2016

Analyzing the characteristics of contaminant transport using the Perron-Frobenius operator in indoor building environments

Anthony David Fontanini
Iowa State University

Follow this and additional works at: http://lib.dr.iastate.edu/etd
Part of the Mechanical Engineering Commons

Recommended Citation
Fontanini, Anthony David, "Analyzing the characteristics of contaminant transport using the Perron-Frobenius operator in indoor building environments" (2016). Graduate Theses and Dissertations. 15909.
http://lib.dr.iastate.edu/etd/15909

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
Analyzing the characteristics of contaminant transport using the Perron-Frobenius operator in indoor building environments

by

Anthony D. Fontanini

A dissertation submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Mechanical Engineering

Program of Study Committee:
Baskar Ganapathysubramanian, Co-Major Professor
Michael G. Olsen, Co-Major Professor
Ming-Chen Hsu
Alberto Passalacqua
Umesh Vaidya

Iowa State University
Ames, Iowa
2016

Copyright © Anthony D. Fontanini, 2016. All rights reserved.
DEDICATION

I dedicate this thesis to my parents Dave and Kathy, who have given me their love and support throughout this journey. I also dedicate this thesis to my wife Stephanie Fontanini who has helped push me to finish my degree.
# TABLE OF CONTENTS

## ACKNOWLEDGMENTS ........................................................................................................ v

## CHAPTER 1 INTRODUCTION .......................................................................................... 1

## CHAPTER 2 CONSTRUCTING MARKOV MATRICES FOR REAL-TIME TRANSIENT CONTAMINANT TRANSPORT ANALYSIS FOR INDOOR ENVIRONMENTS .................................................................................................................. 3

2.1 Abstract .......................................................................................................................... 3
2.2 Introduction ..................................................................................................................... 4
2.3 Example flow fields ......................................................................................................... 9
  2.3.1 Rayleigh–Bénard convection instability problem ..................................................... 9
  2.3.2 IEA annex 20-2D isothermal benchmark ................................................................. 10
  2.3.3 Discussion of the example flow fields .................................................................... 11
2.4 Methods .......................................................................................................................... 12
  2.4.1 Mathematics of Markov matrices and scalar transport .......................................... 12
  2.4.2 Determining the number of Markov states ............................................................. 15
  2.4.3 Determining the timestep for the Markov transition matrix .................................. 17
  2.4.4 Set theory approach for calculation of Markov matrices ..................................... 20
2.5 Flow chart ...................................................................................................................... 22
2.6 Results ............................................................................................................................. 23
  2.6.1 Choosing the number of states .............................................................................. 23
  2.6.2 Choosing the Markov timestep ............................................................................ 25
  2.6.3 Choosing the number of points along the edges of each state ............................ 28
  2.6.4 Comparisons and error analysis ........................................................................ 29
  2.6.5 Real-time evaluation of the method .................................................................. 33
2.7 Discussion: Implications of using Markov matrices for contaminant transport .......... 34
2.8 Conclusions .................................................................................................................... 35
2.9 Nomenclature ............................................................................................................... 36
2.10 References ................................................................................................................... 39

## CHAPTER 3 CONTAMINANT TRANSPORT AT LARGE COURANT NUMBERS USING MARKOV MATRICES .................................................................................................................. 42

3.1 Abstract .......................................................................................................................... 42
3.2 Introduction .................................................................................................................... 43
CHAPTER 5 QUANTIFYING MECHANICAL VENTILATION PERFORMANCE: THE CONNECTION BETWEEN TRANSPORT EQUATIONS AND MARKOV MATRICES

5.1 Abstract .................................................................................................................. 123
5.2 Introduction ............................................................................................................. 124
5.3 Methods .................................................................................................................. 126
  5.3.1 Mathematical preliminaries: contaminant transport using Markov matrices .......... 126
  5.3.2 Canonical form of absorbing Markov matrices ................................................. 128
  5.3.3 Quantifying measures of IAQ .............................................................................. 129
5.5 Flow field and validation ....................................................................................... 133
5.5.1 Air residual life time and age of air comparison .............................................. 136
5.5.2 Air residence time and ventilation effectiveness .............................................. 138
5.6 Discussion .............................................................................................................. 139
5.7 Conclusions ........................................................................................................... 142
5.8 Nomenclature ......................................................................................................... 143
5.9 References ............................................................................................................. 145
5.10 Appendix A: Timestep sensitivity of age of air ...................................................... 149

CHAPTER 6 CONCLUSIONS AND FUTURE WORK

Conclusions ................................................................................................................. 150
Future work .................................................................................................................. 150
ACKNOWLEDGMENTS

I would like to thank my committee co-advisors, Baskar Ganapathysubramanian and Michael Olsen, and my committee members, Umesh Vaidya, Alberto Passalacqua, and Ming-Chen Hsu, for their guidance and support throughout the course of this research.

In addition, I would also like to thank my friends, colleagues, the department faculty and staff for making my time at Iowa State University a wonderful experience.
ABSTRACT

People in the developed world spend the majority of their lives indoors. While inside, people can be exposed to many different gaseous and particulate contaminants. Both short term and long term exposure to these contaminants have a wide range of health related outcomes from mild discomfort to heart disease, cancer, and even death. The purpose of this research is to provide a computational framework for fast predictions of the transport of contaminants, which is used to identify optimal locations for sensors, and finally the framework is utilized to provide performance measures for ventilation system during the design stage. All these applications can be analyzed using the discrete form of the Perron-Frobenius operator, also known as Markov matrices. Two different methods are developed for fast and accurate calculations of contaminant transport at large Courant numbers. Once the Markov matrices are calculated, they are used to position sensors where contaminants are likely to collect and are used to quantify mechanical ventilation performance. The methods developed may be used to develop evacuation for people during a release of a toxic gas or airborne infectious disease, execute containment plans of potentially harmful airborne contaminants, and the control of HVAC systems to provide adequate ventilation and clean/fresh air to the occupied space.
CHAPTER 1. INTRODUCTION

The spread and movement of gaseous contaminants or particulate matter in an indoor environment is due to the underlying flow field produced by ventilation systems and disturbances by the movement of occupants in the space. In order to optimally place sensors to quickly detect the presence and predict the spread of these contaminants, some knowledge of the flow field is needed. In buildings with very large building zones, atriums, or many obstructions, computational fluid dynamics (CFD) can provide fairly accurate approximations to the flow field in a building zone. CFD is a fairly mature analysis tool in building simulations as it has been used to evaluate the thermal comfort, ventilation performance, and indoor air quality since the 1970’s. Spatial distributions of air speed, temperature, contaminant concentrations, and thermal comfort measures are available by using CFD where only bulk average performance measures and solution variables are calculated from multi-zone and energy calculation software. However, this extra information provided in CFD analysis does not come without a cost. The calculation of the air flow, temperature, turbulent quantities, contaminant, and performance measure fields is computationally intensive compared to multi-zone and energy calculation software. This work pursues methods that utilize the spatial information of CFD, but with less computational time are needed to predict the spread and contaminant fields in buildings.

The methods developed in this thesis focus on providing tools to evaluate designs, optimally place sensors, and accurately predicting the transport of contaminants in buildings in a computationally efficient manner. The framework allows for real-time predictions of where potential harmful contaminants may be headed in order to design, direct, and/or
modify evacuation strategies to limit the impact of an extreme event. Building upon these fast models efficient methods for optimally placing sensors are established to detect the presence of contaminants to set off alarms early during extreme events. While detecting and predicting the movements of contaminants are important, many problems can be mitigated during the design process by designing effective ventilation systems. Current computational tools are extended to provide a larger picture of the ventilation performance of systems during the design process. This work provides a platform for future work in creating sensor estimators, robust control of building ventilation systems with the information of the sensors and sensor estimators, and is easily extendible to design of these systems in the presence of uncertainty.

The organization of this thesis is as follows. Chapter 2 introduces a set-theory Lagrangian based method for contaminant transport. Chapter 3 introduces methods for calculating Markov matrices and inlet and volumetric scalar sources at large Courant numbers in time varying and time invariant flow fields. Chapter 4 shows how the information contained in the transition probabilities of the Markov matrix can be used for optimal placement of sensors in a building zone. Chapter 5 shows how absorbing Markov matrices can be used during the design process to quantify mechanical ventilation performance. Chapter 6 concludes by identifying areas and avenues of future research.
CHAPTER 2. CONSTRUCTING MARKOV MATRICES FOR REAL-TIME TRANSIENT CONTAMINANT TRANSPORT ANALYSIS FOR INDOOR ENVIRONMENTS

A paper accepted by *Building and Environment*

Anthony D. Fontanini, Umesh Vaidya, and Baskar Ganapathysubramanian

2.1 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 airflow fields in enclosed and ventilated spaces. The approach can be applied to any general airflow field, and is shown to calculate contaminant transport over 3,000 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).
2.2 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 [1]. 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 [2].

Outside of healthcare facilities, confined spaces are also susceptible to the transmission of infectious diseases. Public transportation (especially aircraft and underground subway systems), school 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 [3] and severe acute respiratory syndrome (SARS) [4] that have been reported in aircraft travel. Other examples include the gas attack on the Tokyo subway system in 1995 [5], and the measles outbreak in office spaces in 1985 [6]. In US schools, students miss
approximately 38 million school days each year due to the influenza virus [7].

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 [8]. 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 [9]. CFD simulations have also been integrated into building simulation programs like EnergyPlus [10] and ESP-r [11] 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 [12]. Other validations of scalar contaminant transport has been performed for ventilation efficiency [13], personal ventilation [14], occupant exposure [15] and dispersion [16] of toxic contaminants, and dispersion of contaminants in hospital wards [17]. The application of contaminant dispersion in the indoor environment applied to emergency situations was investigated by L. Wang et al. [18]. 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 [19], removal of contaminants using different ventilation strategies [20], and contaminant transport in an airliner by moving bodies [21].

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) [22] 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 [23], [24]. The lattice Boltzmann method uses kinetic theory to incorporate physics of micro/mesoscopic processes on the behavior macroscopic modules [25]. Lattice Boltzmann methods have been popular due to their ability for parallelization and implementation on graphical processing units (GPU)s [26], [27]. 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 [28], [29] and Nicas for a multi-zone model [30]. 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
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 [31], [32]. Furthermore, the linear property of Markov matrix can also be used to develop a systematic procedure using linear system theory for the optimal control and estimation of contaminant [33], [34]. Besides control and estimation applications, the spectral properties of the Markov matrix provide 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 Markov approach using finite dimension approximation of Koopman operator is proposed by [35] for reducing order modeling in building system.

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, figure 1. While previous work [28] 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 figure
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.

Figure 1: Motion of a particle and a discrete volume in an airflow field.

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, 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.
2.3 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 [36] initially examined by P.V. Nielsen et al. in 1978 [8]. 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.

2.3.1 Rayleigh-Bénard convective instability problem

The Rayleigh-Bénard convective instability problem consists of an enclosed cavity, figure 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, figure 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 length to 1 height. The Nusselt number for laminar flow ($10^3 \leq Ra \leq 10^5$) agreed with the empirical correlation developed by G. Barakos et al. [37].

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

### 2.3.2 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, figure 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 5,000 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 \(y/H = l_{inlet}/2\) and \(H - l_{inlet}/2\)). Although not shown here, data produced by the simulations agreed well with the published values [36].

![Figure 3: (a) Contour of the normalized air speed in the domain with streamlines showing the large recirculation zone. (b) The nonuniform discretization (112x40 elements) used to solve the governing equations.](image)

2.3.3 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.
2.4 Methods

2.4.1 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, figure 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 \{ω₁, ..., ωₙ\}, figure 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 figure 4). Then by tracking each particle for a period of time, Δt, 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, \( P_{ij} \), 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.
Figure 4: (Please view in color) Motion of a set of particles for a given time step $\Delta t$ that transition to different states in the Markov matrix.

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 figure 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 figure 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)

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

(2)

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

(3)

The underlying flow 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 [28]. 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.

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

(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 \( \mu_t = (\mu_t^1, ..., \mu_t^n)^T \in \mathbb{R}^n \) be the discretization of the contaminant density function \( \varphi(x, y, z, t) \), equation 5. \( \mu_t^k \) is the averaged cell value of the containment in the Markov cell \( \omega_k \) at time \( t \), and is mathematically written as:

\[ \mu_t^k = \frac{1}{V_{\omega_k}} \iiint_{\omega_k} \varphi(x, y, z) \, dV, \quad k = 1, ..., n \]  

(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.

$$\varepsilon_{error}(t) = \sqrt{\frac{\sum_{k=1}^{n}(\phi^k_t - \mu^k_t)^2}{\sum_{k=1}^{n}(\phi^k_t)^2}} \times 100$$ (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.

### 2.4.2 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 04x08 distribution compared to the
64x32 distribution in figure 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 32x16 distribution and the 64x32 distribution in
figure 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.

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\|U\|_i - \langle \|U\| \rangle)^2} \quad (8)$$

$$\varepsilon_{std} = \frac{|\sigma(H/h_i) - \sigma(H/h_{i-1})|}{\|U\|_{max}} \quad (9)$$

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.
2.4.3 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 figure 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 figure 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 figure 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 figure 4 are states 2, 6, and 5. The nearest neighbors of state 2 in figure 4 are states 1, 5, 3, 4, and 6. The nearest neighbors of state 5 in figure 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, $\Delta t_{min}$, 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 figure 6. As the time step is increased from $\Delta t_{min}$, more and more particles transition into their nearest neighbor and the number of absorbed states continue to drop, figure 6.

$$\Delta t_{min} = \frac{1}{2} \frac{h}{\|U\|_{max}}$$  \hspace{1cm} (10)
Figure 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.

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, figure 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 figure 4). The total distance a particle needs to move to leave the neighboring state is $l_{\text{max}} = \frac{h}{2} + h = \frac{3}{2} h$. But considering the spread in air speeds possible in the domain, one half of this maximum distance is used as an average $l_{\text{ave}} = l_{\text{max}}/2 = \frac{3}{4} h$.

$$
\Delta t_{\text{ideal}} = \frac{l_{\text{ave}}}{\|u\|} = \frac{3}{4} \frac{h}{\|u\|}
$$

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 $\epsilon_{\text{tol}}$, figure 6.\footnote{The tolerance may never go to zero, because of recirculation patterns in the domain.} The time
at which $n_{\text{net}}[t > \Delta t_{\text{ideal}}]/n$ drops below $\varepsilon_{\text{tol}}$ 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 $\varepsilon_{\text{tol}} \times 100$ percentile of the average air speeds in the Markov states, equations 12 - 14 [38]. The $\varepsilon_{\text{tol}} \times 100$ 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.

$$a = \lfloor \varepsilon_{\text{tol}}(n-1)+1 \rfloor$$

$$b = a - \varepsilon_{\text{tol}}(n-1)+1$$

$$\|U\|_{\varepsilon} = \|U\|_{a} + b(\|U\|_{a+1} - \|U\|_{a})$$

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.

$$\Delta t_{\text{max}} = \frac{3}{2} \frac{h}{\|U\|_{\varepsilon}}$$

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.

2.4.4 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 [39], [40]. 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. Figure 7 displays the 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, figure 7. Figure 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, figure 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)=0.43). Following this process, the other entries in row 11 of the Markov matrix in the example displayed by figure 7 are P(11,11) = 0.16, P(11,12) = 0.18, and P(11,20) = 0.23 for the ending states 11, 12, and 20 respectively.

Figure 7: The transformation of state 11 for a timestep Δt into some of the nearest neighbors of state 11. The percentages represent the percent of the transformed area in each state.

A key question in this set based concept is the estimation of the discretization of each cell. To estimate the number of particles needed to discretize each state, we utilize the deformation of
the fluid field, the velocity gradient tensor, equation 16. The deformation of the fluid flow encodes the possible transformations that elements can experience (translation, shear, rotation, dilation, or any combination of the previous transformations), figure 7.

\[
\nabla U(x, y, z) = \begin{bmatrix}
\frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\
\frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\
\frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z}
\end{bmatrix}
\] (16)

The elements that transform the most are those that experience the most deformation. Thus, to provide a measure for the total amount of deformation experienced by each element, the \(L_1\) norm of the velocity gradient is calculated for each Markov state, equation 17.

\[
\varepsilon_{def} = \sum_{i=1}^{3} \sum_{j=1}^{3} |(\nabla U)_{ij}|
\] (17)

This allows the identification of the element undergoing the maximum deformation. Then, the number of points used to discretize the boundary of each element is based on the number of points required to discretize the most deformed element. This element is initially discretized with 2 points per edge and then evolved through the previously chosen time step. The number of points per edge is then incrementally increased until the area or volume of the transformed cell converges to within a given tolerance, equation 18.

By multiplying the error times the number of states, the numerical error for the Markov matrix is bounded by the tolerance \(\varepsilon_{npts}\).

\[
\varepsilon_{npts} = n \times \left| \frac{\nu_{i-1} - \nu_i}{\nu_i} \right| \leq \varepsilon_{vol}
\] (18)

2.5 Flow Chart

A flow chart detailing the entire procedure from inputs to transient contaminant transport is provided in figure 8. 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.

Figure 8: A flow chart describing the methodology for creating accurate Markov matrices from inputs to contaminant transport.

2.6 Results

2.6.1 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, figure 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 nondimensional air speed standard deviation asymptotically converges.
The convergence of the non-dimensional air speed standard deviation can be seen in figure 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. Figure 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 64x32 cells for the Rayleigh–Bénard problem, and 96x32 cells for the IEA annex 20 problem. The Markov state discretizations of both domains can be seen in figure 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 3,072.

![Figure 9](image.png)

Figure 9: (a) Standard deviation of the nondimensional air speed as the size of the Markov states are changed. (b) Convergence of the standard deviation of the nondimensional air speed as the size of the Markov states are changed. The dotted line is the recommended tolerance 1% from equation 9.
Figure 10: The chosen Markov state discretization of the Rayleigh-Bénard and IEA annex 20 problems. The Markov state size for both domains is $H/h = 32$.

### 2.6.2 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, figure 11, and the number of states that have transitioned to their nearest neighbors, figure 12, are counted. Initially, the results show that all the Markov states are absorbed states, but as the time step is increased the particles start to leave their initial Markov state, figure 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, figure 11(a).
Figure 11: The fraction of absorbed states as the Markov time step is increased for different Markov state sizes.

Figure 12: The percentage of Markov states that have transitioned into their nearest neighbors as the Markov time step is increased for different Markov state sizes.

Once the particles start to leave their initial states, the particles enter to one of their nearest neighbors, figure 12. The peak of the curves (figure 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 figure 11 and figure 12 are clearly a function of the Markov state size. As the Markov state size is decreased each of the curves 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 figure 11 and figure 12, the minimum, ideal, and maximum time step for the Markov matrices were computed for each Markov size. These data are plotted in figure 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 figure 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, $\Delta t U_{max}/H$, for the Rayleigh-Bénard and the IEA annex 20 problem are 0.0736 and 0.1199 respectively.
Figure 13: Comparisons between the minimum, ideal, and maximum time step equations (equations 10, 11, and 15) and the data produced by tracking particles at the center of each Markov state. The data corresponds to the particle tracking calculations, and the mesh corresponds to the mesh based equations.

2.6.3 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. Figure 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.
Figure 14: The worst cells from the deformation measure, equation 17, for the chosen discretization, dotted line. The red cell is the transformed cell through the chosen time step. The number of points on each edge are the number of points needed to allow equation 18 to converge to 1%.

### 2.6.4 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 figure 15 for the Rayleigh-Bénard problem and figure 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, figure 17, and figure 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, figure 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, figure 18. For both figure 17 and figure 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.

Figure 15: Contaminant transport comparison between the advection diffusion equation and the Markov method for the Rayleigh-Bénard problem for a Markov discretization of $H/h = 32$. 
Figure 16: Contaminant transport comparison between the advection diffusion equation and the Markov method for the IEA annex 20 problem for a Markov discretization of $H/h = 32$.

Figure 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.
Figure 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.

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, figure 19. The maximum difference between the two methods for the Rayleigh-Bénard problem was about 16%, figure 19. The maximum difference between the two methods for the IEA annex 20 problem was about 11%, figure 19.
2.6.5 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 1,000 Markov time steps. The Rayleigh-Bénard problem took a simulation time of $t \frac{U_{\max}}{H} = 0.00864$ to evolve the contaminant a total time of $t \frac{U_{\max}}{H} = 54.7$. The IEA annex 20 problem took a simulation time of $t \frac{U_{\max}}{H} = 0.0315$ to evolve the contaminant a total time of $t \frac{U_{\max}}{H} = 119.8$. These tests used a 1.7 GHz Intel Core i7 processor. The Markov method was able to compute the contaminant transport 6,665 times faster than an optimized PDE solver for the Rayleigh-Bénard problem and 3,723 times faster than an optimized PDE solver for the IEA annex problem.
2.7 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 [39].

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 placed in this column can sense the largest volume of the space [32].
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.

2.8 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 3,000 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 during biological attacks.

2.9 Nomenclature

\( a \) Number of sorted air speed values below the \( \varepsilon_{tot} \times 100 \) percentile.

\( b \) A weighting factor to calculate the \( \varepsilon_{tot} \times 100 \) 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

\( l_{\text{inlet}} \) 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

\( n_{\text{abs}} \) The number of particles that have not transitioned from their initial state

\( n_{\text{nei}} \) The number of particles that have transitioned from their initial state to one of their nearest neighbors

\( n_{\text{pts}} \) The number of points per edge to discretize the Markov states

\( n_{\text{steps}} \) The number of time steps taken for a simulation

\( Ra \) The Rayleigh number

\( t \) Time

\( \Delta t \) Time step associated with the Markov matrix
\[ \Delta t_{\text{min}} \] 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

\[ S_\varphi \] Contaminant sources in the indoor environment

\[ T \] Temperature in the Rayleigh-Bénard Problem

\[ T_{\text{cold}} \] Top surface temperature in the Rayleigh-Bénard Problem

\[ T_{\text{hot}} \] Bottom surface temperature in the Rayleigh-Bénard Problem

\[ U \] Air velocity vector

\[ \| \| U \| \| \] Air speed

\[ \| \| U \| \|_a \] The \( a^{th} \) entry in the sorted (low to high) air speed from the average air speeds in each Markov cell

\[ \| \| U \| \|_i \] Air speed in Markov state \( i \)

\[ \| \| U \| \|_e \] The \( e_{101} \times 100 \) air speed percentile

\[ \| \| U \| \|_{\text{max}} \] Maximum air speed in the domain

\[ \langle \| \| U \| \| \rangle \] Average air speed for all the Markov states

\[ u \] The x-component of the air velocity vector

\[ V_i \] Volume of the transformed cell using \( i \) number of points per cell

\[ V_{\omega_k} \] Volume of the \( \omega_k \) state

\[ v \] The y-component of the air velocity vector

\[ w \] The z-component of the air velocity vector

\[ x \] The x-coordinate direction
The $y$-coordinate direction

The $z$-coordinate direction

$\varepsilon_{\text{error}}$ The $L_2$ Euclidean error between the Markov method and the CFD solution

$\varepsilon_{\text{def}}$ Deformation measure for the velocity gradient tensor

$\varepsilon_{\text{npts}}$ Error in the Markov matrix due to discretization of the boundaries of each transformed state

$\varepsilon_{\text{std}}$ The error of the air speed standard deviation between two successive refinements of the Markov state size

$\varepsilon_{\text{tol}}$ Tolerance to limit the maximum time step associated with the Markov matrix

$\varepsilon_{\text{vol}}$ The desired tolerance in the volume calculation of the transformed Markov state

$\theta$ Nondimensional temperature for the Rayleigh-Bénard Problem

$\mu^k_t$ Contaminant concentration in Markov cell $k$ at time $t$

$\mu_t$ Contaminant concentration field at the current time step

$\mu_{t+\Delta t}$ Contaminant concentration field at the next time step

$\sigma$ Standard deviation of the air speed in the Markov states

$\sigma(H/h_i)$ Standard deviation of the air speed in the Markov states with Markov state size $(H/h_i)$. $i - 1$ corresponds to the previous refinement level

$\sigma(H/h_{i-1})$ Standard deviation of the air speed in the Markov states with Markov state size $(H/h_i)$. $i$ corresponds to the current refinement level

$\varphi$ 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
2.10 References


CHAPTER 3. CONTAMINANT TRANSPORT AT LARGE COURANT NUMBERS
USING MARKOV MATRICES

A paper accepted by Building and Environment
Anthony D. Fontanini, Umesh Vaidya, Alberto Passalacqua, and Baskar Ganapathysubramanian

3.1 Abstract

Volatile organic compounds, particulate matter, airborne infectious disease, and harmful chemical or biological agents are examples of gaseous and particulate contaminants affecting human health in indoor environments. Fast and accurate methods are needed for detection, predictive transport, and contaminant source identification. Markov matrices have shown promise for these applications. However, current (Lagrangian and flux based) Markov methods are limited to small timesteps and steady-flow fields. We extend the application of Markov matrices by developing a methodology based on Eulerian approaches. This allows construction of Markov matrices with timesteps corresponding to very large Courant numbers. We generalize this framework for steady and transient flow fields with constant and time varying contaminant sources. We illustrate this methodology using three published flow fields. The Markov methods show excellent agreement with conventional PDE methods and are up to 100 times faster than the PDE methods. These methods show promise for developing real-time evacuation and containment strategies, demand response control and estimation of contaminant fields of potential harmful particulate or gaseous contaminants in the indoor environment.
3.2 Introduction

The human body may be exposed to many different types of potential pollutants, pathogens, and/or chemical and biological agents. These different gases, particulate matter (PM), and illnesses have many different sources. The products of combustion may be lingering in the air from traffic or industrial emissions. Volatile organic compounds (VOCs) may be evaporating or off-gassing from building materials, paints, or cleaning supplies. As people move around the building, dust, mold, pet hair/dander may be suspended in the air. Chemical or biological warfare (CBW) agents may be released during an attack or an act of terror. People coughing or sneezing in office, hospitals, or public transportation vehicles or terminals could be releasing airborne infectious diseases into an environment that can affect many people. Some of the health risks related with exposure to these contaminants can be as mild as fatigue, headaches, dizziness, or sinus irritation [1], but could be as severe as aggravated asthma, irregular heartbeat, lung cancer, heart disease, respiratory disease, and in extreme cases can even be fatal [1]. Particulate matter is currently controlled by the National Ambient Air Quality Standards (NAAQS). The particle sizes are generally broken into two classes 1) inhalable particle pollution with diameters < 10 μm (PM₁₀) and 2) fine respirable particle diameters < 2.5 μm (PM₂.₅). Exposure to small diameter PM has been linked to heart disease morbidity, lung cancer, cardiovascular and cardiopulmonary diseases [2]. The PM₂.₅ particles are small enough that when air-borne they experience long suspension times [3], [4], potentially effecting the occupants for long periods of time. The emission of VOCs can have many sources in the indoor environment (paints, adhesives, furnishing, clothing, building materials, combustion materials, and appliances) [5]–[9]. Health effects from VOCs include acute and chronic respiratory problems, neurological toxicity, lung cancer, and throat irritation [5], [9]–[15]. Gases from the soil or combustion products can also be harmful in large concentrations. Radon, a radioactive gas produced in soils, is the second leading cause of lung cancer [16]. Combustion products (NO₂ and
SO₂) from power generation and transportation vehicles can lead to an asthma aggravate [17], lead to reduced lung function, reduced lung development in children [18]. These are just some examples of how poor air quality from the surrounding environment, industries, or traffic may affect the human body. For these scenarios methods for detection, source identification, and real-time estimates of the concentrations can be helpful to keep the concentrations below the recommended exposure limits for the indoor environment.

Other instances where people may be exposed to potentially harmful gases and particulates is during extreme events. These events could be the spread of CBW agents during warfare or an act of terror. In 1995, the Tokyo subway system was attacked with sarin gas [19], which attacks the nervous system. In 2001, Florida, New York City, and Washington D.C. were the sites of anthrax attacks [20]. Other extreme events include the transmission of infectious diseases (TID). In these cases, a small number of individuals carrying an infectious disease enters an often close quarters public area like transportation terminals, public transportation vehicles, schools, offices, and hospitals. Influenza [21] along with the SARS virus [22] has been reported in airports in Asia and around the world. Other highly infectious diseases like measles [23], tuberculosis [24] have been reported in office spaces and hospital wards respectively. In the cases of CBW and TID scenarios immediate detection, fast long-term predictions, planning containment and evacuation strategies, along with identifying the source of the threat may help reduce the overall damage and help save lives.

In order to develop methods and strategies for optimal sensor locations, long-term predictions, determining containment and evacuation in real time, and identifying the sources of the contaminant, a fast and accurate contaminant transport method is needed. There are three broad approaches to construct spatial distribution of contaminants within a building zone: the partial differential equation (PDE) method, Lagrangian method, and the Markov method. For each of these
methods, the starting point is the underlying flow field calculated by computation fluid dynamics (CFD) or experimentally by methods like particle imaging velocimetry (PIV). The methods produce a time varying solution of the contaminant concentrations in the building zone. The PDE method is the most common method used in literature [25]–[29], where the advection diffusion equation PDE is solved on an underlying flow field. For particle transport, Lagrangian methods are popular [27], [30]–[34]. More recently Markov methods have been developed [4], [27], [35]–[37], which use Markov/transition matrices to propagate the scalar concentrations. Each of these methods have been shown to accurately describe the process of scalar or particle transport in and around buildings.

Although the PDE and Lagrangian methods are older and more established, the Markov method has some distinct advantages. First, the Markov matrix encodes where contaminants come from and go in one time step. This information has shown to be very helpful in calculating optimal sensor locations [38], which places sensors in locations where contaminants naturally collect. This information may also be helpful for real-time contaminant source location identification and calculating sensor estimators for estimating the scalar field in buildings. Second, the Markov method has been shown to be faster than the PDE method, and in some cases faster than the Lagrangian methods [27], [36]. Markov matrices have also been used to quantify mechanical ventilation performance [39] by using the fundamental matrix of absorbing Markov chains. Although these advantages have shown promise to the applicability of Markov methods, a major limitation to the methods have been using large timestep. The first methods used to calculate Markov matrices came from multi-zone methods [35], [37] which are flux based methodologies. In the flux based methodology, as the timestep associated with the Markov matrix increases to timesteps with Courant numbers greater than 1.0, the transition probabilities in the Markov matrix become more inaccurate [27]. In our previous work, a set-based method with Lagrangian particle dynamics was used to calculate larger timesteps [36], but as the timestep increases the set becomes more deformed and
difficult to accurately track and calculate 3D intersections. Previously published literature on Markov methods have also only been used on steady-state flow fields. Since Markov matrices have shown potential in different applications, resolving these limitations are the motivation for the current work.

The three-fold objective of this paper is to (a) establish methods for calculating accurate and usable large courant number Markov matrices, (b) introduce time varying source terms at large Courant numbers, and (c) extend Markov transport to transient flow fields. We first discuss two different methods for calculating large Courant number Markov matrices. The first method uses multi-step matrices, while the second method uses an Eulerian method to calculate individual rows of the Markov matrix. Calculation of time varying source terms at large matrices is discussed. Calculating contaminant transport in a time varying flow field using Markov matrices is shown. Speed increases seen by using the Markov method versus the PDE method is explored. The methods for calculating the Markov matrices are implemented in an open source CFD framework OpenFOAM. The methods developed in this paper allow for long-term predictions of the movement of contaminants to be made very quickly, optimal sensor locations (using the algorithm in our previous work [38]) for transient flow fields to be determined, and may lead to better utilization for HVAC demand response control of contaminants in the indoor environment.

The outline of this paper is as follows: Methods for simulating contaminant transport in steady and time varying flow fields using Markov matrices is discussed in section 2.1 and the calculation of large Courant number time varying source terms is presented section 2.2 and section 2.3. Next, the benchmark problems using in the analysis are discussed in section 2.4, and validation is shown in section 2.5. Then results are shown for contaminant transport for large courant number steady-state flow fields in section 3.1, time varying source terms under a steady flow field in section 3.2, and time varying flow fields are shown using the example problems in section 3.3. The
computational time is compared between the PDE method and the Markov method shown in this work in section 3.4. Then we discuss the applicability and implications of these methods in section 4 and conclude in section 5.

3.3 Methods

This section first provides a quick overview of how Markov matrices can be used for contaminant transport in steady and time varying flow fields. Then explains the two methods for large Courant number contaminant transport calculation. Finally, the method for calculating time varying source in the building is discussed.

3.3.1 Continuous and discrete representation of contaminant transport

The well-known and established advection diffusion PDE, Eq. 1, for contaminant (scalar) transport has been traditionally implemented with solving of Navier-Stokes and energy equations for CFD simulations.

\[
\frac{\partial \phi}{\partial t} + \nabla \cdot (U \phi) + \nabla^2 (D \phi) = S \tag{1}
\]

The advection diffusion equation, Eq. 1, describes the temporal evolution of a continuous scalar field transported by a continuous vector field with sources. The source term, \(S\), is a combination of both sources from an inlet and a volumetric source. The velocity field can be either time varying (transient), \(U_t\), or time invariant (steady state), \(U\). In order to numerically solve this equation, the equation is spatially and temporally discretized. Spatially, the equation is solved on a domain, \(\Omega\), composed of a set of cells \(\omega = \{\omega_1, ..., \omega_n\} \in \Omega\) and a set of boundary patches, \(\Gamma = \{\Gamma_1, \Gamma_2, ..., \Gamma_{nb}\}\), composed of inlets, \(\Gamma_{in} \subset \Gamma\), walls \(\Gamma_w \subset \Gamma\), and outlets, \(\Gamma_{out} \subset \Gamma\). Temporally, the time interval of interest, \(\tau\), is discretized into a set of discrete time instances, \(t = \{t(0), t(1), ..., t(m)\}\) with \(t(0) < t(1) < t(2) < ... < t(m)\) corresponding time steps, \(\Delta t_{(i+1)} = t(i+1) - t(i)\), that is used to
simulate the scalar concentrations from \( t_{(0)} \) to \( t_{(m)} \) for a set of flow field snapshots \( \mathbf{U}_t = \{ \mathbf{U}_{(0)}, \mathbf{U}_{(1)}, ..., \mathbf{U}_{(m)} \} \). The numerical schemes used to solve Eq. 1 in the domain from a given time instance, \( t_{(i)} \), to the next time instant, \( t_{(i+1)} \), creates a mapping, \( L(\cdot) \), Eq. 2.

\[
\phi_{(i+1)} = L(\phi_{(i)})
\]  

(2)

This mapping for the advection diffusion equation is a generalization of the continuous version of the Perron-Frobenius (PF) operator [40].

The discrete cell volumetric averages of the scalar field at a given time, \( t_{(i)} \), provides the link between the continuous and discrete versions of the PF operator, Eq. 3.

\[
\psi_{(i)}(k) = \frac{1}{V_{\omega_k}} \iiint_{\omega_k} \phi(x,y,z,t_{(i)}) \, dV, \quad k = 1, ..., n
\]  

(3)

For a steady flow field, a single snapshot of the flow field contains the information to evolve the discrete cell concentrations for any time interval of interest. These discrete cell concentrations are mapped from one time instant to another by a matrix-vector product, Eq. 4.

\[
\psi_{(i+1)} = \psi_{(i)} \mathbf{P} + S_{(i,i+1)} \quad \text{for} \quad i = 0: (m - 1)
\]  

(4)

Since the flow field is constant over the time interval of interest, a single matrix \( \mathbf{P} \) with an associated time step calculated from the flow field snapshot, can be used to evolve the scalar field. The matrix \( \mathbf{P} \) is a square probability matrix, Eq. 5, with all non-negative entries called transition probabilities, Eq. 6.

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

(5)

For time varying flow fields, a set of flow field snapshots, \( \mathbf{U}_t \), contains the information to evolve the discrete cell concentrations for the time interval of interest. In this case a set of Markov matrices are

\[1\] The Markov matrix has other properties that are explained in our previous work [36], [38], [39].
constructed for each snapshot $U_{(k)}$, Eq. 6, are used to evolve the scalar field for each time instant in $t$, Eq. 7(a-c).

$$P(t) = \{P_{(0,1)}, P_{(1,2)}, \ldots, P_{(m-1,m)}\} \quad (6)$$

$$\psi_{(1)} = \psi_{(0)}P_{(0,1)} + S_{(0,1)} \quad (7a)$$

$$\psi_{(2)} = \psi_{(1)}P_{(1,2)} + S_{(1,2)} \quad (7b)$$

$$\vdots$$

$$\psi_{(m)} = \psi_{(m-1)}P_{(m-1,m)} + S_{(m-1,m)} \quad (7c)$$

As discussed earlier, using Markov matrices for contaminant transport is a recent but not a new concept. The advantages for using Markov matrices is in the computational speed of propagating the scalar once the Matrix has been calculated [3], [27], [36], identify optimal sensor locations [38], and provide estimates for mechanical system air distribution performance [39]. A major limitation to these previous developed flux based methods is small time steps ($Co \sim 1.0$). The flux based methods start to break down because the entries of $P$ are calculated only for neighbors of any given cell, $\omega_i$. Therefore, for larger time steps larger timesteps, $Co > 1.0$, the transitional probabilities become increasingly more inaccurate. These effects might be limited to small regions of the domain at first, but over long time intervals of interest or long time steps these errors can dominate the results, see [27].

Figure 1 shows a 1D graphical example of how the flux based calculation methods can break down for large Courant numbers. Each of the 4 states are the same size, $dx$, and in each state the air velocity is the same. Therefore, there is an air flow rate between each adjacent states, $Q(i, i + 1)$. If a passive scalar is advected for a $Co = 0.5$ (the small dashed state in Figure 1), the flux based methods work well. The transition probabilities for state 0 would be $P(0,0) = 0.5$, and $P(0,1) = 0.5$. The scalar only transitions to the adjacent state, state 1, and not state 2 or state 3. On the other hand, if a passive scalar is advected for a $Co = 2.5$ (the large dashed state in Figure 1), the flux based methods
to not work well. The transition probabilities for state 0 should be $P(0,2) = 0.5$, and $P(0,3) = 0.5$, while the flux based method would produce an incorrect result of $P(0,1) = 1.0$. The flux based method would restrict the scalar from propagating further within the timestep, because 1st order neighbors are only stored in the mesh connectivity information. To overcome this limitation two options are available: 1) calculate and use multi-step transition probabilities from a smaller time step to propagate the scalar further in time with a single Markov matrix, 2) use a non-flux based method (ex: the Eulerian technique explained in section 2.3) to calculate the transitional probabilities for a larger time step.

![Graphical representation of the breakdown of the flux based methodology for large Courant numbers. The smaller dashed state is state 0 advected by a Co = 0.5, and the larger dashed state is state 0 advected by a Co = 2.5.](image)

**3.3.2 Calculating multi-step Markov Matrices**

The first approach for calculation large Courant number Markov matrices uses the multi-step transition probability feature of Markov Matrices. This technique can be used for both steady and transient flow fields. For steady state flow field problems, calculation of the multi-step transition probability equations takes the form of Eq. 8(a-c).

$$
\begin{align*}
\mathbf{\psi}(1) &= \mathbf{\psi}_0 \mathbf{P} + \mathbf{S}_{(0,1)} \\
\mathbf{\psi}(2) &= \mathbf{\psi}_0 \mathbf{P}^2 + \left( \mathbf{S}_{(0,1)} \mathbf{P} + \mathbf{S}_{(1,2)} \right) = \mathbf{\psi}_0 \mathbf{P} \mathbf{P}_{(0,2)} + \mathbf{S}_{(0,2)} \\
&\vdots \\
\mathbf{\psi}(m) &= \mathbf{\psi}_0 \mathbf{P}^m + \left( \sum_{i=0}^{m-1} \mathbf{S}_{(m-i-1,m-i)} \mathbf{P}^i \right) = \mathbf{\psi}_0 \mathbf{P} \mathbf{P}_{(0,m)} + \mathbf{S}_{(0,m)}
\end{align*}
$$
This strategy for steady flow fields involves taking \( m \) matrix-matrix products of the Markov matrix to produce a longer time step associated with the effective Markov matrix, \( P_{(0,m)} \). The matrix-matrix multiplication only needs to be performed once and can be done offline (before the transport simulation begins). An effective source term, \( S_{(0,m)} \), is constructed, Eq. 8(b-c), since all preceding powers of the Markov matrix would otherwise need to be stored and would eventually result in memory limitations. A method for calculating the effective source terms in the domain offline is discussed in section 2.4. For transient flow field problems, the multi-step transitional probability equations take the form of Eq. 9(a-c).

\[
\begin{align*}
\psi_{(1)} &= \psi_{(0)} P_{(0,1)} + S_{(0,1)} \\
\psi_{(2)} &= \psi_{(0)} P_{(0,1)} P_{(1,2)} + (S_{(0,1)} P_{(1,2)} + S_{(1,2)}) = \psi_{(0)} P_{(0,2)} + \tilde{S}_{(0,2)} \\
\vdots \\
\psi_{(m)} &= \psi_{(0)} \prod_{i=0}^{m-1} P_{(i,i+1)} + \left( \sum_{i=0}^{m-1} (S_{(m-i-1,m-i)} \prod_{j=0}^{i-1} P_{(m-j-1,m-j)}) \right) \\
&= \psi_{(0)} P_{(0,m)} + \tilde{S}_{(0,m)}
\end{align*}
\]

This strategy for transient flow fields shows how to take a set of Markov matrices to produce the effective Matrix, \( P_{(0,m)} \), through a set of matrix-matrix products. In the same way as the steady state flow fields, an effective source term can be calculated to reduce memory requirements and decrease calculation time.

### 3.3.3 Eulerian method for calculating a Markov matrix

Each row, \( i \), of a Markov matrix represents where state \( i \) would transition to in the next Markov time step. The Eulerian method essentially consists of placing a concentration of 1.0 only at state \( i \), Figure 2, and computing how that initial concentration spreads to the rest of the states during the timestep associated with the Markov timestep, Eq. 10.
\[ P_{(k,k+1)}(i,j) = \psi_{(k+1)}(j) \quad \text{for} \quad i = 1: n \]  

Notice in Figure 2 that the concentrations are far away from the 1\textsuperscript{st} order neighbors of state \( i \) at the next Markov time instant.

Figure 2: The calculation of a single row \( i \) of the Markov matrix using the Eulerian Markov method. A single state \( i \) is initialized to a value of 1.0, then the advection diffusion equation is solved propagating the cell concentration to other regions in the domain.

This approach has calculated the Markov matrix by solving \( n_z \) advection diffusion equations. Since each row of the Markov matrix can be calculated independently of any other row, all the rows can be simulated in parallel to reduce computational time. Much larger time steps associated with the Markov matrix, \( \Delta t_{(k,k+1)} \), can be calculated directly without performing the the matrix-matrix multiplication in Eq. 8(a-c) and Eq. 9(a-c).\(^2\) The only difference between steady flow fields and transient flow fields is that after the output of the Markov matrix, \( P_{(i,i+1)} \), the discrete cell concentrations must be reinitialized before calculating, \( P_{(i+1,i+2)} \). As the number of timesteps grows larger, the number of non-zero entries in the Markov matrix increases. To help save memory at least

\(^2\) For cells with different volumes, the row sum of the Markov matrix may not sum to 1.0, but the column sum of the Markov matrix should sum to 1.0. As for uniform discretization of the domain both the row and column sums should sum to 1.0. Therefore, a uniform discretization results in doubly stochastic matrices, and a non-uniform discretization results in a left stochastic matrix.
for shorter timesteps a convenient storage method is using sparse matrices. In sparse matrices only
the non-zero entries are stored in a format \(\{i, j, P(i, j)\}\).

### 3.3.4 Calculating transient source terms

Contaminant sources can originate at any location in the domain or at an inlet boundary, and
the release of the source may take many different forms. The release may be constant, similar to a
scenario where smog may be entering a room through an open window, figure 3a. The release may
decay exponentially, similar to a scenario where a reaction is taking place in the domain or in the
ductwork leading into the domain, figure 3b. The release may be periodic, similar to a scenario
where a human or animal is exhaling carbon dioxide, figure 3c. Many other release scenarios are
possible, where a linear function can be used to approximate smaller intervals of the release, figure
3d. This section discusses a method for calculating volumetric sources and sources from inlets.

In calculating the effective source terms, the goal is to provide a generic functional form that
can easily be calculated offline and easily implemented for online use. The source term in Eq. 1 can
be composed of a number of different source terms, \(n_{source} = n_{inlet} + n_{vol}\). These sources can either
be inlet sources, \(n_{inlet}\), or volumetric sources, \(n_{vol}\), Eq. 11.

\[
\bar{S} = \sum_{i=1}^{n_{inlet}} S_{in,i} + \sum_{i=1}^{n_{vol}} S_{vol,i}
\]  

(11)

Each of these sources in Eq. 11 can be collected into a set of contaminant sources that are present
in the domain, Eq. 12.
Figure 3: Example effective source terms where the source originates from the inlet as a) a constant function, b) an exponential function, c) a periodic function, and d) a linear function.

Each one of these sources can be a function of both time and space, Eq. 13.

\[
\mathbf{S} = \{s_{in,1}, s_{in,2}, ... , s_{in,n_{inlet}}, s_{vol,1}, s_{vol,2}, ... , s_{vol,n_{vol}}\} 
\]

(12)

(13)

The spatial portion of a given inlet or volumetric source is computed by simulating a unit response of the source over the Markov timestep. The unit response of the source term, \( \mathbf{S}_i \), is calculated solving Eq. 1 for the Markov timestep and gathering the cell concentrations into a unit response vector for inlet sources\(^3\), Eq. 14, and volumetric sources, Eq. 15.\(^4\)

---

\(^3\) If the initial concentration of the inlet source is 0 at \( t = 0 \), then \( \Gamma_{in,i} = g_{s,i}(t) \).

\(^4\) For periodic source terms, it is convenient to use a Markov timestep, \( \Delta t_{(i,i+1)} \), equal to the period of the source function such that a constant spatial vector, \( \mathbf{f}_{s,i} \), can be used for the duration of the simulation.
\[ f_{s,i} = \psi_{(1)} \quad \text{with} \quad \Gamma_{in,i} = g_{s,i}(t)/\max\left(g_{s,i}(0 \geq t > \Delta t_{(0,1)})\right), \quad \psi_{(0)} = 0 \quad (14) \]

\[ f_{s,i} = \psi_{(1)} \quad \text{with} \quad \Gamma_{in} = 0, \quad \psi_{(0)}(j) = S_{(0)}/\max\left(g_{s,i}(0 \geq t > \Delta t_{(0,1)})\right) \quad (15) \]

**Remark:** The calculation of the components, \( \tilde{S}_i \), of time varying source term, \( \tilde{S} \), is similar to using a Green’s function. Since the Markov method is a linear transformation of the contaminant concentrations from one time instant to another, in the same way the spatial unit source vectors can be scaled and superimposed to calculate the time varying effective source term, \( \tilde{S} \).

### 3.3.5 Test cases and CFD validation

We illustrate developments using three example problems. One case represents a room with a human to measure the influence of displacement ventilation on contaminant transport and source locations for applications in passive smoking and transport of infectious diseases [41]. The next benchmark case has been used for validating methods for quantifying the performance of ventilation systems and evaluating the effectiveness of the air distribution system in removing contaminants [39], [42], [43]. The final problem is a time varying flow field [38] with heated block obstructions and a window (used in a previous analysis).

#### 3.3.5.1 Benchmark test for a computer simulated person

The room used in this benchmark test has the following geometric setup, and is shown in figure 3. The room has dimensions of 3.5 (m) x 2.5 (m) x 3.0 (m) in the x, y, and z directions respectively. The inlet (air supply) is positioned in the center of the left wall on the lower edge and has dimensions of 0.4 (m) x 0.2 (m) in the z and y directions respectively. The outlet (air return) is the same dimensions as the inlet, but is positioned in the center of the right wall on the edge shared with the ceiling. The benchmark case is manikin free, which allows for different representations of the manikin to be used. Based on the images in the benchmark problem [41], a manikin was created.
to mimic these images. The manikin was placed 0.05 (m) above the floor in accordance with the benchmark problem to avoid heat transfer between the manikin and the floor. To simplify the computational cost of the simulations the solution was assumed to be symmetric along the x-axis, which allows for half the room to be modeled. This half of the room and manikin was discretized into 3 different mesh resolutions 30,729, 131,431, and 223,637 cells to investigate the convergence of the solution at different mesh resolutions. Figure 4c shows the 223,637 element discretization.

![Diagram of the room and manikin with different discretizations](image)

Figure 4: a) The dimensions of the room with the inlet (green) on the lower part of the side wall in front of the manikin and the outlet (red) on the upper part of the side wall behind the manikin. The manikin is positioned in the center of the space. b) The sampling positions along the center of the room where both experimental data and the numerical simulations are compared. c) The discretization of the room and the manikin with 223,637 elements used to create the transition matrices.

Using the three different discretizations, the simulation of the flow field was performed. The turbulence model chosen for this analysis was the RNG k-epsilon model, because of the accuracy
shown in previous works on this benchmark problem. Buoyancy effects were introduced using the Boussinesq approximation for density. The inlet air velocity was 0.182 (m/s) at 22 (C), a turbulence intensity of 30%, and a turbulent length scale of 0.1 (m). This air flow rate for this room gives an air change rate of 0.5 (hr). The manikin generates 76 (W) with 40% convective and 60% radiative heat transfer. While the side walls, ceiling, and floor was designed to be perfectly insulated, some heat transfer was found to have occurred [44]. According to previous work performed on this problem, the heat transfer can be represented as a 10 (W) heat source by the floor [44]. Each mesh discretization and simulation setup was solved to a residual tolerance of 1e-4 using OpenFOAM [45]. Once the simulations converged, the air speed and temperature were compared at 4 different positions along the center of the x-axis from the floor to the ceiling, figure 4b.

The results show that there is a small temperature gradient from the floor to the ceiling due to stratification, while a thermal plume is present around the manikin, figure 5a. Results from the 3 different mesh resolutions are shown at each of the sample locations in figure 5. All the different mesh resolutions agree relatively well with the experimental data, figure 6. The sample positions downstream in the manikin L4 and L5 show some discrepancy, but this is mainly due to the fact that the original manikin geometry was not specified by the benchmark problem and that a hoist was used to hold up the manikin during the experiment which may have affected the results. Based on these results the 223,637 element discretization was determined to be an adequate resolution for the rest of the analysis.
Figure 5: a) The air speed surface contours displaying the thermal plume around the manikin and the streamlines showing how air moves around the room. b) The Temperature surface contours showing the air stratification within the room and the thermal plume around the manikin.

Figure 6: a) Air speed comparisons at the sampling profiles between the three different mesh resolutions and the experimental air speed measurements from the benchmark problem. b) Temperature comparisons at the sampling profiles between the three different mesh resolutions and the experimental temperature measurements from the benchmark problem.
3.3.5.2 Benchmark test for an isothermal room

The room in this benchmark has the following geometry setup and boundary conditions, and is found in figure 7 and figure 8. The room is 4.2 (m) x 3.0 (m) x 3.6 (m). The inlet is positioned on the near sidewall in figure 7, with dimensions 0.3 (m) along the z-axis and 0.2 (m) along the y-axis. The bottom edge of the inlet is 2.05 (m) from the floor and the center is located along the center xz-plane of the domain. The outlet is also positioned along the xz-plane with dimensions of 0.2 (m) along the z-axis and 0.15 (m) along the x-axis. The inlet air velocity is uniform at 1.68 (m/s). The inlet turbulent kinetic energy is based on a 14% turbulent intensity and epsilon is calculated from $\epsilon_{in} = k_{in}^{1.5} / (0.005 \sqrt{A_{in}})$. The air flow field has been shown to be relatively symmetric [43], so only half the domain was discretized and simulated.

![Figure 7: Dimensions of the benchmark room. The inlet (green) is on the near side wall and the outlet (red) is on the ceiling.](image)

The air in the room was simulated using the RNG k-epsilon model to a residual tolerance of 1e-6 for all solution parameters. These conditions were simulated with 2 different mesh resolutions (22,680 elements and 181,440 elements). Validation of this numerical model, boundary conditions, and mesh resolutions are shown in our previous work [39]. In our previous work the age of air, based on passive scalar tracer gas measurements, is compared with experimental values. The results of the simulations of the model, and boundary conditions showed excellent agreement with the
experimental values. The resulting flow field of the simulation is shown in figure 8. The results of the simulated benchmark problem show that air travels from the inlet straight across to the far wall. At this point some of the air goes towards the outlet while the rest spreads out along the far wall and is circulated through the rest of the space.

![Figure 8: a) The discretization used in the simulations and the previously validated simulations [39]. The discretization has 22,680 elements. b) The air flow field of the isothermal benchmark problem from the simulated boundary conditions and discretization.](image)

3.3.5.3 **Benchmark test for time varying flow fields**

The room in this problem was used in our previous work [38]. The room has the same dimensions as the IEA annex 20 problem [46] with the addition of two obstructions located at 1/3 and 2/3 along the x-axis, figure 8a. These obstructions are heated with 70 (W/m$^2$) and there is a window along the right wall of the room that is heated 100 (W/m$^2$). The inlet temperature is 293 (K), while the rest of the walls are zero gradient. The air enters the room with a Reynolds number of 2,500 until the flow field reaches steady state. Then the air flow rate is doubled to a Reynolds number of 5,000 at t = 0 (sec). The sudden increase in the air flow rate causes the flow field to be time varying. Also at t = 0 (sec), a scalar contaminant starts to be injected into the space from the inlet. All other boundary conditions are consistent with the IEA 20 annex problem [46].
Figure 9: a) The geometry and boundary conditions for the transient flow field problem. b) The mesh used with a total of 10,280 elements. c) The velocity magnitude contours at $t = 0$. d) The temperature distribution in the space at $t = 0$.

The flow field was solved for using the RNG-k-ε model for turbulence. The domain was discretized into 10,280 elements, figure 9b. This resolution was sufficient to capture the flow field, figure 8c, and the temperature field, figure 9d, during a spatial convergence study. The steady state problem was solved initially to provide the initial conditions of the transient problem. The governing equations for the steady-state problem was solved to a residual tolerance $1 \times 10^{-7}$ for all solution variables. After the steady-state problem was solved, the Reynolds number was increased to 5,000 and the temporal solution was calculated using a timestep of 0.1. This timestep with the discretization provided a Courant number of approximately 1.0. Each of the solution variables at each timestep was solved to a residual tolerance of $1 \times 10^{-6}$.

### 3.4 Results

This sections shows how the Markov method and PDE methods compare under different flow conditions and source conditions. For example, the two methods are compared using a 3D steady
flow field with constant source release. In this example, the results show the Markov method using large Courant numbers. The next example shows the results of the Markov based transient source terms for large Courant numbers. Then a set of Markov matrices are calculated and used to propagate the contaminant under a transient flow field. Finally, speed comparisons are made between the PDE approach and the Markov approach for different solving tolerances of the PDE method and Courant numbers for the Markov method. These examples show how the Markov method can be used under steady state and transient flow fields along with constant and time varying contaminant release scenarios.

3.4.1 Steady flow field contaminant transport

The case simulated to evaluate the Markov methods on a steady flow field at high Courant numbers is the manikin example in section 3.3.5.1. The advection diffusion PDE was simulated using a gas diffusivity of $7.334 \times 10^{-5} \text{ (m}^2/\text{s})$, which represents helium tracer gas in air at 298 (K) [47], and a timestep of 0.02-sec which corresponds to a maximum Courant number of 1.0 for the chosen discretization. Markov matrices where calculated for Markov timesteps of 0.2-sec, 2.0-sec, and 4.0-sec using the 0.02-sec PDE timestep. These Markov matrices represent a maximum Courant number of 10, 100, and 200. The contaminant was released into the domain at a constant rate by the inlet at a normalized concentration ($\psi/\psi_{in}$) of 1.0. The contaminant was released in the domain for 1 air change (30 min).

As time progressed, the contaminant starts to fill the room, Figure 10. The concentration of the contaminant initially is advected along the floor and then starts to move its way towards the ceiling. Around the manikin, the thermal plume seen in figure 5a helps carry the contaminant towards the ceiling. The normalized concentration is shown at three different time snap shots at 5-min (figure 10a), 10-min (figure 10b), and 15-min (figure 10c). The concentrations along L1, L2,
L4, and L5 of the contaminant at 10-min, 20-min, and 30-min are used to compare the PDE method and the Markov method at different Courant numbers. The results in figure 11 show that for L1, L2, L4, and L5 the Markov method compares very well with the PDE method for the 10-min, 20-min, and 30-min samples. The results of the Markov method are almost overlaid with each other on each of the subplots even at the Courant number of 200. This results shows that the Markov method can be used to take time steps at least two orders of magnitude larger than the PDE method with extremely accurate results.

3.4.2 Time-varying source terms in a steady state flow field

The case used to show how Markov methods perform using transient source terms at high courant numbers is from section 3.3.5.2. The same gas diffusivity is from the previous example. In this problem four different source terms injected from the inlet are analyzed (constant, linear, exponential, and periodic). The four functions are seen in equations 16a-16d.

\[
\begin{align*}
g_{s,i} &= 1.0 \quad & (16a) \\
g_{s,i} &= \frac{1}{450} t \quad & (16b) \\
g_{s,i} &= e^{-\frac{1}{300} t} \quad & (16c) \\
g_{s,i} &= \frac{1}{2} - \frac{1}{2} \cos \left( \frac{\pi}{5} t \right) \quad & (16d)
\end{align*}
\]

For each of the inlet source functions a source term and Markov matrix was calculated for a Courant number of 16.67 or 1 (sec) timestep, except for the periodic source term in equation 16d where a 166.7 Courant number was used or a 10 (sec) timestep. Each of the scalar transport simulations were run for 450 (sec).
Figure 10: a) Left: Iso-contours of the normalized contaminant concentration at (a) 5-min, (b) 10-min, (c) 15-min. Right: Normalized contaminant concentrations along the z-centerline plane at (a) 5-min, (b) 10-min, and (c) 15-min.
Snapshots at 210 (sec) and 450 (sec) of the scalar transport simulations can be seen for each source term in figure 12. The concentration of the scalar in the room for the constant and linear source terms cases increases with time in figure 12a-12b and 12c-12d respectively. For the exponential case, the 210 (sec) snapshot has a higher scalar concentration than the 450 (sec) snapshot, figure 12e-12f. The periodic nature of the inlet source condition can be seen in figure 12g and 12h. At the wall the concentration of the scalar is much higher whereas at the inlet the concentration is much lower.

Figure 13 compares the scalar concentrations along the center z-plane at two locations along the x-axis (x = 1.13 (m) and x = 3.2 (m)). Each of the different source terms are compared at three different time instances t = 150 (sec), t = 300 (sec), and t = 450 (sec). For all the inlet source cases (constant, linear, exponential, and periodic) the Markov method and PDE method show excellent agreement.
These results show that the Markov method for time varying source terms at high Courant numbers provide the same results as the PDE method at lower Courant numbers.

### 3.4.3 Time-varying flow field contaminant transport

The case used for analyzing contaminant transport in time varying flow fields for both the PDE and Markov methods is from section 3.3.5.3. The Reynolds number at $t = 0$ (sec) is increased from 2,500 to 5,000. The transient simulation is run for 400 (sec). One hundred Markov matrices are created for the 400 second simulation. The Markov matrices propagate the scalar from $t = 0$ (sec) to $t = 4$ (sec), then $t = 4$ (sec) to $t = 8$ (sec), …, $t = 396$ (sec) to $t = 400$ (sec). The 4 second timesteps for the Markov matrices correspond to a Courant number of approximately 40. Snapshots of the normalized flow field, temperature field, Markov concentrations, and PDE concentrations can be seen at $t = 132$ (sec), 264 (sec), and 400 (sec) in figure 14. Figure 14 shows that after the increase in inlet Reynolds number, higher velocity air starts to push back against the buoyancy dominated flow seen in figure 9. A recirculation cell starts to grow, figure 14, as the temperature in the area to the left of the first obstruction cools. Eventually the air jet from the inlet comes down on top of the first obstruction. Meanwhile, the scalar starts to enter the space. Initially the scalar follows the higher speed regions of the space, then starts to diffuse to the centers of circulation. The left side of the room fills first, as most of the air is trapped by the jet landing on top of the obstruction.
Figure 12: Results of the transient source simulations with sources at the inlet at a time of 210 (sec) and 450 (sec). a) constant source at 210 (sec), b) constant function at 450 (sec), c) linear function at 210 (sec), d) linear function at 450 (sec), e) exponential function at 210 (sec), f) exponential function at 450 (sec), g) periodic function at 210 (sec), h) periodic function at 450 (sec).
Figure 13: Scalar value comparisons along two lines at the center z-plain at $x = 1.13$ (m) and $x = 3.2$ (m) between the Markov method and the PDE method for different inlet source functions. a) Constant source function at $x = 1.13$ (m), b) linear source function at $x = 1.13$ (m), c) exponential source function at $x = 1.13$ (m), d) periodic source function at $x = 1.13$ (m), d) constant source function at $x = 3.2$ (m), b) linear source function at $x = 3.2$ (m), c) exponential source function at $x = 3.2$ (m), d) periodic source function at $x = 3.2$ (m).
Figure 14: Contours of normalized velocity magnitude (top left), temperature (bottom left), contaminant concentrations from the PDE method (top right), and contaminant concentrations from the Markov method (bottom right) for three different time snapshots a) $t = 132$ (sec), b) $t = 264$ (sec), and c) $400$ (sec).
Visually there is little difference between the contours of the Markov method and the PDE method at the time snapshots shown in figure 14. Samples from the floor to the ceiling at $x = 1.5$ (m), $x = 4.5$ (m), and $x = 7.5$ (m) are also compared at $t = 132$ (sec), $t = 264$ (sec), and $t = 400$ (sec) in figure 15. For each of the times investigated and all 3 sampled locations the Markov method provides the same concentrations as the PDE method. These results shows that A set of Markov matrices can be used to propagate a contaminant field under a time varying flow field and give roughly equivalent contaminant concentrations as the PDE method.

![Figure 15: Scalar concentrations of the Markov and PDE methods compared at three different x-locations a) $x = 1.5$ (m), b) 4.5 (m), and c) 7.5 (m). Three different time snapshots $t = 132$ (sec), $t = 264$ (sec), and $t = 400$ (sec) are shown at three different x-locations.](image)

### 3.4.4 Computational time comparisons between the Markov and PDE methods

This section evaluated the computed speed increases by using the Markov method over the PDE method at large Courant numbers along with the memory requirements necessary to store these matrices. For this comparison the 3D problem stated in section 2.5.1 was used to compare the speed requirements using Markov matrices associated with Courant numbers of 10, 100, and 200 from section 3.1. The time required to propagate the scalar from $t = 0$ (min) to $t = 30$ (min) was recorded
for the PDE method and the three Markov matrices. For the PDE method OpenFOAM was used to simulate the PDE method on the steady flow field, and the Markov methods used MATLAB to calculate the scalar propagation. For the Markov method, the operations for each step includes a single matrix-vector product to propagate the scalar forward in time and a single vector-vector addition to add the effective source term at each time step. For the PDE method, the solution tolerance, \( r \), of the advection diffusion equation makes a big difference in terms of computational time. As the solution tolerance decreases, more iterations at each time step is needed to solve for the scalar field.

After the simulation times were recorded, the speed increase relative to the PDE method was calculated. Figure 16a shows that for all the solution tolerances and Courant numbers the Markov method is between 30 and 120 times faster than the PDE method\(^5\). The increase in speed is mostly due to the use of sparse matrices where only the non-zero entries are stored and require operations during the matrix-vector product. The matrices at least for Courant number less than 200 are still very sparse matrices as seen in Figure 16b. The percentage of non-zero entries range between 0.03% at a Courant number of 10 to 0.4% at a Courant number of 200. These percentages may change based on the individual flow field used. The sparsity percentages should be most sensitive to the Reynolds number or air change rate of the room, the amount of mixing in building zone from obstructions and people, and the diffusivity of the contaminant. In figure 16a, simply increasing the Courant number by an order of magnitude does not necessarily mean an order of magnitude speedup. There is an interplay between the Courant number and the sparsity of the matrix. As the Courant number increase for the Markov matrix larger steps forward in time can be accurately calculated, but more operations are needed to calculate the next step in time. Overall, the Markov method for large

---

\(^5\) The 5.59 times speedup reported in this previous study [27] was reported to be roughly independent of the number of elements in the discretization of the domain.
Courant numbers with sparse matrix implementation is much faster than the PDE method and is faster than the previous implementation by C. Chen et al. [27]. It is expected that the Markov method should result in anywhere from 1 to 2 orders of magnitude speedup for a given problem, and could be even faster if implemented in a non-interpreted language (like C or Fortran).

Figure 16: a) Speedup comparisons between the Markov and PDE methods at different solving tolerances of the PDE method and Courant numbers of the Markov matrix. b) The percentages of non-zero entries of the Markov matrices as the Courant number increases.

3.5 Discussion

In this paper Markov methods for contaminant transport were extended to Courant numbers much larger than 1.0, transient source terms at Courant numbers greater than 1.0, and applications in time varying flow fields. Although these methods show excellent speed increases from the standard PDE approaches, the overhead costs of calculating the Markov matrices can be computationally expensive. In order to construct a Markov matrix, \( n + n_{source} \) PDE simulations need to be computed. If the contaminant transport for a particular flow field and a particular set of source conditions wants to be analyzed, the PDE method is computationally much cheaper than constructing the Markov matrix. The Markov methods are much better for problems that need very quick estimates where contaminants may be at a time far in the future, applications that allow these matrices to be used more than once, or the cases where the information contained in the Markov...
matrix is further exploited. Some examples applications would be determining evacuation strategies once a toxic gas is dispersed somewhere in a building or building zone, using the transport for demand response and control of the HVAC system, or using the matrices to determine sensor locations or quantify mechanical ventilation performance. With Markov matrices already showing applications with determining optimal sensor placement strategies and quantifying mechanical system ventilation performance, the methods developed in this paper allow for analysis in the applications for time varying flow fields. The methods are easily extendible to stochastic systems and problems with probabilistic boundary conditions. Other future directions include extracting more information from the Markov matrices themselves to create estimators for sensors, and determine contaminant source locations.

3.6 Conclusions

With a large number of potential harmful gases and particulates that the human body can be exposed to in indoor environments, it is important to have fast and accurate predictive numerical methods to determine concentrations within the indoor environment. In this paper Eulerian and multistep Markov methods were shown to provide accurate simulation results compared to traditional PDE methods, but at much higher Courant numbers. The methods developed were also applicable to transient inlet and volumetric contaminant sources at large Courant numbers and calculation of contaminant transport in time varying flow fields. The large Courant numbers along with sparse implementation of the Markov matrices allowed for relative speed increases from 30 times to 120 times faster than the more established PDE methods. Thus allowing for larger problems to be used in a predictive sense for real-time analysis and control.
3.7 Nomenclature

\( Co \)  The Courant number

\( D \)  Contaminant diffusivity in air

\( f_{s,i}(x) \)  The spatially varying function of the \( i \)th source term in the set of all contaminant sources

\( g_{s,i}(t) \)  The time varying function of the \( i \)th source term in the set of all contaminant sources

\( g_{s,in} \)  The time varying function of the inlet source

\( L(\cdot) \)  The Perron-Frobenius operator

\( m \)  The number of time instances between the starting time and \( \tau \)

\( n \)  The number of cells or states in the domain

\( nb \)  The number of boundary patches

\( n_{inlet} \)  The number of inlet contaminant sources

\( n_{source} \)  The number of sources terms

\( n_{vol} \)  the number of volumetric contaminant sources

\( P \)  Markov matrix

\( P(i,j) \)  The \( i \)th row and \( j \)th column entry of the Markov matrix.

\( P_{(i,i+1)} \)  Markov matrix that propagates the scalar from time instant \( i \) to time instant \( i+1 \)

\( P(t) \)  The set of time varying Markov matrices used to propagate a scalar from time instant 0 to time instant \( m \)

\( S \)  Spatially continuous contaminant source

\( S_{(i,i+1)} \)  Discrete contaminant source between the \( i \)th and \( i+1 \) time instants

\( S_{(0,i)} \)  The effective source term from the 0\(^{th} \) time instant to the \( i \)th time instant

\( \mathcal{S} \)  The set of all contaminant sources
\( \bar{S}_i \)  The ith source term in the set of all contaminant sources

\( Re \)  Reynolds number

\( T_{\text{inlet}} \)  Temperature at the inlet

\( t \)  time

\( t_{(i)} \)  The ith time instant

\( t_{PDE} \)  The wall time required to complete a simulation with the PDE method

\( t_{\text{Markov}} \)  The wall time required to complete a simulation with the Markov method

\( U \)  Steady state velocity field

\( U_t \)  Time varying velocity field

\( U \)  Air speed in the domain

\( U_{in} \)  Air speed at the inlet

\( U_{(i)} \)  The velocity field at the ith time instant

\( V_{ok} \)  The volume of the kth state

\( x \)  The x-direction

\( y \)  The y-direction

\( z \)  The z-direction

\( \Delta t_{(i, i+1)} \)  The times step between the ith and the i+1 time instant

\( \Gamma \)  The set of boundary patches in the domain

\( \Gamma_{in} \)  The set of inlet boundary patches

\( \Gamma_{out} \)  The set of outlet boundary patches

\( \Gamma_w \)  The set of wall boundary patches

\( \phi \)  Continuous contaminant scalar field

\( \psi_{(i)}(k) \)  Discrete contaminant concentration of state k at the ith time instant.

\( \psi_{in} \)  Discrete contaminant concentration at the inlet
\( \psi_{(i)} \)  The discrete contaminant concentration field at the \( i \)th time instant

\( \tau \)  Time interval for which the contaminant is spreading in the domain

\( \theta \)  Normalized temperature

\( \omega \)  a set of cells or stats in the domain

\( \Omega \)  Domain of interest

3.8 References


CHAPTER 4. A METHODOLOGY FOR OPTIMAL PLACEMENT OF SENSORS IN ENCLOSED ENVIRONMENTS: A DYNAMICAL SYSTEMS APPROACH

A paper accepted by Building and Environment

Anthony D. Fontanini, Umesh Vaidya, and Baskar Ganapathysubramanian

4.1. Abstract

Air quality has been an important issue in public health for many years. Sensing the level and distributions of impurities help in the control of building systems and mitigate long term health risks. Rapid detection of infectious diseases in large public areas like airports and train stations may help limit exposure and aid in reducing the spread of the disease. Complete coverage by sensors to account for any release scenario of chemical or biological warfare agents may provide the opportunity to develop isolation and evacuation plans that mitigate the impact of the attack. All these scenarios involve strategic placement of sensors to promptly detect and rapidly respond.

This paper presents a data driven sensor placement algorithm based on a dynamical systems approach. The approach utilizes the finite dimensional Perron-Frobenius (PF) concept. The PF operator (or the Markov matrix) is used to construct an observability gramian that naturally incorporates sensor accuracy, location constraints, and sensing constraints. The algorithm determines the response times, sensor coverage maps, and the number of sensors needed. The utility of the procedure is illustrated using four examples: a literature example of the flow field inside an aircraft cabin and three air flow fields in different geometries. The effect of the constraints on the response times for different sensor placement scenarios is investigated. Knowledge of the response time and coverage of the multiple sensors aides in the design of mechanical systems and response
mechanisms. The methodology provides a simple process for place sensors in a building, analyze the sensor coverage maps and response time necessary during extreme events, as well as evaluate indoor air quality. The theory established in this paper also allows for future work in topics related to construction of classical estimator problems for the sensors, real-time contaminant transport, and development of agent dispersion, contaminant isolation/removal, and evacuation strategies.

4.2. Introduction

4.2.1 Sensors in buildings and applications

Indoor air quality has been a major concern since the 1970s, and is a major objective for the Environmental Protection Agency (EPA), the National Institute for Occupational Safety and Health (NIOSH), and the American Society of Heating Refrigeration and Air Conditioning Engineers (ASHRAE). People in the western world can spend up to 90% of their lives indoors [1], [2]. Leaky roofs, problems with the HVAC systems, construction of overly tight buildings, the use of synthetic building materials, and chemical based personal care products are primary causes for a variety of health related problems. The exposure to air pollutants (CO2, VOCs, CO, radon, mold, atmospheric particulate matter, and microbial contaminants) may lead to immediate or long-term effects health issues. Immediate symptoms may be as mild as headaches, dizziness, fatigue, and irritation of the sinuses [3]. People most susceptible to these air pollutants include the young, the elderly, and the chronically ill [3]. These effects may also be amplified for people suffering from asthma, which affects approximately 26 million people in the US[4], resulting in 2 million hospital visits in 2010 [3], and costing Americans approximately $56 billion from 2002-2007 [4]. Long-term exposure may lead to respiratory disease, heart disease, cancer, and can even be fatal [3]. Due to these factors, improving or maintaining indoor air quality (IAQ) is essential during the design process, construction, and the operating lifetime of buildings.
Besides air quality in the indoor environment, another area that has received a lot of attention is indoor environment safety/security during extreme events. Examples of extreme events include chemical and biological warfare (CBW) attacks as well as the transmission of infectious diseases (TID). Particular examples of CBW attacks are the Sarin gas attack on the Tokyo subway system in 1995, [5] and the anthrax attacks in Florida, New York City, and Washington D.C. in 2001 [6]. The agents used during these attacks were highly lethal, and a rapid detection may have mitigated their impact. The TID are of concern in highly populated and confined areas such as close office quarters, train/subway stations, train/subway cars, airports, aircraft, hospitals, and schools. Documented cases of the spread of influenza [7] and the SARS virus [8] have been reported in aircraft travel, outbreak of measles in office spaces [9], and tuberculosis infection of health care workers [10]. A recent outbreak is the MERS-CoV outbreak in health care facilities in South Korea and China [11]. As in the case of the CBW agents, rapid detection may have helped isolate the spread of the infections before a large number of people were exposed to these diseases.

A plan of action for addressing indoor air quality (IAQ), CBW attacks, and TID outbreaks includes risk assessment and preventive measures, identification of problematic areas and potential sources, and development of response strategies, Figure 1. The common link between these steps are the design, deployment, data collection, and data analysis of sensors in the building. The risk assessment stage relies on predictive modeling (zonal airflow network modeling, CFD and contaminant transport simulations, design of sensor locations, and construction of sensor estimators) to provide a priori design information for effective systems. The identification stage depends on sensors to provide critical information about contaminant levels and sources locations (using the sensor estimations for real-time estimation and control). The response strategies help control, isolate, or remove the detected contaminants (using sensor estimators for source identification, determine where to disperse neutralizing agents, and using real-time data to develop an evacuation
strategy). In the application of IAQ the response strategies are HVAC controls for demand response of HVAC systems, which have been shown to be a viable energy saving control strategy and to increase IAQ [12]–[18]. For CBW/TID applications, the response strategies can include evacuation plans, a plan for dispersion of a neutralizing agent, or an isolation plan to prevent further spread of the CBW agent/TID pathogen.

Due to recent advances in sensors and wireless sensor networks [19]–[21], data can be collected and harnessed for control of the mechanical systems, as well as source identification and response strategies for both IAQ and CBW applications. Wireless sensors networks connected by either Bluetooth or the internet to the building’s automation system enables timely decisions about the building’s indoor environment. The standardization of high-level communication protocols from low power devices [19], have extended the battery life and provide robust sensing platforms to transmit data quickly over longer distances. Other guidelines developed for the optimization of network communication by clustering and data aggregation [22] further enable sensor deployment in sophisticated ways.

Figure 1: Plan of action for design, identification, and response strategies to mitigate the risk involved for IAQ, CBW, and TID applications.
4.2.2 Sensor placement methods

Although wireless sensor networks have general design guidelines that standardize communication, power consumption, the number of cluster heads, and data aggregation, guidelines for placement of the sensors is usually based on their application. For IAQ applications some design criteria has been established for multiple sensors that may aid in tracking sources and enabling HVAC controls [23]. These criteria include the coverage of deployed sensors, a notion of sensitivity or detection threshold, and response time of the sensors [23]. For CBW applications the focus is on reducing the overall impact of the attack in terms of average impact damage, worst-case impact damage, and the cost of the sensors [24]. Understanding the basic goals or objectives of the sensors and the building environment can guide the sensor placement strategy.

Based on these concepts, current methods for sensor placement design can be classified into the following categories: 1) engineering/heuristic methods, 2) optimization methods and inverse methods. Engineering methods are generally based on experience and rules of thumb for a given set of products and applications. For example a common choice for sensor placement is to uniformly place the sensors in the space. The number of sensors in this scenario may be very large, which for fairly expensive sensors may not be feasible. Other limitations of the engineering methods include a lack of control of the response time, some areas left uncovered, and lack of generalizability to multiple rooms or zones [23].

Due to these limitations of engineering methods, optimization methods have recently become popular. Optimization methods try to place sensors by taking into account the airflow patterns (from either CFD, a zonal model, or a multizone airflow model) in a building [24]–[32]. An objective/cost/fitness function is created based on the goals of the deployed sensors. This cost function is maximized using an optimization algorithm. Optimization methods typically result in understanding some notion of the sensor locations, coverage volume, and response time for a given
set of release scenarios. There has been some investigation as to which air flow model is most appropriate (CFD, zonal, or multizone) for optimization problems [31], [32]. During this investigation multizone air flow models seem to perform just as well as CFD airflow models, and may be easier to setup and solve. In contrast, CFD simulations provide more detail information about the flow field than multizone approaches.

Over the last few years many different optimization techniques have been developed to optimally place sensors in a building’s indoor environment. Genetic algorithm, simulated annealing, and perturbation stochastic approximation methods have been used to place CBW sensors with a CFD enhanced multizone airflow model [25], [30]–[32]. Other examples include optimal sensor location for aircraft cabins starting from a set of initial sensor locations and a set of release scenarios [26]. General recommendations for sensor locations for an aircraft cabin have been given in [27]. The applicability of inverse methods for advection-diffusion problems have been recently reviewed [28]. The adjoint probability inverse modeling method has been shown to work well in a multizone airflow model where the number of potential sensor locations is relatively small [29]. However, when the number of potential locations for a sensor is fairly large, the method (especially when involving CFD for the forward problem) may become computationally prohibitive. The use of some heuristics is usually needed to alleviate some of the computational complexity. Inverse methods [23] along with artificial neural networks [33] have been used to quickly find the source location in a building through both CFD and multizone methods.

Although optimization methods can be a powerful tool, designing a proper cost function that enables quick and smooth minimization is not always easy. Furthermore, these methods usually involve solving the contaminant advection-diffusion PDE. Finally, optimization methods are inherently iterative, and may require solving a large number of simulations to find the optimal locations. This iterative process can be computationally intensive. Optimization methods are also
dependent on a set of release scenarios that are designed based on the space geometry and sensor application. The bottleneck for both optimization and inverse methods is the necessity of solving a partial differential equation for contaminant transport for numerous scenarios and multiple iterations.

In this context, recent work on reformulating the contaminant transport problem as a dynamical system opens up powerful, computationally cheap, and fast methods for predicting contaminant transport and contaminant evolution in the indoor environment. Such a formulation also allows novel developments in control theory to be applied for identifying sensor locations for multizone and CFD airflow simulations. Furthermore, such a formulation can leverage the impressive body of work on sensor placement, which is a classical problem in dynamical systems and state space systems. For instance, the continuous form of the Perron-Frobenius (PF) operator has been effectively used to determine where density (of, say, contaminants) gathers under the effect of non-linear vector fields [34]. More recently, the discrete finite dimensional form of the PF operator, which takes the form of a Markov Matrix, has been used to identify optimal sensing locations [35], [36]. Utilizing the finite dimensional PF operator has been shown to be a simple way to calculating real-time contaminant transport in buildings [37]. Additionally, such analysis also provides insight into the age of the air and where contaminants spend the most time [38]. The critical advantage of this method is that the finite dimensional PF operator method is relatively computationally inexpensive, as they do not require solving the (direct or adjoint) advection-diffusion PDE or minimization of a complicated cost function.

This paper is built upon previous work [35], [37] of utilizing a dynamical systems approach to indoor air analysis. The main idea is determining sensor locations from velocity vector fields (acquired from analytical methods, CFD, particle imaging velocimetry, etc.). We leverage the finite dimensional PF operator framework and showcase a sensor placement algorithm that is very easy to solve. This algorithm naturally accounts for complex geometries, constraints in sensor
placement, as well as sensing constraints. We illustrate the capabilities of this framework by comparing results with established recommendations from published literature. The sensor placement strategy for a given building zone is investigated for both IAQ and CBW applications. This approach is a data driven method that maximizes the sensing volume by taking into account the number of sensors and the response time for a given accuracy of the sensors. The method described in this paper has the potential to provide designers, engineers, and architects with not only the sensor locations and sensor coverage for deployed wireless sensors, but also the framework for classical estimator problems and real time contaminant transport.

The outline of rest of the paper is as follows. Some mathematical preliminaries are discussed followed by the sensor placement algorithm. Then sensor placement constraints and sensing constraints are introduced into the sensor placement algorithm. The placement strategy is compared with published recommendations in an aircraft cabin. Finally, multiple sensors are placed for a building zone using a CFD datasets with applications in IAQ and CBW.

4.3. Methods

In this section some mathematical preliminaries on dynamical systems and the PF operator are introduced. Following the mathematical preliminaries, the sensor placement algorithm is described. Finally, methods for including sensor accuracy, sensor location constraints, and sensor sensing constraints into the sensor placement algorithm are explained.

4.3.1 Mathematical preliminaries

The advection of a contaminant\(^1\) (gas or particle) in an airflow field, \(U\), is described
by the well-known scalar transport model (advection-diffusion partial differential equation),
\[
\frac{\partial \varphi}{\partial t} + \nabla \cdot (U \varphi) + \nabla^2 (D \varphi) = S \varphi
\]  
(1)

The advection diffusion PDE describes how a contaminant density evolves/changes under a flow field, figure 2a and figure 2b. Note that the flow field can be generated analytically, by computational fluid dynamics (CFD) simulations, or by experimental methods like particle image velocimetry (PIV). This flow field, \( U = f(x, t) \), represents a (2D or 3D) spatially varying vector field. Additionally, the flow field may be a function of time. From a dynamical systems perspective, we denote the flow field and the associated transport as a time varying dynamical system.

\[
U = \dot{x} = f(x, t) \text{ with } x = \{x, y, z\} \in X
\]  
(2)

Analogously, for steady state flows, the dependence on time is removed. From a dynamical systems perspective, we now denote the system as a time invariant dynamical system.

\[
U = \dot{x} = f(x)
\]  
(3)

For either a steady or time varying flow field the advection diffusion PDE, equation 1, maps a contaminant concentration in a building zone at some time \( t \) to some future time \( t + \tau \). This mapping, \( L(\cdot) \), is known as the PF-operator\(^2\).

\[
\varphi_{t+\tau} = L(\varphi_t)
\]  
(4)

If, say, two sensors are placed at locations \( j \) and \( l \), the contaminant concentration at these locations are measured by these sensors. This can be mathematically expressed in terms of the output equations, \( Y(x, t) \)

\[
Y(x, t) = \lambda_k(x) \varphi(x, t); \quad \bar{k} = \{j, l\}
\]  
(5)

\(^2\) The PF operator has two forms: continuous and discrete. The continuous form of the PF operator takes the form of the passive scalar transport [34], equation 1 under high Reynolds number conditions. The discrete form takes the form of a Markov matrix and is discussed in the following section.
where \( \lambda_E(\mathbf{x}) \) is the indicator function for the sensor location (the indicator function is zero everywhere except in a finite region around a sensor, i.e. in the ‘sensing’ volume).

### 4.3.2 Representing a vector field as a Markov matrix:

Although scalar transport by solving the PDE has been a popular method for determining contaminant concentrations, the method has recently been pointed out to have some limitations [38]. These limitations include loading a potentially large amount of data into memory, solving a PDE, which can be time consuming for real-time decision making, and may require specialized software packages or commercial software. As a result of these limitations the discrete finite dimensional PF operator has become of interest (Figure 2). The discrete finite dimensional PF-operator breaks up the domain into a set of cells \( \{\omega_1, \ldots, \omega_n\} \). These cells are also called states. The movement of contaminants over some time interval is represented as a (probabilistic) transition from an initial state, \( i \), to a final state, \( j \). The collection of all these probabilistic transitions over a time period in the domain is the discrete finite dimensional PF-operator and corresponds to a Markov matrix (also viewed as an adjacency or transition matrix), equation 6.

\[
P = \begin{bmatrix}
P_{11} & P_{12} & \cdots & P_{1n} \\
P_{21} & P_{22} & \cdots & P_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
P_{n1} & P_{n2} & \cdots & P_{nn}
\end{bmatrix}
\] (6)

The Markov matrix has some special properties. All the values are non-negative, equation 7, and the sum of the elements in a given row is always 1.0, equation 8.

\[
P_{ij} \geq 0 \quad \text{for all } i, j \tag{7}
\]

\[
\sum_{j=1}^{n}(P_{ij}) = 1 \quad \text{for all } i \tag{8}
\]

We refer the interested reader to [28] which exhaustively details the construction of Markov matrix, \( P \), from the velocity field information, \( U \). The Markov matrix, \( P \), serves as the finite
dimensional representation of the advection-diffusion PDE, equation 1, on the states of the Markov matrix. The (volumetric) average contaminant concentration in the Markov states at a given time, $t$, is given by $\mu_t = (\mu_1^t, ..., \mu_n^t)^T \in \mathbb{R}^n$. The contaminant concentration in a specific cell, $\omega_k$, at time $t$ is related to the contaminant density function and can be obtained for Markov state, $\mu_t^k$, as

$$\mu_t^k = \frac{1}{V_{\omega_k}} \int_{\omega_k} \varphi(x,y,z,t) \, dV, \quad k = 1, ..., n$$  \hspace{1cm} (9)

Figure 2: a) Continuous field of a contaminant at time $t$, b) continuous field of a contaminant at time $t + \tau$ from applying the scalar transport equation, equation 3, c) Discrete field of a contaminant at time $t$, d) discrete field of a contaminant at time $t + \tau$ after the discrete form of the PF operator has been applied, equation 9c.
For steady flow fields: given a Markov matrix, $P$, a contaminant concentration in the Markov states at time $t_0$, and a source, $S$, the evolution of the contaminant for some time $\tau = m\Delta t$ is governed by the simple nested matrix-vector products shown in equation 10(a-c). A graphical example of the discrete finite dimensional PF operator is shown in figure 2c and figure 2d.

$$
\mu_{t_0+\Delta t} = \mu_{t_0}P + S \\
\mu_{t_0+2\Delta t} = \mu_{t_0+\Delta t}P + S \\
\vdots \\
\mu_{t_0+m\Delta t} = \mu_{t_0+(m-1)\Delta t}P + S
$$

(10a)  
(10b)  
(10c)

For unsteady flow fields: a set of Markov matrices governs the contaminant evolution at a set of time snapshots, equation 11.

$$
P(t) = \{P^{(0,1)}, P^{(1,2)}, ..., P^{(m-1,m)}\} \quad with \quad t = \{t_0, t_1, ..., t_m\} \\
t_0 < t_1 < t_2 < ... < t_m \quad and \quad \tau = t_1 + t_2 + ... + t_m - t_0
$$

(11)

The Markov matrix $P^{(i,i+1)}$ takes the contaminant density from time $t_i$ to time $t_{i+1}$. The evolution of the contaminant on the time dependent flow field is calculated by equations 12(a-c).

$$
\mu_{t_1} = \mu_{t_0}P^{(0,1)} + S \\
\mu_{t_2} = \mu_{t_1}P^{(1,2)} + S \\
\vdots \\
\mu_{t_m} = \mu_{t_{m-1}}P^{(m-1,m)} + S
$$

(12a)  
(12b)  
(12c)

Finally, the discrete counterpart of the output equation (equation 5) for both steady and unsteady flow fields for sensor locations at state $j$ and $l$ ($k = \{j, l\}$) can be written as equation 13a-c. We first define the sensing volume of a sensor placed at cell/state $a$ as;

$$
c^{(a)} \in \mathbb{R}^{n \times 1}, \quad c_i^{(a)} = \begin{cases} 1 & i = a \\ 0 & \text{else} \end{cases} \quad for \quad i = 1: n.
$$

(13a)

This is simply the discrete form of the indicator function. For multiple sensors, we generalize this to a matrix of indicator functions, $C = [c^{(j)} c^{(l)}]$. Now, the value sensed by each sensor is simply
the inner product of the contaminant vector with the indicator, and is written as

$$Y_t^j = [\mu_t \cdots \mu_t^n] c^{(j)} \quad \quad Y_t^l = [\mu_t \cdots \mu_t^n] e^{(l)}$$

$$(13b)$$

$$Y_t = [Y_t^j \quad Y_t^l] = \mu_t[c^{(j)} \quad e^{(l)}] = \mu_t C_k$$

$$(13c)$$

Thus, the discrete sensor outputs at time $t$ are given by, $Y_t$.

Markov Matrices have shown promising results as an alternative to scalar transport [38]–[40]. A robust procedure of constructing Markov matrices from a generic flow field is shown in the authors’ previous work [38]. The data driven process uses set theory and involves choosing a Markov state size, choosing the time step associated with the Markov matrix, the calculation of the entries of the Markov matrix, and efficient storage of the often sparse Markov matrix.

### 4.3.3 Sensor placement algorithm

The sensor placement algorithm involves the construction and analysis of the discrete time observability gramian. The observability gramian is a control theoretic tool and is used to characterize the relative degree of observability of various states in the system (more specifically, the state space) [41]. The current work build upon our earlier work in [37] and [42], where we extended the definition of the observability gramian to nonlinear flow field and advection diffusion PDEs. Roughly speaking, the observability gramian for advection diffusion PDE can be defined as a function from the physical space $X$ to real valued nonnegative number. If the observability gramian on a particular region, say $B \subset X$, is positive then the density function (contaminant distribution) on the set, $B$, can be estimated by building an estimator. On the other hand, if the gramian on a particular set is zero then one cannot design an estimator to estimate the density supported on the set. Furthermore, if the value of the gramian on set $B_1 \subset X$ is large compared to $B_2 \subset X$, then it is relatively easier to observer the density function on set $B_1$ compared to set $B_2$. In other words, the gain of the estimator to estimate the density function on $B_1$ is smaller compared to one required to
estimate density function on set $B_1$. We refer the reader to [35] for more details on this discussion. Given the fundamental role that the observability gramian plays in the design of estimators, it is widely used in system and control theory as a natural cost function for the purpose of sensor placement. We extend this to the current application. Specifically, the observability gramian is constructed using the Markov matrix. We then use the observability gramian to identify locations/cells that increase the relative degree of observability of various cells. These cells will serve as the locations of the sensors.

In this section the calculation of the observability gramian is discussed for both steady state and transient flow fields. Also, the process of including sensor accuracy into the calculation of the observability gramian and placement constraints and sensing constraints on the sensors are discussed.

4.3.3.1 Calculation of the contaminant tracking matrix

Towards the construction of observability gramian we first construct the contaminant tracking matrix, $Q^{(\tau)}$, for the time interval $\tau$ by identifying how a constant source of contaminants is transported over the time interval, $\tau$. For steady flows this is given as:

$$Q^{(\tau)} = I + P + P^2 + P^3 + \cdots + P^m$$

Here the exponent, $m$ satisfies $\tau = m\Delta t$, where $\Delta t$ is the time step used to construct the Markov matrix [37]. Note that $P$ propagates the contaminant from time $t$ to time $t + \Delta t$, $P^2$ propagates the contaminant from $t$ to $t + 2\Delta t$, and $P^k$ propagates the contaminant from $t$ to time $t + k\Delta t$. The sum of $I, P, P^2, ..., P^m$ builds a history of states in which contaminants travel over the time interval $\tau = m\Delta t$, given that the contaminant started in an initial state at time $t$. $Q^{(\tau)}$, thus, represents the cumulative effect over $m$ steps of the Markov matrix. A similar analysis results in the construction of $Q^{(\tau)}$ for transient flow fields. For transient flow fields, there is a set of time
snapshots \((t_1, t_2, \ldots, t_m)\) where Markov matrices propagate the contaminant field from \(t_i\) to \(t_{i+1}\). As \(P^{(0,1)}\) propagates the contaminant from time \(t_0\) to time \(t_1\), \(P^{(0,1)}P^{(1,2)}\) propagates the contaminant from \(t_0\) to \(t_2\), and \(\prod_{i=1}^{k} P^{(i-1,i)}\) propagates the contaminant from \(t_0\) to time \(t_k\). Equation 15 gives the contaminant tracking matrix for transient flow fields for the time interval \(\tau = t_m - t_0\).

\[
Q^{(\tau)} = I + P^{(0,1)} + P^{(1,2)} + \prod_{i=1}^{3} P^{(i-1,i)} + \cdots + \prod_{i=1}^{m} P^{(i-1,i)}
\] (17)

**Remark:** The \(^i\)th row of the Markov matrix stores the destination states of containment released in state \(i\) [37] after one time step. Similarly, the \(^i\)th column of the Markov matrix stores originating states from which contaminant reaches state \(i\), after one step. Since the contaminant tracking matrix builds a sensing history as time evolves from time \(t_0\) to \(t_0 + m\Delta t\), the columns of the contaminant tracking matrix store the originating states over \(m\) time steps. A graphical example of contaminant tracking matrix is seen in figure 3, for various values of \(m\). Here, the non-zero entries of the contaminant tracking matrix are displayed. The Markov matrix is from published data of contaminant transport in an aircraft cabin [40]. The contaminant tracking matrix starts filling up for increasing \(m\), indicating that that flow is spreading across the domain.

---

**Figure 3:** Contaminant tracking matrix calculated from equation 14 at a) 4 seconds, b) 8 seconds, and c) 12 seconds of an aircraft (Markov matrix from C. Chen et al. 2014 [40]).
4.3.3.2 From the contaminant tracking matrix to the observability gramian

Since the flow field has been formulated in terms of a dynamical system and the Markov matrix has defined a notion of states, the concept of observability can be utilized. In a classical sense, observability is a metric that determines how well the states can be inferred by a system’s output. The outputs (in the context of this work) are the observations of sensors placed in discrete states of the domain. The discrete time observability gramian determines whether an arbitrary state is observable within a given time horizon, $\tau$. Consider a case where a sensor is placed at location $i$ with reference to the contaminant tracking matrix $Q^{(\tau)}$. The indicator matrix $C$ corresponding to a sensor placed in state $i$ is given by $C_i$. The relative degree of observability for different cells is given by

$$\sigma_i = Q^{(\tau)}C_i$$ (14)

Note that $Q^{(\tau)}C_i$ for a single sensor placed in state $i$ is a column vector with nonnegative entries. In particular, $\sigma_i$ will be a column vector where the positive entries are states that are observable from state $i$ and the zero entries are states that are not observable from state $i$. For placement of 2 sensors in locations $\vec{k} = \{i, j\}$, $Q^{(\tau)}C_{\vec{k}}$ is a 2 column matrix with nonnegative entries, equation 15.

$$\sigma_{\vec{k}} = Q^{(\tau)}C_{\vec{k}}$$ (15)

The positive entries of the 1st column of $\sigma_{\vec{k}}$ correspond to the states that are observable from state $i$, and the positive entries of the 2nd column correspond to the states that are observable from state $j$. The zero entries of $\sigma_{\vec{k}}$ in the 1st and 2nd column correspond to states that are not observable from states $i$ and $j$ respectively. The construction of $\sigma_{\vec{k}}$ can easily extended to more than 2 sensors. With
some abuse of terminology, we will call $\mathbf{\sigma}_{\mathbf{\hat{K}}}$ as the observability gramian\(^3\) for a set of sensor locations $\mathbf{\hat{K}}$ and corresponding indicator matrix $\mathbf{C}_{\mathbf{\hat{K}}}$.

4.3.3.3 Including sensor accuracy into the contaminant tracking matrix

Based on the sensor’s sensing accuracy the signal will be unavoidably corrupted by noise. The quality of the sensor can be naturally accounted for in the contaminant tracking matrix. In the contaminant tracking matrix, large values in a given column represent a very strong signal while smaller values represent weaker signals. Although a theoretically perfect sensor may be able to accurately sense the contaminant in states with low values, realistically sensors have accuracy thresholds. This problem is remedied by thresholding the values in the contaminant tracking matrix. The threshold value is a nondimensional value that describes the ratio of source release rate, the release time, and the value detected at the sensor, equation 18. For a given source, the source releases contaminants at a given rate, $S_{\text{source}}$, over the given sensing period, $\tau$. The sensors accuracy in this context is the percentage of the contaminant needing to be sensed over the sensing period. Typically, a detection threshold and sensor accuracy is known for a given sensor.

$$\varepsilon_{\text{acc}} = \frac{\mu_{\text{detect}}}{S_{\text{source}} \tau} = \frac{\mu_{\text{detect}}}{\mu_{\text{source}}}$$

(18)

Using the sensor’s accuracy as the threshold value, the values less than the accuracy of the sensor in the contaminant tracking matrix (example: 1%, 5%, 10%, etc.) are replaced by zero, while any value larger is kept for the placement analysis. This creates a new thresholded contaminant tracking matrix, equation 19. The effect of this thresholding can be seen in figure 4a and figure 4b.

$$\mathbf{Q}^{(\tau)} = \mathbf{Q}^{(\tau)} > \varepsilon_{\text{acc}}$$

(19)

\(^3\) For more rigorous definitions please refer to [35].
4.3.3.4 Including constraints into the contaminant tracking matrix

Other factors that affect the decisions of sensor placement in buildings are location constraints and sensing constraints. Location constraints exist because some practical limitations exist that preclude placing a sensor in a given state. Some practical limitations include the following: the sensor location is not serviceable; occupants have the ability to move or bump into the sensor, damage the sensor, or be annoyed by the location of the sensor; and sensor design limits the placement to a wall or ceiling. Location constraints exist for IAQ, CBW, and TID applications. Often expensive CBW and TID sensors would have to be placed outside the occupied zone, such that the sensors are not tampered with or easily damaged. Occupants may not want CO2 and VOC sensors in the occupied space such that they have to walk around them, avoid stepping on them, or for ascetic reasons. The placement of sensors at the top of a tall atrium may not be a good location, because if a sensor malfunctions the sensor may not be easily replaced or serviced. Incorporating these constraints is done after the contaminant tracking matrix has been thresholded. If a certain state cannot accommodate a sensor (location constraint), then the column is removed from the matrix. To remove a set of states \( j \in \mathbb{N}^{n_{loc}} \) from the sensor selection process, the entries in column \( j^{th} \) are replaced by zeros, figure 4c and equation 20.

\[
\tilde{Q}^{(r)}(:, j) = 0
\]  

(20)

Sensing constraints is more applicable for IAQ sensors as a designer, engineer, or architect is primarily concerned with the CO2 and VOC levels in the occupied space. In this case, we want to remove any consideration of states that lie outside the occupied zone. To remove a set of states \( i \in \mathbb{N}^{n_{sen}} \) from sensor placement consideration, the entries in the \( i^{th} \) rows are replaced by zeros, figure 4d and equation 21.

\[
\tilde{Q}^{(r)}(i, :) = 0
\]  

(21)
Figure 4: a) Theoretical contaminant tracking matrix from equation 16 or 17, b) the contaminant tracking matrix after the accuracy thresholding equation 16, c) the thresholded contaminant tracking matrix after including a sensor location constraint to the occupied space in equation 20, d) the thresholded contaminant tracking matrix with sensor location and sensing constraints in equation 21.

4.3.3.5 Placement of Sensors

After the construction of the contaminant tracking matrix, thresholding for accuracy requirements, and selection of placement and sensing constraints, the sensors can be placed. We perform the analysis for contaminant released in an arbitrary state, and then derive the generalized sensor placement algorithm from this discussion. Consider a specific release scenario in the \( a^{th} \) Markov state, \( r^{(a)} \), given as

\[
r^{(a)} \in \mathbb{R}^{1 \times n}, \quad r_i^{(a)} = \begin{cases} 
1 & , i = a \\
0 & , otherwise 
\end{cases}
\]

A set of sensor locations, \( \bar{K} \), should be designed such that the \( a^{th} \) Markov state is maximally observable from the chosen sensors. This can be represented as

\[
s^{(a)} = r^{(a)} \tilde{Q}^{(r)} c_{\bar{K}}
\]
The preceding statements can be extended to a case of \( b \) release scenarios, \( A = \{a_1, a_2, ..., a_b\} \), as.

\[
\mathbf{r}^{(A)} \in \mathbb{R}^{b \times n}, \quad r^{(A)}_{ij} = \begin{cases} 1 & j = A_i \\ 0 & \text{else} \end{cases} \quad \text{for} \quad i = 1:b, \ j = 1:n
\]  

(24)

The sensor locations, \( \mathbf{k} \), are designed to maximize the number of observable states in the release scenarios from \( A \), equation 25.

\[
\mathbf{s}^{(A)} = \sum_{i=1}^{b} (\mathbf{r}^{(A)} \mathbf{Q}^{(\tau)})_{ij} \mathbf{C}_k
\]  

(25)

Finally, to include all possible release scenarios for every Markov state, the release scenario matrix, \( r^{(\text{all})} \), becomes the identity matrix, equation 26.

\[
r^{(\text{all})} \in \mathbb{R}^{n \times n} = \mathbf{I}
\]  

(26)

Similar to equation 25, the sensor locations are chosen to maximize the number of observable states for all possible states, equation 27.

\[
\mathbf{s}^{(\text{all})} = \sum_{i=1}^{n} (\mathbf{I} \mathbf{Q}^{(\tau)})_{ij} \mathbf{C}_k
\]  

(27)

Equations 23, 25, and 27 are design problems for \( \mathbf{C}_k \) to maximize \( \mathbf{s}^{(a)}, \mathbf{s}^{(A)}, \) and \( \mathbf{s}^{(\text{all})} \) respectively. That is, identify number and location of sensors to completely cover the states in \( \mathbf{Q}^{(\tau)} \).

The choice of sensor locations, \( \mathbf{k} \), to establish complete coverage of the states using the minimal number of sensors is identical to a problem in combinatorics called the “set cover problem.” The set cover problem is one of Karp’s NP-complete problems [43]. We use a near optimal greedy algorithm [44] to determine the sensor locations. The basic idea of the greedy algorithm is to choose the sensor locations based on the largest number of uncovered states. The positivity property of the Markov operator is used to prove that the greedy algorithm is optimal for sensor placement [36]. We remind the reader that the columns of the contaminant tracking matrix holds the observable states (nonzero entries) for a sensor placed in the respective column. The 1\textsuperscript{st} sensor, \( \mathbf{k}_1 \), is placed where column support is maximized [35] (i.e. the column with the largest number of non-zero entries):

\[
\mathbf{k}_1 = \max_j \left\{ L_0 \left[ (\mathbf{Q}^{(\tau)})_{\text{(:,j)}} \right] \right\}
\]  

(28)
Placing the first sensor at $k_1$ results in a fraction of the states being observed. For subsequent sensor placement consideration, these states are removed from the contaminant tracking matrix, which is denoted by $\bar{Q}^*(r)$. With every new sensor being placed, the contaminant tracking matrix is updated to only reflect the states that are not currently observed. The next sensor is then placed based on the set of release scenarios, $B$, not yet covered by all previously placed sensors, equation 29.

$$
\bar{k}_p = \max_j \left\{ L_0 \left[ r^{(B)} \bar{Q}^*(r) \right]_{(c,j)} \right\} \quad \text{for} \quad p = 2: N \tag{29}
$$

This process is illustrated in Figure 5. By placing a sensor in state 7, the states that can be sensed are 2, 6, 7, 8, 9, and 10. A sensor placed in state 7 can sense 60% of the domain. After a sensor has been placed in state 7, entries in the rows and columns of state 2, 6, 7, 8, 9, and 10 are no longer of interest. A sensor in state 7 can sense these states, so column 7 and rows 2, 6, 7, 8, 9, and 10 are set to zero, (grayed in figure 5c and figure 5d). After the placement of the first sensor and the removal of its sensing volume, subsequent sensors can be placed to sense the remaining volume. Placing the second sensor in state 1 allows for the sensors to sense state 1, 3, 4, and 5 in addition to the volume sensed by state 7. A second sensor in state 1 is able to sense the remaining 40% of the building. For this example, only 2 sensors are necessary to sense the entire space for the time interval, $\tau$. The columns with the largest number of nonzero entries can observe/sense the largest percentage of states/volume in the building environment. The indices of the nonzero entries in the columns of the contaminant tracking matrix provide a coverage map of the observable states for a given Markov state, figure 6.
Figure 5: Outline of the Greedy algorithm. a) An example contaminant tracking matrix at time $t + \tau$, b) selection of the state with the largest column support to place the first sensor, c) removal of the states sensed by the 1st sensor, d) selection of the 2nd sensor based on the largest remaining column support.
4.3.4 Overview of Algorithm

The sensor placement problem is formulated as the well-known set cover problem, and a greedy algorithm is used to place a minimum number of sensors with maximum coverage. This algorithm is designed to maximize the sensing volume of each subsequent sensor for the volume in the space not yet sensed by previously placed sensors. This algorithm accounts for all possible release scenarios (as the Contaminant tracking matrix includes all release scenarios) instead of selecting only a few scenarios (as is the case for other methods). That is, the algorithm includes release scenarios for every discrete volume (cells) in the CFD simulation. Furthermore, the total amount of contaminant released (independent of release time) within the sensing time determines the sensor placement. The variables in the placement algorithm are the number of sensors, the response time, and the total coverage of all the sensors. Based on these inputs, three different placement strategies are outlined below. The first few steps for each strategy are the same and are displayed in the flow chart in figure 7.
Figure 7: Flow chart that displays the initial steps to calculate the thresholded constrained contaminant tracking matrix required for the sensor placement algorithm

**Case 1: For a known response time and a desired total coverage percentage**

1. Use flow chart to calculate the thresholded constrained contaminant tracking matrix
2. Determine column with maximum support and place a sensor in this state, eq. 28 or 29.
3. Replace the column of the sensor and the rows of the sensed states with all zero entries.
4. Repeat steps 2 and 3, until desired coverage percentage is reached.

**Case 2: For a known number of sensors and a response time**

1. Use flow chart to calculate the thresholded constrained contaminant tracking matrix
2. Determine column with maximum support and place a sensor in this state, eq. 28 or 29.
3. Replace the column of the sensor and the rows of the sensed states with all zero entries.
4. Repeat steps 2 and 3 until all sensors are placed to calculate coverage.

**Case 3: For a known number of sensors and a desired coverage**

1. Use flow chart to calculate the thresholded constrained contaminant tracking matrix for \( m = 1 \)
2. Determine column with maximum support and place a sensor in this state, eq. 28 or 29.
3. Replace the column of the sensor and the rows of the sensed states with all zero entries.

4. Increment \( m \) and repeat 1 through 3 while coverage is less than desired

### 4.4. Results

This section discusses results including the airflow field used, contaminant transport using the Markov matrices, and the sensor placement algorithm. Three airflow fields are generated using CFD. Based on these flow fields, Markov matrices are calculated and are shown to provide similar contaminant transport as the advection diffusion equation. Using these Markov matrices, sensors are placed for applications in CBW and IAQ applications using different sensing and placement constraints. A literature comparison with an aircraft cabin is discussed in Appendix A, which provided the initial motivation for this work.

#### 4.4.1 Preliminary analysis of air flow fields

4.4.1.1 Generation of the air flow fields

Three flow fields simulated by CFD have been chosen to illustrate this sensor placement methodology, figure 7. The cases are based on the IEA annex 20 geometry [45] with an inlet dimension of 0.168 [m] and an outlet dimension of 0.48 [m]. All the geometries were simulated at a Reynolds number of 5000. The inlet temperature is 293 [K], the obstructions represent people and generate 70 [W/m\(^2\)], the window heated the space with 100 [W/m\(^2\)], inward and all other boundaries are insulated. The inlet temperature and the heat flux of the people and the wall were taken from a similar sensor placement CFD problem from Liu et al. 2009 [29]. The 2\(^{nd}\) and the 3\(^{rd}\) case have 1 person and 2 people in the space respectively, and a heated window on the right boundary above the outlet, figure 8c and figure 8d. The CFD simulations used the turbulent RNG k-\( \varepsilon \) model, and all the
residuals were solved to $1e^{-6}$. Buoyancy in flow field 2 and flow field 3 was introduced by the Boussinesq approximation for density. Each simulation was discretized into 4480, 11140, and 10280 hexahedral elements for case 1, 2, and 3 respectively. These discretizations of the flow field produced sufficiently spatially converged values of the $u$ and $v$ components of velocity and the temperature when compared with meshes at higher resolutions.

The air flow field for the isothermal problem, figure 8b, has been compared with the data specified by the benchmark problem, IEA 1993, and the data that agreed well with the published literature. For the other flow fields the same fluid flow model was used that validated the turbulent Rayleigh-Benard problem. In the validation of an aspect ratio 1x1 Rayleigh-Benard cell, the Nusselt number was compared with correlations in literature [46], [47]. The data produced by these simulations agreed well with the published correlations.

4.4.1.2 Generation of the Markov Matrices

In order to examine the sensor placement problem Markov matrices are required to calculate the contaminant tracking matrix. The Markov matrices for the flow fields are calculated by the procedure explained in [37]. The domains were split up into 10800 (180x60 uniform) Markov states with an additional state added for the outlet for a total of 10801 states. The timestep associated with the Markov matrix for each problem is 0.1281 [sec], 0.3476 [sec], and 0.3569 [sec] for case 1, 2, and 3 respectively. To ensure the quality of the Markov matrices generated for this analysis, contaminant transport of the Markov matrices was compared to the transient passive scalar advection diffusion equation, equation 1. The initial condition of the comparison is a contaminant concentration of 1 is placed in the top half of the domain for each problem. The contaminant concentration at the centerline $y=1.5$ [m] at two different times for both the Markov method and passive scalar transport is shown in figure 9. The excellent comparison confirms the accuracy of the Markov matrices used in this analysis.
Figure 8: a) Description of the boundary conditions for the 3 example flow fields. b) flow field 1: Flow field of the IEA annex 20 isothermal problem, c) flow field 2: the IEA annex 20 problem with 1 person and a heated window, d) flow field 3: the IEA annex 20 problem with 2 people and a heated window.

Figure 9: Comparison between contaminant transport between the transient advection diffusion equation and the Markov Method for a) the isothermal IEA annex 20 problem, b) the flow field with 1 person, the flow field with 2 people.
4.4.2 Sensor placement results: CBW/TID and IAQ applications

In this subsection the sensor locations, coverage maps, and sensing times are investigated for the three flow fields with the addition of placement and sensing constraints. The effect of the adding the constraints on sensing time and the coverage maps are analyzed. Three different placement problems are addressed for each flow field; 1) the placement of sensors with no constraints, 2) placement of the sensors with a placement constraint outside of the occupied space, and 3) a placement constraint outside the occupied space along with a sensing constraint of sensing only the occupied space.

Placement problem 1) with no constraints is analyzed as a reference point for when constraints are added. Placement problem 2) is designed for critical application like CBW and TID applications. The expensive sensors are placed outside the space where occupants may damage these sensors, and the entire volume is observable by the sensor network to ensure that the source release location is observable. Placement problem 3) is designed for IAQ applications since the volume of interest is the occupied space where contaminants like CO₂ and VOCs levels need to be controlled. The placement of the sensors in problem 3) are also outside the occupied zone to limit the damage or movement of the sensors by occupants. Each problem is analyzed individually, and an analysis of all the problems is discussed at the end. To be consistent, a sensor accuracy is 1% is used for all the problems.

4.4.2.1 Sensor placement: no constraints

The problem of interest is case 3 described in section 2.4 (overview of algorithm), and answers the question “how quickly can the sensor network sense the entire space and respond with a known number of sensors?” The contaminant tracking matrix was calculated initially for a
response time equal to the timestep associated with the Markov matrix. The sensors were placed and the total coverage of the sensor network was evaluated. If the coverage of the sensor network was less than the volume of the space, the response time was incremented by the Markov timestep. This process continued until the domain was 100% observable by the sensor network. The response times, sensor locations, placement order, and the sensor coverage maps are displayed in figure 10.

![Figure 10](image)

**Figure 10:** This figure displays the minimum time at which a maximum of 2, 3, and 4 sensors can sense the entire domain with no constraints on placement or sensing. The sensor locations, placement order, coverage map, and sensing times of 2, 3, and 4 sensors for the three air flow fields are also shown. When a sensor is placed at the outlet the shape is placed outside the domain. For placement of 1 sensor, the sensor was always placed at the outlet. The response times for the isothermal, one obstruction, and two obstructions for the placement of 1 sensor is 146.8 [sec], 190.8 [sec], and 164.9 [sec] respectively.

Based on the results presented in figure 10, several observations can be made. In all cases there seems to be a dominant sensor that can sense the majority of the domain. The outlet seems to be a dominant place for a sensor for most of the cases. This result suggests that over long response times contaminant dispersed anywhere in the space eventually ends up in the outlet, which is why the outlet is often chosen as a sensor position. For all the flow fields, as the number of sensors
increase, the sensing time decreases. Introducing obstructions increases the sensing duration needed to sense the entire domain. The case with one obstruction/person takes the longest to sense. This may be due to the large buoyancy driven recirculation area on the right side of the domain. The most difficult areas to sense seem to be centers of recirculation. In the placement of multiple sensors, the location of the sensors seems to be clustered near regions of low air velocity.

4.4.2.2 Sensor placement: CBW and TID application

For CBW and TID applications, the sensor network needs to be able to have the ability to sense the entire domain such that any release scenario can be sensed as quickly as possible. The sensors may be expensive and cannot be damaged by occupants in the space. Figure 11 shows the sensor locations, response times, and coverage maps include a placement constraint of the sensors outside the occupied space. For each of the domains in figure 11, all the states in the domain are observable. For the placement of a single sensor the times and locations are the same as the no constraint case. Many observations can be made by comparing the no constraint situation with this situation, figure 10 and figure 11. The sensor locations, response times, and coverage maps may be the same if the sensors with no constraints are already placed outside the occupied space (for example, placement of 2 sensors for each flow field). Another situation where the response time is the same is when there are multiple sensor locations that can cover the state that are not yet observable (ex: placement of 3 sensors and 2 obstructions). Coverage maps are mostly similar, which is to be expected since the regions of low air speed do not change by simply adding placement or sensing constraints. The single obstruction flow field takes the longest to sense. The response time are either the same or longer, with some response times being substantially longer (ex: placement of 3 sensors for the 1 obstruction case).
4.4.2.3 Sensor placement: IAQ application

For IAQ applications, contaminant concentrations inside the occupied zone are more a concern than concentrations outside the occupied zone. For this reason, a sensing constraint is added to the contaminant tracking matrix such that only the occupied zone is sensed. Just like the CBW and TID applications, sensors also should not be placed where occupants can damage or move the sensors. For this reason, a location constraint is also added in this situation to constrain the sensors to locations outside the occupied space. Because the sensing volume is smaller, the response times are faster than the other two applications, figure 12. For this application sensors tend to be placed near the occupied zone or at the outlet. Also in this situation for the case 3 algorithm, the entire occupied space may become observable before the desired number of sensors is reached. In the examples of 4 sensors for both the isothermal and 2 obstruction cases, 3 sensors are required to sense the entire occupied space and a placement of a 4th sensor provides only redundant information. This is due to the telescoping nature of the placement algorithm. When considering the union of the column supports for multiple sensor locations, this situation may not occur.

4.4.2.4 Effect of constraints on coverage of multiple sensors

We next investigate the effect of placement and sensing constraints on the total coverage. The number of sensors and response times are known, while the coverage is determined. This corresponds to case 2 (as defined in section 2.4). A set of response time (approximately 30 second intervals) and up to 20 sensors were placed in the three air flow fields. For each of the response times and the number of sensors, the sensor coverage of the interested region is displayed in figure 13. The interested region for the no constraint and the placement constraint situations is the entire domain, while the placement and sensing constraints is only the occupied space.
Figure 11: This figure displays the minimum time at which a maximum of 2, 3, and 4 sensors can sense the entire domain with a placement constraint outside the occupied zone. The sensor locations, placement order, coverage map, and sensing times of 2, 3, and 4 sensors for the three air flow fields are also shown. When a sensor is placed at the outlet the shape is placed outside the domain.

Figure 12: This figure displays the minimum time at which a maximum of 2, 3, and 4 sensors can sense the entire domain with a placement constrain outside the occupied space and a sensing constraint of only the occupied space. The sensor locations, placement order, coverage map, and sensing times of 2, 3, and 4 sensors for the three air flow fields are also shown. When a sensor is placed at the outlet the shape is placed outside the domain. For a single sensor the response times are 90.1 [sec], 150.2 [sec], and 90.3 [sec] for the isothermal, 1 obstruction, and 2 obstructions flow fields respectively.
The results in figure 10 show that the response times for the isothermal flow field are faster than the obstruction non isothermal cases. More sensors placed in the domain results in faster response times. The addition of a placement constraint outside the occupied space affects the total sensing volume during shorter response times, figure 13, but not as much for longer response times. For each of the situations the isothermal flow field has a higher coverage fraction as compared to the 1 obstruction and 2 obstruction flow fields.

Overall, the algorithm takes the discrete form of the PF-operator and constructs a contaminant tracking matrix. The column support of the contaminant tracking matrix is used for
determining sensor positions. Placement of successive sensors maximizes the coverage of the sensor being places based on the states that are currently not observable by the sensor network. From the showcased flow fields, 4 major observations can be made from the algorithm.

1. The algorithm places sensors near regions with low air speed.
2. Regions where buoyancy dominants the air flow field are harder to sense than regions with higher Reynolds numbers and inertial driven air flow patterns.
3. The addition of placement constraints and sensing constraints has a greater effect on the shorter response times.
4. As more sensors are placed the response time decreases.

4.5. Discussion

We developed a simple algorithm that can rapidly identify optimal sensor location under various scenarios. We illustrated this methodology for three different steady state flow fields under different sensor placement constraints and sensing constraints that are relevant to CBW and IAQ applications. While the methodology is general, we choose to illustrate the approach using steady state flow fields for simplicity. In reality, flow fields experience perturbations by occupants moving around the space, changes in air flow conditions (boundary conditions), and are usually transient. The flow fields produced by these perturbations and the changes in supply air temperature, supply air flow rate, and changes in the building envelope surface temperatures or thermal loads are transient in nature. The elegance of considering the flow field (and the associated contaminant transport dynamics) from a dynamical systems perspective allows leveraging rigorous mathematical tools for sensor placement. More importantly, this methodology can be naturally extended to transient flow fields (as formulated in Equation 17 and also to uncertain (stochastic) flow fields [48]. This is the basis of two forthcoming papers that 1) investigate sensor locations for transient flow
fields or multiple operating conditions, and 2) incorporate uncertainty in boundary conditions and
the flow field to determine robust sensor locations.

4.6 Conclusions

This paper utilizes a dynamical systems approach for sensor placement for air flow fields. The algorithm uses the discrete form of the Perron-Frobenius operator (a Markov matrix) to construct a contaminant tracking matrix in which sensors are placed based on the largest column support in the contaminant tracking matrix. The addition of sensor accuracy along with placement constraints and sensing constraints are included in the algorithm. The algorithm considers all release scenarios of the contaminant (i.e. release in all states of the Markov matrix). We envision this algorithm to be an efficient alternative to current optimization/inverse based approaches to sensor placement in buildings.

The designed algorithm is showcased with a literature example and three additional flow fields based on different configurations of a reading room. The effect of different placement and sensing constraints on the sensor network response time is also investigated. The algorithm provides a simple and rigorous framework to place sensors for various scenarios. In particular, this framework has direct utility for placing sensor nets for applications involving sensing of chemical and biological weapons, transmission of infectious diseases, and indoor air quality. The contributions of this paper has the ability to help designers, engineers, and researchers better understand the coverage maps, response times, and number of sensors needed to effectively control the mechanical systems in buildings and design response and evacuation strategies during extreme events. The proposed framework for sensor placement can be easily extended to account for various sources of uncertainty in the building environment.
4.7 Nomenclature

$A$ A set of Markov states that define the release scenarios

$B$ A set of Markov states that define the release scenarios not covered by previously placed sensors

$C_{\bar{k}}$ The output matrix for a set of sensors described by $\bar{k}$

$D$ Contaminant diffusivity in the fluid medium

$\bar{k}$ The vector of sensor locations

$N$ Number of sensors

$n$ Number of Markov states

$n_{loc}$ The number of Markov states that are constrained by a location constraint

$n_{sen}$ The number of Markov states that are constrained by a sensing constraint

$\sigma_{\bar{k}}$ The observability gramian with sensors placed at the locations in the set $\bar{k}$

$P$ Steady state Markov matrix

$P(t)$ Time dependent Markov matrix

$P^{(i,j)}$ Markov matrix from time $t_i$ to time $t_j$ with $t_i < t_j$

$Q^{(\tau)}$ Contaminant tracking matrix with response time $\tau$

$\tilde{Q}^{(\tau)}$ Thresholded contaminant tracking matrix with response time $\tau$

$\tilde{Q}^{*(\tau)}$ Thresholded contaminant tracking matrix with response time $\tau$, with the observable states removed by previously placed sensors

$r^{(a)}$ An individual release scenario for the $a^{th}$ Markov state

$r^{(A)}$ A set of release scenarios with the release states being $A_i$

$r^{(all)}$ All possible release scenarios

$S_\phi$ Continuous contaminant source term
$S_{t0}$  Discrete contaminant source term at time $t_0$

$s_{source}$  Contaminant source release rate

$s^{(a)}$  The support of the $\mathbf{k}$ sensor locations for an individual release scenario for at the $a^{th}$ Markov state

$s^{(A)}$  The support of the $\mathbf{k}$ sensor locations for a set of release scenarios with the release states being $A_i$

$s^{(all)}$  The support of the $\mathbf{k}$ sensor locations for all possible release scenarios

$t$  Time

$U$  Air velocity vector with components $\{u, v, w\}$

$u$  The $x$-component of the air velocity vector

$V$  Volume

$V_{\omega k}$  Volume of the $\omega_k$ state

$v$  The $y$-component of the air velocity vector

$w$  The $z$-component of the air velocity vector

$x$  The position vector with directions $\{x, y, z\}$

$x$  The $x$-coordinate direction

$y$  The $y$-coordinate direction

$z$  The $z$-coordinate direction

$\Delta t$  Timestep associated with the Markov matrix

$\varepsilon_{acc}$  Accuracy threshold for the contaminant tracking matrix

$\lambda_{B_k}(\mathbf{x})$  The indicator function of the $\mathbf{k}$ sensor locations

$\mu_{detect}$  The contaminant concentration detected at the sensor

$\mu_{source}$  The source contaminant concentration

$\mu_t$  Contaminant concentration field at time $t$
\( \mu_k^t \)  
Contaminant concentration in Markov cell \( k \) at time \( t \)

\( \tau \)  
Sensing duration/time

\( \varphi \)  
Continuous contaminant concentration

\( \varphi_t \)  
Continuous contaminant concentration at time \( t \)

\( \varphi_{t+\tau} \)  
Continuous contaminant concentration at time \( t + \tau \)

\( \omega_k \)  
The \( k \)th Markov cell in the domain \( \Omega \)

\( L(\cdot) \)  
Perron-Frobenius (PF) operator

### 4.8 References


4.9 Appendix A: Aircraft literature comparison

In this appendix a sensor is placed based on the methodology in this paper using a published Markov matrix [40] and compared with general recommendations from literature [27], [49]. The non-zero entries of the Markov matrix from C. Chen et al. 2014 [40] are shown in figure 3. The Markov matrix was generated for a 4 seat and 1 aisle wide aircraft cabin\(^4\). The flow field was generated by the RNG k-\(\varepsilon\) model for 3 of the rows on the aircraft cabin. The transition time associated with the Markov matrix is 4 seconds. The accuracy chosen for these calculations is 0.002, which based on the release scenario of Zhang et al 2007 [49]. The only constraint introduced in problem is that the sensor cannot be placed at the outlet, since Zhang et al. 2007 [49] and Mazumdar et al. 2008 [27] did not consider the outlet as an option to place the sensor.

Using the sensor placement algorithm provided in this paper, a single sensor as well as 2 sensors are considered. For a single sensor to sense all the Markov states in the scenarios using the aircraft, the sensor can be placed in the aisle of row 2 and has a response time of 16 seconds, figure 14a. For 2 sensors to sense all the Markov states in the aircraft, the sensors are also placed in the aisle of rows 1 and 2, figure 14b. The response time of the sensors is 8 seconds, which is less than the response time of a single sensor. The thresholded and constrained contaminant tracking matrix was computed for a sensing time of \(\tau = \{4,8,12,16\}\) seconds. The column support in terms of the sensing volume for each Markov state and each time is shown in figure 15. The results show that as time progresses contaminates will tend to gravitate towards the center of the aircraft.

Based on this analysis there are two observations which are consistent with the conclusions of Zhang et al 2007 [49] and Mazumdar et al. 2008 [27]. These authors state that the best location for a single sensor is in the center of the aircraft, and that the response time for two sensors is less

\(^4\) Please see the original articles for the geometry of the aircraft cabin [40].
than the response time for a single sensor. Heuristically, these observations are seen in figure 14 and figure 15 that serve as motivation toward the next examples.

Figure 8: Sensor placement for a 4 seat and 1 aisle aircraft. a) Placement of 1 sensor that can sense the entire cabin. b) Placement of 2 sensors that can sense the entire cabin.

Figure 9: Percent sensing volume for a sensing time of a) 4 seconds, b) 8 seconds, c) 12 seconds, and d) 16 seconds for a 4 seat and 1 aisle aircraft cabin for a Markov matrix with a timestep of 4 seconds.

---

We are limited to make quantitative comparisons for this data as the published Markov matrix does not distinguish Markov states at different heights of the aircraft. The cabins for both Zhang et al. 2007 [49] and Mazumdar et al. 2008 [27] are 7 seat 2 aisle wide cabins, while the published Markov matrix is from a 4 seat 1 aisle wide aircraft cabin. The sensing time also seems short since the published Markov states are extremely large (the size of a seat in the aircraft cabin).
CHAPTER 5. QUANTIFYING MECHANICAL VENTILATION PERFORMANCE: THE CONNECTION BETWEEN TRANSPORT EQUATIONS AND MARKOV MATRICES

A paper accepted by *Building and Environment*

Anthony D. Fontanini, Umesh Vaidya, Alberto Passalacqua, and Baskar Ganapathysubramanian

5.1. Abstract

Most people spend approximately 90% of their lives indoors. Thus, designing effective ventilation systems is essential to mitigating problems with indoor air quality. The measures of mechanical ventilation design performance considered in this study are age of air, air residual life time, air residence time, and ventilation effectiveness. This paper presents two different methods to help quantify these measures. The first method is based on transport equations, where a continuous representation of these quantities are calculated. The second method is based on Markov matrices, where a discrete representation of these quantities are calculated. We show 1) how both the continuous and discrete methods are related and 2) that the age of air and residual life time are adjoints. A new transport equation for residual life time along with methods for these quantities using Markov matrices are established. The two approaches are validated and compared using previously established experimental data. The results show that both approaches provide similar results. Using these techniques allows for the quantities of residual life time and residence time to be integrated into the design processes. This paper provides a simple framework that enables designers to get a comprehensive picture of the ventilation systems they design.
5.2. Introduction

As people in North America and Europe spend 90% of their lives indoors [1], [2], the quality of the air they breathe is essential to their health. Symptoms from exposure to air pollutants in the short term may lead to headaches, dizziness, fatigue, and sinus irritation [3], while long term effects may lead to respiratory disease, heart disease, and cancer [3]. The pollutants may have a greater effect on people suffering from asthma, the elderly, and children. One of the easiest ways to reduce exposure to potentially harmful pollutants is through effective ventilation and providing fresh air to the occupied zone of buildings.

Although there are many ways to ensure proper ventilation, a popular method that has been used in many applications is through measuring or calculation of the age of air [4]–[16]. The concepts of internal age of air, the air residual lifetime, and air residence time was first explored by Sandberg and Sjoberg [4] using tracer gas experiments. The internal age of air, \( \tau_i \), is defined as the time required by a discrete fluid element to reach some point in the space since entering the room. The air residual life time, \( \tau_{rl} \), is defined as the time required by a discrete fluid element to leave the room from an initial starting point in the room. The air residence time is the total time a fluid element spends in the room by passing through a given point in space. The concepts of age of air, residual life time, and residence time is pictorially represented in figure 1.

Experimentally, age of air can be measured through tracer gas experiments [4], [5], [9], [13], [17] through a decay method, a source method, and a pulse method. These experimental techniques essentially determine the age of air through the response of the tracer gas concentration as the concentration is allowed to decay, applied as a constant source, and applied as a single pulse, respectively. Numerically, age of air has been calculated using an advection-diffusion partial differential equation (PDE) [6], [9]–[13], [15], [18]. This transport equation describes a fluid element being advected by an underlying flow field and that diffuses at a rate based on the laminar
and turbulent Schmidt numbers along with laminar and turbulent viscosities. During advection, a source term (age) is added until the element leaves the space. Although the transport equation for age of air has been fairly well established to provide accurate estimates, the corresponding transport equation for air residual lifetime has not been well established. Both the age of air and the residual life time is needed to calculate the air residence time.

Figure 1: Description of the age of air, $\tau_i$, the air residual life time, $\tau_{rl}$, and air residence time, $\tau_r$ for a fluid element that passes through a given point, $x_{pt}$, in the building.

The computational approaches above are based on solving a PDE for the transport of contaminants in a room or building zone. Some recent work has been performed on a different method for contaminant transport, the Markov method [19]–[24]. The Markov method is a useful method for contaminant transport because it has been show to be faster than the PDE approach [23], [24]. The real-time prediction of contaminant propagation provides promise for demand response HVAC control. Applications of Markov matrices have also been shown in the areas of optimal sensor placement algorithms [25]–[27].

In this paper, the structure of Markov matrices is exploited to provide performance measures for use in mechanical ventilation system design. The purpose of this paper is to 1) establish two different approaches to calculate age of air, air residual life time, air residence time, and ventilation effectiveness, 2) show how both techniques are related, and 3) establish the mathematical
relationship between residual life time and age of air. The numerical developments in this paper addresses the major gaps in analyzing mechanical system ventilation performance. These gaps in knowledge are 1) the need for a transport equation that describes the residual life time and subsequently the calculation of residence time and 2) the connection between continuous and discrete methods for calculating measures of mechanical system ventilation performance.

The two methods extended and developed in this paper are based on PDEs and Markov matrices. A new transport equation for residual life time is established that gives similar results to the Markov method for residual life time. The results of both the PDE method and the Markov method are compared with experimental data. The work presented in this paper provides a link between the PDE method and Markov methods and closes the loop for computationally calculating age of air, air residual life time, and air residence time to estimate aspects of mechanical ventilation system performance.

5.3. Methods

This section explains how a Markov matrix is used for contaminant transport, investigates the structure of the Markov matrices calculated for contaminant transport, and develops two different methods to quantify measures of mechanical ventilation system performance.

5.3.1. Mathematical preliminaries: contaminant transport using Markov matrices

A common well established computational method for tracking a particulate contaminant or a gas, $\phi$, in the indoor environment under some flow field, $U$, is by solving the advection-diffusion partial differential equation (PDE) [28]–[34], eq. 1.

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (U\phi) + \nabla^2 (D\phi) = S_\phi$$

The advection-diffusion equation, shown to be a generalization of the infinite dimensional Perron-
Frobenius operator [35], advects and diffuses a concentration of a contaminant on a continuous vector field from one time instant to another. In the discrete finite dimensional case, we consider a finite discretization of the room or building into a set of cells \( \{\omega_1, \ldots, \omega_n\} \). These cells are also called states. For a specific cell \( \omega_k \) the volumetric concentration of the contaminant is calculated, eq. 2.

\[
\psi^{k}_t = \frac{1}{V_{\omega_k}} \iiint_{\omega_k} \phi(x, y, z, t) \, dV, \quad k = 1, \ldots, n
\]  

(2)

The discrete finite dimensional Perron-Frobenius operator [25], [27] maps these volumetric average concentration of a set of cells across a time period, \( \Delta t \). This map is known as a Markov matrix, eq. 3. A Markov matrix describes the probabilistic transition from an initial state to all other states during a timestep, \( \Delta t \).

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

(3)

A Markov matrix is a probability matrix, which implies that the entries of the matrix are all non-negative, eq. 4.

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

(4)

Since a Markov matrix describes a probabilistic process, the row sum and/or the column sum of the matrix must be equal to 1.0. The rows of a right stochastic matrix sum to 1, eq. 5, and describes a probabilistic transition from an initial state \( V \) to states in the domain.

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

(5)

The columns of a left stochastic matrix sum to 1, eq. 6, and describes a probabilistic transition from states in the domain into a given final state \( W \).

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

(6)

Using a right stochastic or a left stochastic matrix, the propagation of a contaminant concentration at time \( t \) to a time \( t + \Delta t \) is calculated by a simple matrix vector product, eq. 7.
For the calculation of Markov matrices for a given flow field two approaches have been developed: a flux based approach [21], [22] and a set theory based approach [24]. In this paper, the method for calculating the Markov matrix is not important, as long as the appropriate flow physics is contained in the Markov matrix. The physics of interest are accurate representations of advection and diffusion of air within a building, with the spatially varying diffusion rate based on the laminar and turbulent Schmidt numbers along with laminar and turbulent viscosities\(^1\), eq. 8.

\[
\begin{align*}
D &= \left( \frac{\mu}{\sigma_l} + \frac{\mu_t}{\sigma_t} \right) \\
\end{align*}
\]

5.3.2. Canonical form of absorbing Markov matrices

An absorbing Markov matrix is defined as having at least 1 absorbing state, and there is a non-zero probability that a non-absorbing (transient) state can reach an absorbing state in a finite number of steps. Figure 2 is an example of an absorbing Markov matrix. Absorbing states (like state 3, fig. 2) is a state that is impossible to leave. Absorbing states have a single entry of 1.0 on the diagonal, which implies that a contaminant in an absorbing state has a probability of 1.0 of staying in the current state in the next timestep. States that are not absorbing states are called transient states (like state 1 and state 2, fig. 2). For ventilated systems the outlets serve as the absorbing states, and all other states are transient states. Due to diffusion each state is guaranteed to be connected to an outlet.

\(^1\) The particular method used to calculate the Markov matrices for this study involves 4 steps. 1) Initializing a given cell \(a_k\) with a value of 1.0 and the rest of the domain to be 0.0. 2) Solving equation 1 with a diffusivity equivalent to equation 8 for the desired time step of the Markov matrix. 3) Collecting the non-zero scalar concentrations into the \(k^{th}\) row of a sparse Markov matrix, \(P\). 4) Performing this process for each \(k = 1: n\) to produce the transition probabilities of each row of the Markov Matrix.
Figure 2: An example of an absorbing Markov matrix. State 3 is an absorbing state, while state 1 and 2 are transient states.

Absorbing Markov matrices have a standard ‘canonical’ form, eq. 9, where the transient states (TR) are grouped together and absorbing states (AB) are grouped together.

\[
P = \begin{bmatrix}
G_{(TR) (TR)} & R_{(TR) (AB)} \\
0 & d_{(AB) (AB)}
\end{bmatrix}
\]

(9)

The matrix \( G \) is a \((n - n_o) \times (n - n_o)\) matrix that contains all the transient states, \( R \) is a \((n - n_o) \times n_o\) matrix that specifies which states will transition to an absorbing outlet (air return) state, the matrix \( d \) is a \(n_o \times n_o\) identity matrix for the outlet states, and \( 0 \) is a \(n_o \times (n - n_o)\) matrix containing all zeroes. The above decomposition of the finite dimensional Markov matrix into upper triangular matrix can also carry over to the infinite dimensional Perron-Frobenius (P-F) operator.

We refer the reader to [36], [37] for the details on the decomposition of P-F operator.

5.3.3. Quantifying measures of IAQ

In this section, two different methods for quantifying measures of mechanical system ventilation are developed. The first method is based on the more traditional transport equations (PDE method). The second method is using the Canonical form of absorbing Markov matrices (Markov method). Both methods are used to quantify internal age of air, air residual life time, air residence time, and ventilation efficiency.
5.3.3.1 PDE method

Computationally, age of air has been calculated using a steady state transport PDE [6], [9], [13], [18], [38], [39], eq. 10.

\[ \nabla \cdot \left( \rho \mathbf{U} \tau_i - \left( \mu_l / \sigma_t + \mu_t / \sigma_t \right) \nabla \tau_i \right) = 1 \]  
(10)

Physically, this equation represents a fluid element gaining a source concentration (age) while being advected along the velocity field, \( \mathbf{U} \), and diffusing at a rate proportional to the effective viscosity at each point in the room. Some references use \( \sigma_l = \sigma_t = 0.7 \) [11], [13], while recent work gives an alternate method for calculating the turbulent Schmidt numbers in stratified flows [40]. The boundary conditions of this equation are \( \tau_i = 0 \) at all inlets and \( \partial \tau_i / \partial x = 0 \) at all the outlets and the walls.

Although this transport equation has been used extensively, a transport equation for air residual life time has not been established. Equation 11, is similar to the age of air equation, except the velocity field has been reversed. This transport equation represents the movement of a passive discrete fluid element from the outlet into the room along the reverse velocity field, \( -\mathbf{U} \). The fluid element gains a source concentration (residual life) and diffuses at the rate of the effective viscosity at each point in the room as the fluid element moves further away from the outlet. The boundary conditions of this equation are \( \tau_{rl} = 0 \) at all outlets and \( \partial \tau_{rl} / \partial x = 0 \) at all the inlets and the walls.

\[ \nabla \cdot \left[ \rho (-\mathbf{U}) \tau_{rl} - \left( \mu_l / \sigma_t + \mu_t / \sigma_t \right) \nabla \tau_{rl} \right] = 1 \]  
(11)

**Remark:** Mathematically, this equation is the adjoint [41] of equation 10. Looking at Figure 1, it is clear that \( \tau_{rl} \) can be calculated by starting at the outlet and moving backwards in time. The adjoint equation formally captures this intuition.

Once the internal age of air and the residual life time fields have been calculated, other quantities of mechanical system ventilation performance can be calculated. The residence time relates both the age of air and the air residual life time, eq. 12.
Another measure for mechanical ventilation system performance is ventilation effectiveness, eq. 13. Ventilation effectiveness [17] is calculated by the time constant of the room or building, \( \tau \), and the age of air. The ventilation effectiveness provides a notion of the ventilation systems ability to provide fresh air into the space [17]. A value of 1.0 indicates that the system is well mixed, a value less than 1.0 indicates less than perfect mixing (stagnation or short-circuiting flow), and a value greater than 1 indicates the space is experiencing air flow similar to displacement ventilation or plug flow [17].

\[
\varepsilon_t = \frac{\tau}{\tau_i}
\]

5.3.3.2 Markov method

In the context of absorbing Markov matrices, the residual life time is related to the fundamental matrix. The fundamental matrix of an absorbing Markov matrix describes the average number of times state \( i \) visits state \( j \) before being absorbed [42], eq. 14.

\[
N = N_{ij} = (I - G)^{-1}
\]

The row sum of the fundamental matrix represents the total number of steps taken by a given state before it is absorbed. The row sum multiplied by the timestep (associated with the Markov matrix) provides the time duration until that state is absorbed (residual life time), eq. 15.

\[
\tau_{rl} = \Delta t \sum_{j=1}^{n-n_o} N_{ij} = \Delta t \; N \mathbf{1}
\]

The calculation of \( \tau_{rl} \) in equation 15 involves taking an inverse of a potentially large matrix \( G \) which computationally takes \( O[(n - n_o)^3] \) operations [43]. Since this becomes computationally demanding for large matrices, a linear system of equations can be formulated by substituting equation 14 into equation 15.

\[
\tau_{rl} = \Delta t (I - G)^{-1} \mathbf{1}
\]
The result is a linear system for the unknown $\tau_{rl}$ with coefficient matrix $(I - G)$ and solution vector $\Delta t \mathbf{1}$. Solving the linear system is computationally more efficient (both memory and flops) and takes $\sim O((n - n_o)^2)$ operations using standard linear solvers.

In the context of Markov matrices, reverse advection is performed by using the transpose of the Markov matrix, $\bar{P} = P^T$ (this is also equivalent to taking the adjoint [41] and in the infinite dimensional setting this adjoint operator is also known as Koopman operator. Spectral analysis of Koopman operator is used for dynamical system analysis and for solving problems involved in building systems applications [44]–[46]). The corresponding canonical form for the reverse advection Markov matrix is shown in equation 17.\(^2\)

\[
\begin{bmatrix}
\bar{G} & \bar{0} \\
\bar{R} & \bar{d}
\end{bmatrix}
\]

The matrix $\bar{G}$ is a $(n - n_i) \times (n - n_i)$ matrix that contains all the transient states, $\bar{R}$ is a $n_i \times (n - n_i)$ matrix that specifies which states will transition to the absorbing inlets states, the matrix $\bar{d}$ is a $n_i \times n_i$ identity matrix for the inlet states, and $\bar{0}$ is a $n - (n_i \times n_i)$ matrix containing all zeroes. Similar to the forward advection case, the same techniques in equations 14-16b can be used on the reverse advection Markov matrix in equation 17 to produce a linear system of equation for the internal age or air, $\tau_i$, eq. 18.

\[
(I - \bar{G})\tau_i = \Delta t \mathbf{1}
\]  

Once the residual life time field and internal age of air field are calculated by equation 16b and equation 18 respectively, residence time and ventilation effectiveness can be calculated using

\(^2\) The rows containing the transitions for the inlets and the outlets may need to be swapped after the transpose.
the equations 12 and 13 respectively. $\tau_i$ has an infinite dimensional analog obtained using the infinite dimensional Perron-Frobenius operator. In particular, this infinite dimensional analog can be obtained by solving the linear advection PDE, i.e., eq. (1), or discrete time approximation of the PDE. In [36], this infinite dimensional analog is rigorously defined and is used as a stability certificate for verifying weaker notion of almost everywhere stability of the absorbing state.

5.4. Flow field and validation

The room considered in this study is a 4.2 (m) x 3.0 (m) x 3.6 (m) room, fig. 3, that has been used in previous age of air studies [9], [13]. Both the inlet and outlet dimensions are 0.2 (m) x 0.3 (m). The longer side of the inlet and outlet is along the z-direction, figure 3. The bottom of the inlet and outlet are positioned symmetrically along the center of the z-axis. The bottom edge of the inlet is 2.05 (m) above the floor. The ventilated room has an air change rate of 8 (ACH), which corresponds to the room time constant $\tau = 0.125$ (hr) and a constant inlet air speed of 1.68 (m/s). The air flow in this space is considered to be isothermal [9].

The air flow in the ventilated room was computed by solving the Navier-Stokes equations. The Reynolds number for this based on the inlet height of 0.2 (m) is approximately 21,800 and is considered to be turbulent. The Reynolds stresses were modeled using the RNG k-ε model. The turbulent conditions at the inlet are based on a turbulent intensity of 14%. The turbulent kinetic energy at the inlet was calculated by $k_{inlet} = \frac{3}{2} [U_i] l_i^2$. The kinetic energy dissipation rate at the inlet was calculated by $\epsilon_{inlet} = (k_{inlet})^{1.5} / (0.005 \sqrt{A_{inlet}})^{\frac{3}{3}}$.

3 For more information about the room setup, numerical simulations, and the experiments the readers are referred to work of M. Bartak et al., 2001 [9].
The geometry, fig. 3, was discretized into 2 different meshes to investigate the effects of spatial convergence. Due to symmetry of the problem, only half of the domain was discretized. The first (coarse) discretization contains 22,680 hexahedra elements with sides approximately 0.1 (m), fig. 4a. The second (fine) discretization contains 181,440 hexahedra elements with sides approximately 0.05 (m), fig. 4b.

The Navier-Stokes equations, turbulent transport equations, age of air transport equation (eq. 10), and the residual life time transport equation (eq. 11) were solved on both discretization to a residual tolerance of 1e-6. After the simulations converged, data was sampled along the symmetry
plane (z=0) at three different x locations [x=1.13 (m), x=2.20 (m), and x=3.2 (m)] from the floor to the ceiling. The normalized age of air data from the simulations along with the experimental and numerical data from Bartak et al. [9] can be seen in figure 5. The normalizing constant is 538 seconds$^4$. From these results, figure 5, the age of air calculated in this study is fairly accurate compared to the experimental data, and in generally performs better than the previous study. This reason may be due to the increased discretization used in the current study as there is a decrease in the age of air as the discretization is increased, figure 5. Since the coarse mesh representation performs reasonably well compared to the fine mesh discretization, the coarser representation will be used to display the results for the rest of the paper.

![Figure 5](image.png)

Figure 5: Normalized experimental and numerical age of air based on the equation 9 along three x-locations, a) x = 1.13 (m), b) x = 2.2 (m), and c) x=3.2 (m).

5.5. Results

This section provides comparisons between experimental data and numerical results for both the PDE method and Markov method for the quantities of interest. The air residual life time and age of air are compared first as they are quantities directly solved by the transport PDEs (eq. 10 and 11)

---

$^4$ This normalizing constant of 538 seconds is used for all the normalized data presented in this paper.
or a linear system from the Markov matrix (eq. 16b and 18). Since air residence time and ventilation efficiency are post-processed values, these quantities are compared after the age of air and residual life time comparison.

5.5.1. Air residual life time and age of air comparisons

The age of air and air residual life time are calculated by both the PDE method and the method using Markov matrices. Contour plots show the normalized iso-contours of the age of air and residual life time in figure 6. The age of air, figure 6a and 6c, is small near the inlet and increases as the jet carries the air all the way to the opposite all. Once the air hits the opposite wall, the air spreads out into the rest of the domain. For the regions away from the jet, the age is roughly constant. The normalized mean age of air in the room for both the PDE and Markov methods is calculated to be 0.97 and 1.00 respectively. The residual life time of the air is small near the ceiling, figure 6b and 6d, which suggests that the air needs to reach the top center part of the room before it has a chance to leave. The areas near the far wall \([z \approx 1.8 \,(m)]\) has a higher residual life time as the air slowly circulates back to the center of the space. The normalized mean residual life time in the room for both the PDE and Markov methods is calculated to be 1.00 for both methods. Sensitivity of the results to a change in the timestep associated with the Markov matrix can be see in appendix A.

The results of the Markov method and the PDE method for simulating internal age of air and air residual life time is compared in figure 7. The results show that both the PDE method and the Markov method provide fairly good agreement with the experimental age of air data in figure 7a – 7c. At these locations in the domain the Markov method performs slightly better than the PDE method. The largest differences in age of air for the two methods is near the ceiling. For air residual life time both the PDE approach and the Markov method essentially give the same result, figure 7d – 7f. Based on these results, both the PDE method and the Markov method seem to provide
reasonably good estimates for age of air and roughly equivalent measures of calculating air residual life time.

Figure 6: Normalized iso-contours of the a) the internal age of air field using the Markov method, b) the residual life time field using the Markov method, c) the internal age of air field using the PDE method, d) the residual life time field using the PDE method.
Figure 7: Normalized internal age of air comparisons at different x-locations along the z symmetry plane for a), b), and c). Normalized comparisons of air residual life time at different x-locations along the z symmetry plane for d), e), and f).

5.5.2. Air residence time and ventilation effectiveness

The previous subsection showed that the Markov method and the PDE methods provide similar estimates of internal age of air and air residual life time. This section compares the performance of the Markov method and the PDE method for measures of IAQ based on the age of air and air residual life time. The two quantities of interest are the residence time and ventilation effectiveness, figure 8. The regions of the space with highest residence times are near the far wall \([z \approx 1.8(m)]\), figure 8a and 8c. This is not surprising as this region has the largest residual life times and age of air. The normalized mean residence time in the room for both the PDE and Markov methods is calculated to be 1.97 and 2.00 respectively. For ventilation effectiveness the region near
the jet has very high ventilation effectiveness, due to the high air speed of this region, figure 8b and 8b. The rest of the domain is a ventilation effectiveness of about 0.8. The mean ventilation effectiveness in the room for both the PDE and Markov methods is calculated to be 1.02 and 0.90 respectively.

Residence time and ventilation efficiency calculated using the Markov method and the PDE method are compared in figure 9. The two methods essentially provide the same results for residence time with the exception of regions near the ceiling, figure 9a – 9c. The difference between the two techniques is due to the differences seen in the internal age of air in figure 7. For ventilation effectiveness, the two methods roughly provide the same estimates of ventilation effectiveness, figure 9d – 9f. The Markov method shows a slight decrease in the peak for each location at the center of the jet. Once again, a slight difference in the ventilation effectiveness is shown near the ceiling, where the Markov method predicts a slightly higher ventilation effectiveness. Overall, the two methods provide similar predictions for both the residence time and the ventilation effectiveness.

5.6. Discussion

This study used a validated benchmark problem to explore the connection between continuous and discrete methods for calculating performance measures of mechanical system ventilation. We show that these two alternate approaches yield equivalent and accurate results. A new transport equation for residual life time is developed. The new equation is the adjoint of the age of air transport equation. This mathematical connection (in both the continuous and discrete setting) opens up the utility of a variety of mathematics tools in linear control theory that can be applied to the analysis of mechanical ventilation systems. The introduction of an equation for residual lifetime also allows for residence time to be calculated.
Figure 8: Normalized iso-contours of the a) the residence time field using the Markov method, b) the ventilation effectiveness using the Markov method, c) the residence time field using the PDE method, d) the ventilation effectiveness using the PDE method.

Just as the transport equations provide a continuous representation of these metrics, Markov methods are shown to provide the corresponding discrete representation of these metrics. We identified the fact that previous studies [9], [13] calculating the age of air over predicted the values due to coarse mesh resolutions. These advances show the importance of spatial convergence and provide a complete picture of the relationships between continuous and discrete methods for calculating mechanical system performance.
The two different approaches developed in this paper have their own advantages and disadvantages. The advantage of using Markov based methods is their broad applicability beyond just mechanical system performance. If a Markov based method is utilized to quantify mechanical system performance, then the resultant data can be subsequently used to determine optimal sensor locations [47], perform fast contaminant transport [22], [23], [48] and build estimators for demand control, and for rapid contaminant source detection. A potential disadvantage, however, is the increased computational effort needed in calculating the Markov matrices. In contrast, for the PDE method, the passive scalar transport equations for age of air and residual life time calculations can be easily solved for when the flow field is being solved with very little additional computational effort.
overhead [49], [50]. The disadvantage, in turn, is the lack of broader applicability beyond mechanical ventilation system performance assessment (like contaminant transport, sensor placement, and source identification) which Markov based methods can naturally perform.

5.7. Conclusions

This paper expands and develops two different methods for quantifying mechanical ventilation system performance. The first method uses transports equations and the second method uses the structure of absorbing Markov matrices to calculate age of air and air residual life time. The methods are validated with experimental tracer gas data for a ventilated room. We have shown that the Markov method is the discrete counterpart of the PDE based method, that the residual life time and age of air are related through adjoint equations, and have established a transport equation (and its discrete counterpart) for residual life time to quantify mechanical ventilation system performance. Both Markov (discrete) and PDE (continuous) methods show similar results for age of air and residual life time along with the processed quantities of residence time and ventilation effectiveness. Based on these results both methods are valid and useful for calculating age of air, air residual life time, air residence time, and ventilation efficiency. We advocate the use of the PDE method when only system performance assessment is needed, as it has minimal computational overhead over the CFD simulation required to compute the flow field. If a more integrated approach to building analysis is taken -- including contaminant release scenario assessment, emergency response planning, or sensor placement strategies – then we advocate the use of the Markov method approach. This work also provides approaches for integrating residual life time and residence time into the mechanical system design process. Using these methods designers, architects, and researchers can obtain a better understanding of the ventilation performance of rooms and buildings they design, thus ensuring safety and comfort while potentially minimizing energy usage.
5.8 Nomenclature

$A_{\text{inlet}}$ Area of the inlet

$D$ Diffusivity of the contaminant in air

$d$ An identity matrix of size $n - n_o \times n - n_o$

$\bar{d}$ An identity matrix of size $n - n_t \times n - n_t$

$G$ The matrix of the transition probabilities between transient states of the forward transport Markov matrix

$\bar{G}$ The matrix of the transition probabilities between transient states of the reverse transport Markov matrix

$I$ An identity matrix

$I_t$ The turbulent intensity at the inlet

$k_{\text{inlet}}$ Turbulent kinetic energy at the inlet

$N$ The fundamental matrix of the forward transport Markov matrix

$n$ The number of Markov states

$n_o$ The number of outlet Markov states

$P$ The forward transport Markov matrix

$\bar{P}$ The reverse transport Markov matrix

$R$ The matrix of the transition probabilities between transient states and absorbing states for the forward transport Markov matrix

$\bar{R}$ The matrix of the transition probabilities between transient states and absorbing states for the reverse transport Markov matrix

$S_\phi$ Spatially dependent contaminant source term

$t$ Time

$U$ Air flow velocity field
The air velocity magnitude at the inlet

\[ |U_1| \]

Volume of state \( \omega_k \)

\[ V_{\omega_k} \]

The x direction

\[ x \]

The position vector \((x, y, z)\)

\[ \mathbf{x} \]

The y direction

\[ y \]

The z direction

\[ z \]

The timestep associated with the Markov matrix

\[ \Delta t \]

Turbulent kinetic energy dissipation at the inlet

\[ \varepsilon_{\text{inlet}} \]

Ventilation effectiveness

\[ \varepsilon_{i} \]

Dynamic viscosity

\[ \mu \]

Dynamic turbulent viscosity

\[ \mu_t \]

The density of air

\[ \rho \]

The laminar Schmidt number

\[ \sigma_l \]

The turbulent Schmidt number

\[ \sigma_t \]

The room time constant

\[ \tau \]

The internal age of air

\[ \tau_i \]

The internal age of air normalized by 538 (sec)

\[ \tau_i \]

The air residence time

\[ \tau_r \]

The air residence time normalized by 538 (sec)

\[ \tau_r \]

The air residual life time

\[ \tau_{rl} \]

The air residual life time normalized by 538 (sec)

\[ \tau_{rl} \]

Continuous contaminant concentration

\[ \phi \]

Discrete volumetric average concentration in state \( k \) at time \( t \)

\[ \psi_{\omega_k}^k \]

The \( k^\text{th} \) Markov state

\[ \omega_k \]
A matrix of size $n_o \times n - n_o$ containing all zeroes

A matrix of size $n_t \times n - n_t$ containing all zeroes

A vector with all elements equal to 1.0

The magnitude operator

5.9 References


5.10 Appendix A: Timestep sensitivity of age of air

In order to test the sensitivity of the Markov matrix to the timestep used to calculate the matrix, three different Markov matrices were calculated with a timestep of 0.1 (sec), 0.5 (sec), and 1.0 (sec). The mean Courant numbers associated with these timesteps are 0.13, 0.64, and 1.3 respectively. The internal age of air field was sampled along the same locations presented in figure 5, and the results are shown in figure 10. The results show that the Markov method for calculating age of air provides a fairly accurate profile at these locations compared to the experimental values. The results of the two larger timesteps of 0.5 (sec) and 1.0 (sec) are almost overlaid on top of each other, while the smallest timestep of 0.1 (sec) increases the age of air at these locations. This may be due the small timestep. During the calculation process of the Markov matrix, a small timestep results in the majority of the states having large values along the diagonal. These small values on the off diagonal may not fully capture the transition probabilities. Based on these results, it is recommended by the authors that a mean Courant number larger than 0.5 is needed to capture the transition probabilities needed for these calculations.

Figure 10: The normalized internal age of air comparisons for different timesteps associated with the Markov matrix. The locations of the are along the z-axis symmetry plane from floor to ceiling at an x-location of a) x=1.13 (m) b) 2.20 (m), and c) 3.2 (m)
CHAPTER 6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In this thesis methods for calculating the discrete form of the PF-operator (Markov matrix) have been established and applications have been show for sensor placement problems and quantifying mechanical ventilation performance. In chapter 2 a robust procedure for calculating a set-theory Lagrangian method for calculation Markov matrices was established for steady flow fields. In chapter 3, a multistep and Eulerian method for creating Markov matrices with large Courant numbers are developed. These methods are also extended to transient source terms for large Courant numbers and time varying flow fields. The calculation of the contaminant fields is faster using the Markov method than using transport equations by 1 or 2 orders of magnitude, but may require some significant overhead for the initial calculation process. In chapter 4, Markov matrices are used to place sensors in optimal locations for detecting contaminants quickly. Spatial constraints to physical locations of sensors and sensing accuracy constraints are introduced into this formulation. In chapter 5, the connection between transport equations and Markov matrices are explored through absorbing Markov matrices. Transport equations and Markov methods are used to predict age of air, air residual life time, air residence time, and ventilation efficiency.

6.2 Future Work

The methods introduced in this thesis lays the foundation for many other applications of the discrete form of the PF-operator. In this work sensors have been placed based on the
regions where contaminants naturally collect, but sensors only provide a pointwise discrete realizations of the contaminant field. The next step in this area is to construct sensor estimators to provide volumetric approximation of the contaminant field in the sensing region upstream of the sensor. In the process of constructing a sensor estimator, contaminant source locations can be easily determined. Thus, providing avenues of neutralizing the contaminant source in the building. Another path of is extending these methods to problems with variable or uncertain boundary conditions (temperature, air flow rates, occupant locations, etc.). In these types of problems, a probabilistic Markov matrix could be calculated to provide averages, individual realizations, and higher order statistics. Finally, the connection of these tools with dynamic energy and whole building simulators. In these cases, reduced order models, surrogate models, may need to be used to create coupled simulations for long-term performance, HVAC demand control of contaminant concentrations, or predictive and analysis.