mean of a vector