

Reduced Order Modeling For Large-Scale Linear Systems

by

Linyan Xiang

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Mechanical Sciences and Engineering)  
in the University of Michigan-Dearborn  
2022

Doctoral Committee:

Associate Professor Cheol W. Lee, Co-Chair  
Professor Oleg Zikanov, Co-Chair  
Assistant Professor Hugo Casquero  
Assistant Professor Zhen Hu  
Professor Dewey Dohoy Jung

Linyan Xiang

linyanx@umich.edu

ORCID iD: 0000-0002-6243-909X

© Linyan Xiang 2022

## ACKNOWLEDGMENTS

I would like to express my greatest appreciation to my supervisors, Dr. Oleg Zikanov and Dr. Cheol W. Lee. All the works are built upon the endless help, strong financial support, and careful guidance of the two mentors along with doctoral career.

I acknowledge, gratefully, Professor Oleg Zikanov for his constant encouragement, precious advice, and patience which made this work possible. Excellent guidance and helpful comments are greatly appreciated. I have been lucky to have a professional and knowledgeable supervisor who is passionate about research and life. I thankfully take the inspiring memo, and may it be with me in the future, as the Hitchhiker's Guide to the Galaxy says, don't panic.

I also acknowledge many thanks to Professor Cheol W. Lee for inspiring me academically and professionally. I hope that I can go to work and study in the future with a great profession, diligent work attitude, and the pursuit of perfection.

I would like to thank my peers and friends for their inspiration. Looking forward to the opportunity to continue working with you in the future. Special thanks to Dr. Xuan Zhang for encouragement and support; thanks to Ruslan and Ali for their mutual motivation during the epidemic; and Dr. Zhuo Wang for their excellence the inspiration from working together. Many special thanks to my friends and colleagues for their support during the Ph.D. journey.

Thanks to the excellent managers who have provided me with opportunities in the industry, their support has enabled me to get in touch with the impressive work experience in the industry. Thanks to Dr. Shu for allowing me to work at Eberspächer in 2018; thanks to Dr. Hsu of GM for the opportunity to dedicate contributions to the R&D process of the industry; thanks to Kerry and Qi for providing the opportunity to have wonderful summer at Mathworks.

At the same time, thanks to the computing resources provided by Great Lakes HPC cluster and technical consulting from Bennet Fauber. This research was supported in part through compu-

tational resources and services provided by Advanced Research Computing at the University of Michigan, Ann Arbor.

Finally, I would like to thank my parents for their support and trust in me, allowing me to follow my path on my own. Without their support and encouragement, I would not be able to get to where I am today.

# TABLE OF CONTENTS

ACKNOWLEDGMENTS . . . . .	ii
LIST OF TABLES . . . . .	vii
LIST OF FIGURES . . . . .	ix
LIST OF ALGORITHMS . . . . .	xiii
ABSTRACT . . . . .	xiv
CHAPTER	
<b>1 Introduction . . . . .</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Statement of the Problems . . . . .	1
1.3 Proposed Approaches . . . . .	4
1.4 Literature Survey . . . . .	4
1.5 Objectives . . . . .	11
<b>2 Theoretical and Numerical Models . . . . .</b>	<b>12</b>
2.1 High-Fidelity Large-Scale Linear Systems . . . . .	12
2.1.1 Conjugate Heat Transfer in a Battery Pack of an Electric Vehicle . . . . .	12
2.1.2 Airborne Transmission of Respiratory Infections . . . . .	15
2.1.3 Consistent Steady-State Representation of Large-Scale Linear Systems . . . . .	18
2.1.4 Boundary Condition as an Input . . . . .	19
2.2 Reduced-Order Modeling Methodologies . . . . .	21

2.2.1	Krylov-Subspace Method . . . . .	21
2.2.2	Parametric Reduced-Order Modeling . . . . .	23
<b>3</b>	<b>Algorithms . . . . .</b>	<b>26</b>
3.1	Indirect Arnoldi Algorithm for Single-Input Systems . . . . .	26
3.2	Indirect Arnoldi Algorithm for Multiple-Input Systems . . . . .	28
3.3	Frequency-Shift Arnoldi Algorithm . . . . .	30
3.4	Parametric Reduced Order Modeling Algorithm . . . . .	32
3.5	Change of Variable Method . . . . .	33
<b>4</b>	<b>Reduced-Order Modeling for Conjugate Heat Transfer in a Battery Pack . . . . .</b>	<b>35</b>
4.1	Battery Pack Model . . . . .	36
4.2	Error of ROM Approximation . . . . .	38
4.3	Single-Input Cases . . . . .	39
4.4	Multiple-Input Cases . . . . .	46
4.5	Boundary Condition as Input . . . . .	53
4.6	Parametric Reduced-Order Modeling . . . . .	56
4.7	Systems With Time-Varying Parameters . . . . .	58
4.8	Conclusion . . . . .	61
<b>5</b>	<b>Reduced-Order Modeling for Airborne Transmission of Respiratory Infections . . . . .</b>	<b>64</b>
5.1	Comparison Between Eulerian Approach and Lagrangian Approach Applied to Aerosol Transport . . . . .	64
5.2	ROM Generation and Optimization for Ventilated Indoor Environments . . . . .	75
5.2.1	Convergence Issues: Regarding the Residual of Transport Equation So- lution . . . . .	75
5.2.2	ROM Generation Using Single-Input Algorithm . . . . .	76
5.2.3	ROM Generation Using the Frequency-Shift Algorithm . . . . .	84

5.3 ROM Application . . . . .	93
5.3.1 Exhalation Cycle . . . . .	94
5.3.2 ROM Validation Under the Designed Coughing Cycle . . . . .	95
5.3.3 Time-Varying Process Predicted by ROM . . . . .	102
5.3.4 Risk Assessment Using ROM Data . . . . .	103
5.4 Conclusion . . . . .	106
<b>6 Conclusion and Future Works . . . . .</b>	<b>109</b>
<b>BIBLIOGRAPHY . . . . .</b>	<b>111</b>

## LIST OF TABLES

### TABLE

4.1	Model specifications. . . . .	37
4.2	The norms (in °C) of the error at various time instants for the test case 1 (see Figures 4.2, 4.4 and 4.5). . . . .	42
4.3	The norms (in °C) of the error at $r = 15$ for the test case 2. . . . .	44
4.4	The computation cost of the ROM creation and transient simulations in the single input case compared with the cost of the FOM simulations. . . . .	45
4.5	The norms of the error vector (in °C) for the test case 3 (see Figures 4.8–4.10). . . . .	48
4.6	The computation cost of the ROM creation and transient simulations compared with the cost of the FOM simulation of transient process in the test case 3 lasting 1000 seconds. . . . .	49
4.7	The norms (in °C) of the error vector for the test case 4 (see Figures 4.11, 4.12 and 4.13). . . . .	53
4.8	The norms of the error vector (in °C) for the boundary condition as the input (see Figures 4.16). . . . .	56
4.9	The norms of the error vector (in °C) for parametric ROM at mass flow rate = $0.02\text{kg/s}$ (see Figures 4.17 and 4.18). . . . .	58
4.10	The norms of the error vector (in °C) for parametric ROM at time-varying mass flow rates (see Figures 4.19). . . . .	60
5.1	The quantitative study between Eulerian/Lagrangian methods at $t=50\text{s}$ . . . . .	74
5.2	The quantitative study between Eulerian/Lagrangian methods at $t=100\text{s}$ . . . . .	75
5.3	The quantitative evaluation for ventilation ROM using <b>Algorithm 1</b> . . . . .	82
5.4	The 11 Sigma points for frequency-shift Krylov method . . . . .	85

5.5	The quantitative evaluation for ventilation ROM using <b>Algorithm 3</b> . . . . .	91
5.6	The norms at $t=10, 50$ and $220s$ . . . . .	102
5.7	Volume integration of passive scalar concentration over boxes in front of students at $t=2000s$ . . . . .	106

## LIST OF FIGURES

### FIGURE

1.1	Dissertation outline . . . . .	11
2.1	Schematic representation of various modes of respiratory infection transmission. . . . .	17
2.2	Breathing cone. Spatial distribution of the source intensity $B(\mathbf{x})$ is shown. . . . .	18
2.3	Boundary cells illustration . . . . .	21
4.1	Schematic of the battery pack with dual coolant channels. . . . .	36
4.2	Heat input signal for the test case 1 – single input in the form of a stepwise heating rate. . . . .	40
4.3	Location of the point within the battery pack, at which temperature is probed. (a) Front view; (b) Side view. . . . .	40
4.4	Comparison between ROM and FOM solutions for the test case 1 (see Figure 4.2). Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of $r$ are shown. . . . .	41
4.5	Temperature observed in the middle cross-section of battery pack for the test case 1. The heating input signal is shown in Figure 4.2. The results of ROM with $r = 15$ and FOM at different instants are presented. (a) FOM $t=3s$ ; (b) ROM $t=3s$ . (c) FOM $t=402s$ ; (d) ROM $t=402s$ . (e) FOM $t=800s$ ; (f) ROM $t=800s$ . . . . .	43
4.6	Results for the test case 2 with a realistic transient single input. The heat input signal (yellow line, right-hand side axis) and the temperature values at the probed point obtained in the FOM and the ROM with $r = 15$ are shown. . . . .	44
4.7	The Bode diagrams of the FOM and the ROM at different values of $r$ for a single-input system. (a) Magnitude plot; (b) Phase plot. . . . .	45
4.8	Heating rates for the test case 3 – multiple inputs with two stepwise heating rates. . . . .	46

4.9	Comparison between the ROM results with $r = 16$ and the FOM results for the test case 3 (see Figure 4.8 for input signals). Temperature distributions in the middle cross-section of battery pack are shown. (a) FOM $t=3s$ ; (b) ROM $t=3s$ . (c) FOM $t=510s$ ; (d) ROM $t=510s$ . (e) FOM $t=1000s$ ; (f) ROM $t=1000s$ . . . . .	47
4.10	Comparison of ROM and FOM solutions for the test case 3 (see Figures 4.8 and 4.9.) Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of $r$ are shown. . . . .	48
4.11	Heating rate input of the battery cells for the test case 4 – multiple inputs. The numbers indicate the heating rates applied to individual cells. . . . .	49
4.12	Comparison between the ROM results at $r = 50$ and the FOM results for the test case 4 (see Figure 4.11 for input signals). Temperature distributions in the middle cross-section of the battery pack are shown. (a) FOM $t=400s$ ; (b) ROM $t=400s$ . (c) FOM $t=505s$ ; (d) ROM $t=505s$ . (e) FOM $t=600s$ ; (f) ROM $t=600s$ . . . . .	51
4.13	Comparison between the ROM and FOM results for the test case 4 (see Figures 4.11 and 4.12). Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of $r$ are shown. . . . .	52
4.14	Extruded layer method for BC as input. The extruded layer is marked in pink. . . . .	54
4.15	Inlet temperature signal for boundary condition as input validation. . . . .	54
4.16	Comparison of ROM and FOM solutions for the boundary condition as input. Transient temperature signals at the probed point obtained in the FOM and the ROM of order 20 are shown. . . . .	55
4.17	Comparison between the parametric ROM results at $MFR = 0.02kg/s$ and the FOM results. Temperature distributions in the middle cross-section of battery pack are shown. (a) FOM $t=5s$ ; (b) parametric ROM $t=5s$ . (c) FOM $t=400s$ ; (d) parametric ROM $t=400s$ . . . . .	57
4.18	Comparison of parametric ROM and FOM solutions at mass flow rate = $0.02kg/s$ . Transient temperature signals at the probed point obtained in the FOM and the ROM of order 15 are shown. . . . .	58
4.19	Comparison of parametric ROM and FOM solutions under transient mass flow rates. Temperature signals at the probed point obtained in the FOM and the parametric ROM are shown. . . . .	60
4.20	Reverse flow developing in the cooling channel with time-dependent mass flow rate . . . . .	61
5.1	Model geometry and iso-surface of infectivity field $C=10$ at $t=100s$ . . . . .	65

5.2	Illustration of (a) the classroom model and (b) the computational mesh used in the CFD simulations. The imagine from [1] is represented with permission of AIP Publishing. . . . .	66
5.3	(a) Turbulent kinetic energy, (b) velocity magnitude distribution, and (c) velocity vectors across a slice going through students 2, 5, and 8. The imagine from [1] is represented with permission of AIP Publishing. . . . .	67
5.4	Distribution of 1 $\mu\text{m}$ aerosol particles in the classroom at different points in time for the (a) student 5 source. The imagine from [1] is represented with permission of AIP Publishing. . . . .	68
5.5	Transient Scalar Intensity signal of the aerosol cloud. . . . .	69
5.6	Initial scalar intensity source cloud (Source is marked in red box with $0.38 \times 0.4 \text{ m}^2$ in area and 0.3m in height). . . . .	70
5.7	The spatial distribution of infectivity field C at t=20s. . . . .	71
5.8	Infectivity field C at t=50s (on the left) and 100s (on the right) . . . . .	72
5.9	Line plot of total percentage of Infectivity Field C/Particle along central direction of classroom. The plot at t=50s is shown in (a); the plot at t=100s is shown in (b). . . . .	72
5.10	Schematic diagram of the partition of the classroom . . . . .	73
5.11	Transient signal at the probed point in front of student 5 . . . . .	78
5.12	Distributions of C along the line at $y=2.25\text{m}$ , $z=4.5\text{m}$ obtained by the FOM and by SIMO ROMs with various r. . . . .	79
5.13	Contour plot at t=5s in the cross-section $z=4.5\text{m}$ under the signal of 5.5. ROM of (a) $r=5$ ; (b) $r=15$ ; (c) $r=30$ ; (d) $r=60$ ; (e) $r=85$ ; (f) FOM . . . . .	80
5.14	Contour plot at t=50s in the cross-section $z=4.5\text{m}$ under the signal of 5.5. ROM of (a) $r=5$ ; (b) $r=15$ ; (c) $r=30$ ; (d) $r=60$ ; (e) $r=85$ ; (f) FOM . . . . .	81
5.15	Infectivity field C at t=100s. (a) Optimal ROM, with $r=65$ , using SIMO algorithm, and (b) FOM. . . . .	84
5.16	Results of frequency-shift ROMs for the source signal in Fig 5.5 . . . . .	86
5.17	Distributions of C along the line at $y=2.25\text{m}$ , $z=4.5\text{m}$ obtained by the FOM and by frequency-shift ROMs with various r. . . . .	87
5.18	Contour plot at t=5s in the cross-section $z=4.5\text{m}$ under scalar intensity signal 5.5. Frequency-shift ROMs of (a) $r=11$ ; (b) $r=22$ ; (c) $r=33$ ; (d) $r=66$ ; (e) $r=99$ ; (f) FOM . . . . .	88

5.19	Contour plot at $t=10s$ in the cross-section $z=4.5m$ under scalar intensity signal 5.5. Frequency-shift ROMs of (a) $r=11$ ; (b) $r=22$ ; (c) $r=33$ ; (d) $r=66$ ; (e) $r=99$ ; (f) FOM . . .	89
5.20	Contour plot of error vector at $t=10s$ . ROM of (a) $r=11$ ; (b) $r=99$ . . . . .	89
5.21	Contour plot at $t=50s$ in the cross-section $z=4.5m$ under scalar intensity signal 5.5. Frequency-shift ROMs of (a) $r=11$ ; (b) $r=22$ ; (c) $r=33$ ; (d) $r=66$ ; (e) $r=99$ ; (f) FOM . . .	90
5.22	Breathing and coughing cycles used in simulations. Source modulations $u(t)$ corresponding to breathing cycles (a) and coughing cycles (b) are shown. . . . .	95
5.23	Source modulation signal $u(t)$ used in the numerical experiment. . . . .	96
5.24	Transient scalar signals, under the input of 5.23, at the probed points in front of students whose frontal space are identified as 'high-risk' area. . . . .	97
5.25	Cross section at $t=10, 50, 220s$ of ROM at (a), (c), (e) and FOM at (b), (d), (f). . . . .	98
5.26	Probe line signal of (a) $t=10s$ , (b) $50s$ and (c) $220s$ at $y=2.5$ m (elevation) . . . . .	99
5.27	Probe line signal of $t=10s$ (a), $50s$ (b) and $220s$ (c) at $y=1.3$ m (elevation) . . . . .	101
5.28	Time-varying source modulation signal $u(t)$ in simulating a lecture. . . . .	103
5.29	Concentrations of infectivity field $C$ at probe points in front of students under the input signal of 5.28. Left axis data for student 5; Right axis-data for the other students. . . .	104
5.30	At $t=2000s$ , (a) Iso-surface of passive scalar with iso-value of 0.8; (b) Cross-section of $C$ . . . . .	105
5.31	Integration area in front of students. . . . .	105
5.32	The data for student 5 (the source) are excluded. . . . .	106

## LIST OF ALGORITHMS

1	Indirect Arnoldi algorithm for single-input systems . . . . .	27
2	Indirect Arnoldi algorithm for multiple-input systems . . . . .	30
3	Frequency-shift Arnoldi algorithm for single-input systems . . . . .	32
4	Parametric Model Order Reduction algorithm . . . . .	33

## ABSTRACT

A large variety of physical phenomena can be described by large-scale systems of linear ordinary differential equations (ODEs) obtained by one of the discretization methods, in particular one of the methods of Computational Fluid Dynamics (CFD). The solution of such ODE systems is relatively straightforward with well-developed methods, which makes the large-scale linear systems one of the powerful ways of analyzing physical phenomena. Their practical applicability is, however, severely limited by the computational expense. Days or even weeks may be needed to simulate an unsteady behavior of a system with typical  $10^6$  or more degrees of freedom. This limits applications in many important areas, from the demand for extensive solution results for fast-paced optimization design to the need for industrial online predictive control. Therefore, efficient yet accurate models that approximate large-scale linear systems are critically needed. We focus on two major application scenarios: thermal management system in battery packs of electrical/hybrid electric vehicles and the prediction of airborne transmission of respiratory infections, e.g., SARS-COV-2, in indoor environments. The reduced-order modeling (ROM) Krylov-subspace method is developed to reduce the computational effort of CFD. It is based on the projection of the original model onto a Krylov subspace by the Arnoldi-type algorithms. Versions of the method for both single-input and multiple-input systems are presented. The algorithms do not require access the original system matrix, which is usually inaccessible from commercial CFD software. The comparison between the results using the ROM and the original CFD models shows a reduction by a factor of  $10^3$  in computational time without significant loss in the accuracy of the results.

# CHAPTER 1

## Introduction

In this chapter, the motivation is introduced firstly, followed by the definition of two large-scale linear systems. The reduced-order modeling is proposed as a tool of generation of a small yet accurate models of original large-scale linear systems. Finally, the outline of the dissertation is presented.

### 1.1 Motivation

The project is motivated by the demand for computing low-order approximated models of large-scale linear systems. Firstly, a large variety of physical phenomena are modeled with linear, time-invariant (LTI) systems. Such models are generally produced as a result of the discretization of the governing equations, e.g., by the finite volume and finite difference methods, into a large number of ordinary differential equations (ODEs). The solution of an ODE system can be achieved in various ways, e.g., by the iterative methods. Thus, large-scale linear systems are often used in analyzing physical phenomena from practical perspectives.

### 1.2 Statement of the Problems

The examples of such applications include the modeling of conjugate heat transfer in a battery pack of an electric vehicle. Despite the advantage of high specific energy and energy density with relatively low cost compared to other types, the Li-ion battery technology is not free from prob-

lems. One of the major concerns is the thermal behavior of a battery pack in its on-board operation [2]. It is known that operating Li-ion batteries outside the normal temperature range negatively affects their efficiency, safety, reliability, and lifespan [2, 3]. A battery thermal management system (BTMS) capable of effective control of the heat transfer processes, so as the battery temperature remains within the desired range is, thus, critically important for practical applications [2].

In order to play its role, a BTMS must be able to accurately describe the temperature field inside a battery pack and predict the evolution of this field in response to variations of load, changes in operation of cooling condition, and other factors. The conjugate heat transfer between batteries and cooling liquid must be reproduced. High-fidelity computational fluid dynamic (CFD) simulations using such numerical techniques as the finite volume method or finite element method are often utilized. The numerical techniques are applied to discretize the system's spatial and time domain into numerous control volumes to generate a detailed reproduction of the heat transfer process [4]. While important and often indispensable for high-fidelity analysis, the CFD approach is often computationally unaffordable for rapid analysis. The reason is the large scale of the models ( $\sim 10^6$  to  $\sim 10^9$  degrees of freedom), which requires tens to thousands of core-hours to complete a single simulation. This severely limits the use of CFD for onboard control, extensive simulation studies, and simulations coupled with electrochemical battery cell models [5].

A second example of large-scale linear system is the modeling of airborne transmission of respiratory infections, e.g., SARS-COV-2, in indoor environments. During the current epidemics, society has shown a poor understanding of the mechanisms of transmission of respiratory infections. As the variants of COVID-19 virus continue to break out with stronger infectious risk even to people who have already been vaccinated [6, 7]. The understanding of these mechanisms and the ability to predict the transmission is essential for fighting the pandemics in many perspectives, e.g., for making consistent health policy decisions, designing prevention facilities [8, 9]. A comprehensive review of various aspects of the airborne transmission methods of numerical analysis, and open questions can be found in [10].

The study of transmission mechanism usually requires extensive laboratory or field experiments

on infection transmission using human subjects and actual virus [11]. However, this is either impossible or prohibitively expensive due to safety concerns. In order to successfully predict the airborne transmission of respiratory infections, a feasible alternative is to conduct the experiments in the form of numerical simulations. The airborne transmission process after virus-laden droplets are generated by a respiratory activity of an infected person can be modeled in that way. The respiratory activity forms a turbulent cloud of the virus-laden droplets that have various sizes and velocities. The droplets diameters range from 0.1 to 1000  $\mu m$ . Droplets of large size descend to the ground or other solid surface quickly due to their high settling speed. The Lagrangian approach is usually applied to determine particle dispersion pattern and track the pathway of each individual particle [12].

In our work, we only consider the behavior of small droplets ( $< 10\mu m$ ) which is referred to as the airborne transmission [10, 13]. Their own inertia is negligible (since the Stokes number is below  $10^{-4}$ ) and settling speed is small (less than 1  $mm/s$ ), and they travel suspended in air currents. The propagation of small droplets in a turbulent air flow is studied in the form of particle concentration distributions in indoor environment. Since we focus on the particle concentration distributions, the Eulerian approach, which considers the particle cloud as a continuum where the concentration of some particle-related characteristics, e.g., alive virions, is applied. The concentration is expressed by a scalar field evolution of which is determined by a transport equation [10, 14]. Details will be discussed later in Section 2.1.2.

It is important to conduct numerical experiments in order to [15, 16]:

- Predict the distribution of the airborne virus-laden particles from various configurations, e.g., population distribution or density in the room, different droplet sizes, different mass flow rates of ventilation;
- Understand the decay of virus viability in turbulence;
- Assess the risk of airborne transmission in an indoor space;
- Draft the public regulation to mitigate the indoor airborne transmission;

- Understand the mechanism of infection.

Therefore, extensive numerical experiments are needed to study airborne transmission. However, the computation cost of each such experiment makes it difficult to accumulate a sufficiently large number of data by performing parametric studies under extensive parameters.

### **1.3 Proposed Approaches**

Solution of large-scale linear system modeling is common and typically required for a detailed and accurate description of the physical phenomena. Such solutions are fairly expensive, which limits the use of numerical modeling to be inefficient in practical applications. The systems in all examples mentioned above consist of ODEs with more than  $10^6$  degrees of freedom. A simulation of a transient process takes days of calculations even with parallel computations. Although such high-fidelity models tend to accurately describe the behavior of the physical system, the computational expense makes it highly impractical.

One of the promising ways to save the computation cost while maintaining high fidelity in simulations of large-scale linear system is the reduced-order modeling (ROM). The ROM approximates a large-scale system by a small yet accurate system. Compared with large-scale linear systems, the ROM has the critical advantages of efficiently predicting the system behavior while dramatically reducing the computation cost. The work presented in this dissertation focus on development, verification, and use of such models.

### **1.4 Literature Survey**

Several attempts to develop an ROM for battery thermal management have been made before. A control-oriented model based on the singular perturbation method was proposed in [17]. The method predicts the battery cell's internal temperature profile by applying the Laplace transform to the one-dimensional boundary-value problem and further reducing it into a low-order linear model

in the frequency domain. However, the effect of heat removal from the cooling components is not considered in the model, making its applications very limited.

In [18] and [19], ROMs are developed by parameter identification using experimental data under different geometries, inlet flow velocities, and temperature. The models relatively accurately predict temperature on the cell's surface, in its core, and average temperature of the cell, which are applied for control purposes. However, the extensive experiments under various system parameters required to derive the models make their use ineffective. Besides, the models become impractical when the complete temperature field is needed for the design and optimization of the BTMS.

An efficient reduced-order modeling for large-scale BTMS based on singular value decomposition (SVD) was presented in [20, 21]. The model was obtained by applying the SVD to snapshots of high-fidelity CFD solutions. The method was shown to be effective. Its major drawback is that the approximation accuracy strongly depends on the selection of snapshots, which makes it impractical.

Comprehensive reviews of the ROM methods in general can be found in [22, 23, 24]. The proper orthogonal decomposition (POD) method uses selected instantaneous states of the system during its evolution as basis vectors. The method is applicable to linear and nonlinear systems, but it is computationally expensive for large-scale systems [25]. The balanced truncation method determines a balancing transformation by computing the controllability and observability Gramians from the Lyapunov equations. The method can preserve the asymptotic stability of a stable full-order modeling (FOM) while allowing the optimization of ROM by the error bound. The computations needed to solve the high-dimension Lyapunov equations are, however, extensive, which makes the method less suitable for large-scale systems [24, 25].

The projection-based Krylov-subspace method based on moment matching is considered to be a good candidate for ROM development. The method computes the basis vectors that span the Krylov subspace. The basis vectors are used to reduce the FOM into a low-order state-space system such that the Taylor series coefficients of the transfer function (also called moments) are matched between the ROM and FOM. The method is only applicable to linear systems, but known

to deliver good performance (see [26] for a review). Examples of its use are the developments of reduced-order models for micro electro-mechanical system (MEMS) actuators and microfluidic chip thermal systems [27, 28], the near-range atmospheric dispersion [29], structural-acoustic phenomena [30], and the aeroelastic analysis of turbomachines [31].

There are many algorithms available for construction of the Krylov subspace. One is the Padè approximation via Lanczos (also known as ‘PVL’) algorithm [32], which generates two sequences of basis vectors spanning the input and output Krylov subspaces. The method is impractical because the generated ROMs occasionally become unstable even though the original FOM system is stable [33]. A better alternative is the Arnoldi algorithm, in which the modified Gram-Schmidt process is applied to construct the orthonormal basis of the Krylov subspace [34].

A common feature of the Krylov-subspace method algorithms is that they require access to the system matrix that contains the coefficients of the discretized governing equations of the FOM. This hinders application of the methods for practical CFD analysis, in particular for the thermal analysis in the BTMS, since most commercial CFD solvers, such as ANSYS Fluent [35] or STAR-CCM+ [36], do not provide a user the access to the system matrix. While an accurate ROM is often needed for the efficient thermal analysis in the BTMS and multiphysics modeling in the industrial product lifecycle management environment, few studies have attempted the Krylov-method ROM development using approaches that do not require access to system matrix. One such indirect method based on the Arnoldi algorithm was reported in [29] to predict the hazardous pollutant’s dispersion into the atmosphere. The method, however, was demonstrated for the single-input systems only. To our best knowledge, several Arnoldi-type algorithms developed for the multiple-input systems using Krylov subspace are not applicable to the situations when system matrices are not accessible [27, 37, 38].

We attempt to fill the gaps mentioned above by reviewing the previous works. With this purpose in mind, the preliminary result includes the developed algorithms for large-scale linear dynamical system describing the conjugate heat transfer. The major advantage of proposed approach is that it does not need access to the system matrix. The method is based on the projection of the FOM onto a

Krylov subspace by the Arnoldi algorithm. Versions of the algorithm for single-input and multiple-input systems are developed. The algorithms are applied to derive ROM models of conjugate heat transfer in a battery pack. Single-input (when the system is controlled by a single heat source term) and multiple-input (when components of a pack are controlled individually) cases are considered. The ROM is used to predict the unsteady transient conjugate heat transfer in an entire battery pack. The comparison between the predictions and the results of high-fidelity CFD modeling shows a dramatic reduction in computational time without significant loss in the accuracy. The developed techniques are not restricted to BTMS and can be employed, in general, for reduced-order modeling of time-invariant linear dynamical systems including those describing conduction, convection or conjugate heat transfer or transport of admixtures.

Concerning the second system considered in the dissertation, there have been extensive discussions of the airborne transmission of respiratory infections, since it is identified as one of the major risks of infection [39]. Studies indicate that the airborne transmission at outdoor locations are posing a light risk of infection [40]. Compared with that, the airborne transmission is identified as a substantially risky scenario in indoor environments [41, 42].

In [43, 44, 45], it is found that the parameters of the droplets, such as their number, size, initial velocity, and viral load, vary greatly depending on the type of activity (breathing, sneezing, coughing, singing, etc.), and physical characteristics and the stage of infection of the source. As an example, the droplet size can be anywhere between 0.1 and 1000  $\mu m$  [10, 12, 44]. Droplets larger than a few tens of  $\mu m$  have a significant settling speed, so they soon descend to the ground or another solid surface<sup>1</sup>. There are also some droplets with a diameter between 10  $\mu m$  and 100  $\mu m$  that will attach to the top of a possibly moving surface in a short time  $\sim 10$  min. There, virions may remain infectious for several hours to several days depending on the type of the surface, humidity, UV irradiation, and other factors. The last part of the droplets with a diameter of less

---

<sup>1</sup>The typical distance traveled by such particles before attaching to a surface was determined in the 1930s and 1940s, most notably in [46]. It appears that those findings form the basis of the modern recommendation of the safe person-to-person distance of 6 ft. The problem with the findings and, thus, the recommendation is that the threshold was identified in [47] as a distance traveled by droplets of size 100  $\mu m$  issued by an average coughing source. It is now understood that smaller particles ejected with higher velocity (e.g., by sneezing) may travel 20 ft or even further [48].

than  $10\ \mu\text{m}$  will travel in the indoor air with the airflow in the form of aerosol for a long period of time and broad range of space. The proposed work will focus exclusively on this mechanism of transmission, namely by droplets of  $10\ \mu\text{m}$  and smaller. Their own inertia is negligible (since the Stokes number<sup>2</sup> is below  $10^{-4}$  and settling speed is small (less than 1 mm/s), and they travel suspended in air currents. Water quickly evaporates from their surface, so they become particles consisting of virions and semi-solid residue. The viability and, plausibly, infectious potential of the virions have been measured to decrease by half in about 1.2 h [49].

Considering that each particle contains a large number of virions, and that a small number of (theoretically, one) virions are sufficient for infection, each particle retains infectious potential for many hours. This scenario suggests the possibility of airborne transmission [50, 51]. A person is infected after inhaling or contacting in some other way a small airborne particle generated, possibly, long time ago, by a source located far away. The possibility appears especially realistic in indoor environments, where strong recirculating flows of air are generated by ventilation systems and heat sources (humans, computers, etc).

In a typical approach, (see, e.g., [48]), one estimates the number of potentially infectious doses produced by a source over a time period and then evaluates the probability of transmission based on the assumption that the doses are uniformly distributed in the room. The assumption and the approach itself are prone to criticism from the physical viewpoint. Turbulent transport from a source of time-dependent intensity inevitably results in a distribution of particles, which is both strongly non-uniform (high concentration in some areas and zero in others) and unsteady. These models currently used in epidemiology ignore the uncertainties or treat them in an oversimplified way. Computational Fluid Dynamics (CFD) approach offers a more accurate alternatives. Instead, one of the viable ways to study the airborne transmission is achieved by a RANS (Reynolds-Averaged Navier-Stokes) model that computes steady-state averaged velocity and turbulent diffusivity of air flow fields [52] as a ‘base’. Then, either Lagrangian or Eulerian can be used to reproduce the propagation of aerosol particles. Depending on the focus of research, these two methods are applied to

---

<sup>2</sup>The Stokes number is defined as the ratio of the characteristic time of a particle to a characteristic time of the flow or of an obstacle. It is a dimensionless number characterising the behavior of particles suspended in a fluid flow.

substantially different scenarios according to characteristics of the methods.

Lagrangian method is applied in [53] to analyze transmission in a bus with windows and doors closed, at half-seated occupancy under COVID-19 restrictions, with the main HVAC at maximum flow rate, the driver HVAC on low, and the defroster on medium. The work in [15] presents CFD study of the influence of an alternate ventilation configuration on the possible flow path of infectious aerosols using Lagrange method. The investigation in [44] computed the transport of the droplets exhaled by the index patient at designed position. The bulk airflow pattern was calculated using CFD RANS while the airborne droplet was modeled using a Lagrangian method. It can simulate the trace of particles under different diameters, and even the process of thermodynamic changes of the droplet during the propagation process, and therefore can predict the dispersion pattern of the particles. However, a very dense mesh must be used otherwise the calculated path will vary greatly due to poor quality of the grid.

A numerical model using RANS and the Eulerian approach is presented in [54] to study the transient behavior of cough particles transport in a chamber. The work evaluated the risk of infection under different ventilation scenarios for which the conclusion was confirmed by experimental data. The study [10] investigated the aerosol transmission under different droplet sizes considering the particle lingering over time in a generic public place. It assessed the risk of infection as ‘exposure time’ over indoor environment. Through solving the transport equation with Eulerian method, the research in [55] estimated the transport of indoor contaminant. Better agreement with respect to measured chamber concentrations could be found both qualitatively and quantitatively over those using the uniform inlet velocity.

The Eulerian approach can predict the spatial distribution of a concentration field. The distribution can then be used in a statistical analysis. The result can be validated by comparing both qualitatively and quantitatively with physical experiments. Through the spatial distribution of aerosol, we can effectively predict the high-risk infection area. The Eulerian method is more efficient than the Lagrangian method in prediction of particle distribution. The limitation of the Eulerian method is its inability to reproduce the inertial effects of the particle evolution. It is,

therefore, recommended to apply the Lagrangian method for large particles, for which the inertial effects are significant. The Eulerian method is best suited for studying propagation of small particles characterized by small Stokes number.

Overall, due to a substantial computation effort, the Lagrangian method cannot efficiently handle the amount of calculations in simulating series of activities. Because the number of particles produced by speaking is significant as it is normally done continuously over a long period of time. The simulation constantly exhales a huge number of particles to calculate its trajectory. This task will consume a huge amount of computing resources for Lagrangian approach.

Since the focus of our research is on predicting the particle concentration distributions and since the attention is limited to aerosol particles with  $St \ll 1$ , we apply the Eulerian approach. The method model the particles as a continuum where the concentration of particle-related characteristics, e.g., alive virions, is represented by a scalar field in a transport equation [10]. Then, a transport equation for the concentration of the droplets can be solved with precomputed velocity and diffusivity fields. The intensity of the source can be derived from the experimental data on expiratory droplet formation, and virus loading [44, 47, 48].

CFD modeling the airborne transmission is computationally expensive, e.g., [10]. It cannot, therefore, be used in large-scale statistical studies taking into account variability of the source of infection. It cannot also be used for real-time control of the strategies of prevention of infection. An effective alternative explored in this work is to develop an ROM. It in general requires off-line computation to approximate the original full-order model into a model that has much smaller size. Once ROM is generated, however, it can be used in a large number of fast and accurate predictions. To the best of our knowledge, the method of reduced-order modeling has never been applied to analysis and prediction of indoor airborne transmission of respiratory infections. On this basis, algorithms that use the Krylov subspace method are developed to build an efficient, reliable ROM for the study of the field.

## 1.5 Objectives

As a summary, we will develop and apply reduced-order models for two configurations: the heat transfer in the battery pack of an electric vehicle and the airborne transmission of respiratory infections. The work is outlined in Figure 1.1. Each study will start with the modeling of a high-fidelity large-scale system that predicts the physical behavior accurately. This part is categorized as the ‘Modeling based on physics’. The second part identified in Figure 1.1 as ‘Reduced-order modeling’ will include development, analysis and validation of ROMs. Efficient algorithms for single-input and multiple-input scenarios will be developed and implemented to construct the ROM that can accurately approximate the high-fidelity large-scale system. The accuracy of the ROMs is confirmed in verification studies, in which predictions of the ROM and of the full-order models are compared for realistic scenarios.

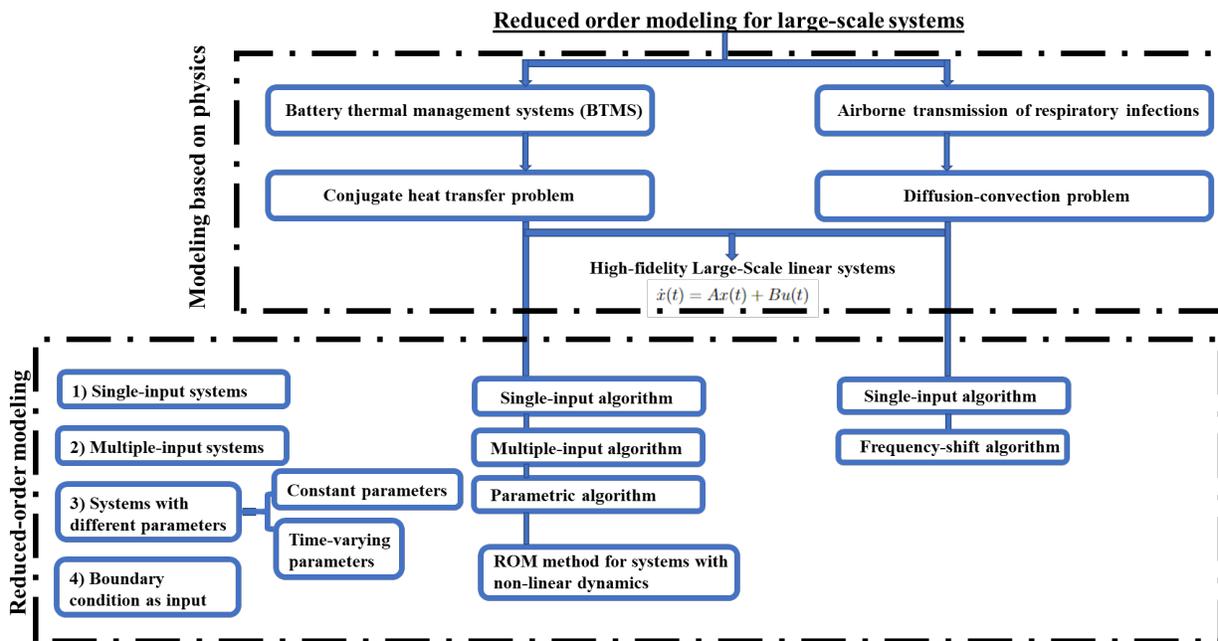


Figure 1.1: Dissertation outline

## CHAPTER 2

# Theoretical and Numerical Models

As introduced in Chapter I, two examples of large-scale linear systems are considered. Although different governing equations are used, the numerical solution leads to similar linear systems which can be written in the form of state-space representation as presented in 2.1. The reduced-order modeling methodologies for large-scale linear systems are presented in 2.2.

## 2.1 High-Fidelity Large-Scale Linear Systems

### 2.1.1 Conjugate Heat Transfer in a Battery Pack of an Electric Vehicle

Firstly, we consider the conjugate heat transfer in the battery pack of an electric vehicle. The proposed ROM method, while generally applicable to a wide array of linear problems, will be applied and tested for the specific conjugate heat transfer system described in this section. The system is a simplified model of a typical automotive battery pack. Details of battery chemistry are ignored, and the effects of charging and discharging on heat transfer are simulated as time-dependent internal heat sources distributed over the battery interiors. At the same time, all the key features of a typical battery pack directly relevant to heat transfer and BTMS are accounted for by the model. This includes a realistic set of physical properties and a realistic geometry with multiple coupled subdomains: solid battery cells, casing, cooling plate, and cooling channels with water flowing through them (see section 4.1).

The remaining discussion in this section has two parts: *(i)* the list of further simplifying assump-

tions and corresponding governing equations of heat transfer and (ii) the state-space representation of the problem from a CFD discretization.

The battery system includes coupled solid battery cells and liquid cooling substance domains. The battery is cooled by heat transfer from solid battery cells to the liquid cooling substance. While the heat generation rate in cells is determined by Joule heating, phase change, mixing effects and electrochemical reactions [56], the electrochemical behavior of the battery cell and the effect of temperature on cell behavior are not considered. Instead, since we focus on examining the cooling performance of the pack design, the cumulative heat generation rate by battery cells is imposed as input.

The battery system has multiple coupled subdomains, including solid battery cells and liquid coolant zone. The heat transfer in the solid region is by conduction and, thus, described by the linear partial differential equation for temperature:

$$\frac{\partial T}{\partial t} = \frac{1}{\rho C_p} \nabla (\kappa \nabla T) + \frac{1}{\rho C_p} Q, \quad (2.1)$$

where  $T$  is the temperature,  $Q$  is the volumetric density of internal heat source, and  $\rho$ ,  $C_p$ ,  $\kappa$  are the density and specific heat capacity of the solid, which do not depend on time and temperature, but can vary in space since their values are different for different subdomains. The thermal conductivity  $\kappa$  is also a function of space. It can also be anisotropic, so  $\kappa$  is a tensor, rather than a scalar.

The heat transfer in the liquid subdomain is a non-linear process governed by coupled momentum, continuity, and energy equations. This problem can be simplified to a linear system of equations with constant coefficients for temperature by adopting the following commonly valid assumptions:

**Negligible effect of natural convection.** In many thermal management systems, the convection heat transfer in liquid subdomains is dominated by forced convection. The temperature-dependent buoyancy force in the momentum balance can be ignored.

**Constant flow properties.** We assume that the fluid is incompressible and Newtonian and that all the physical properties, such as density, thermal conductivity, specific heat, and viscosity are constant. This approximation is widely adopted in heat transfer analysis and should be considered as adequate so far as the temperature variations within the system are not too large (see [57] for a detailed discussion).

**Steady-state flow velocity field** The fluid flow is assumed to be fully developed under a certain constant mass flow rate at the inlet. Thus, the flow velocity field  $\mathbf{U}$  is steady-state. This is justified by our examples considered in section 4.1, where we show that the values of the Reynolds number are below or about the value  $Re \approx 2000$  of transition to turbulence. We note that the assumption can be extended to the case of turbulent flows with time-independent mean velocity, in which case a RANS turbulence model has to be applied (see, e.g. [52]).

The first two assumptions assure that the values of physical properties and velocity are temperature-independent. The third condition implies that the coefficients are also independent of time. The steady-state flow velocity field and, in the case of turbulence, eddy diffusivity can be pre-computed using a CFD solver. The solver computes solutions of the equations of mass and momentum conservation for incompressible fluid flows in the coolant channel. Then, the fields are used to calculate the coefficients in the temperature equation for both FOM and ROM generation. It must be noted that the computational time required for pre-computing the fluid flow is small in comparison with the time needed to compute the steady-state temperature field. The flow calculations are limited to a small portion of the system consisting of the coolant channel with simple geometry, which substantially reduces the computation effort. Overall, the assumptions decouple the Navier-Stokes equations for velocity and pressure from the energy equation for temperature in the fluid region. In each analysis of the system, a steady-state solution of the Navier-Stokes equations at given flow parameters, such as the flow rate, is calculated once to determine the velocity field. The energy equation describing various heat transfer scenarios at the same flow can then be considered as a linear equation with constant physical properties and constant three-dimensional flow velocity field  $\mathbf{U}$  (the eddy diffusivity term is omitted for simplicity):

$$\frac{\partial T}{\partial t} + \mathbf{U}\nabla\mathbf{T} = \frac{\kappa}{\rho\mathbf{C}_p}\nabla^2\mathbf{T} + \frac{1}{\rho\mathbf{C}_p}\mathbf{Q}. \quad (2.2)$$

Our next step is to represent (2.2) as a linear time-invariant (LTI) system. The partial differential equation (2.2) is discretized, in our case by the finite volume method, to produce a system of ordinary differential equations for the values of temperature at  $n$  grid points (centroids of finite volume cells). The system can be formally written as a state-space system with  $m$  input and  $n$  state variables as shown later in this chapter.

## 2.1.2 Airborne Transmission of Respiratory Infections

The second physical system to be analyzed is the propagation of virus-laden droplets in turbulent air flow. Airborne transmission of a respiratory viral infection, such as COVID-19, is considered. Macro droplets containing active virions are exhaled by an infected person [39]. The parameters of the droplets, such as their number, size, initial velocity, and viral load, vary greatly depending on the type of activity (breathing, sneezing, coughing, singing, etc). In line with the intended focus on airborne transmission, the model only considers droplets of small size, i.e., diameter below  $10\mu m$ . We analyze the transport of such droplets in a turbulent air generated by an indoor ventilation system.

The RANS (Reynolds-Averaged Navier-Stokes) model is used to study the flow dynamics of ventilation system. The three-dimensional steady-state flow equations are solved to find the spatial distributions of the mean velocity and turbulent diffusion coefficients. Then, the diffusion-convection equation based on the Eulerian description is solved to predict the spatial-temporal evolution of a scalar field representing the volume concentration of infectious virions in air. We assume that:

**Negligible effect of aerosol particles size on the air flow.** Their own inertia is negligible (since the Stokes number is below  $10^{-4}$ ) and settling speed is small (less than 1 mm/s), and they travel suspended in air currents. In this way, the air flow equations are decoupled from

the transport equation. Once the flow velocity field and turbulent diffusivity of air flow are found, it is frozen and used in calculating the transport of the aerosol particles

**Droplet size does not change during the airborne transmission process.** In the process of exhalation, a mucous droplet undergoes evaporation. The process reduces the diameter of the droplet and makes it nearly vitrified. The process depends on the environment conditions such as air humidity and temperature [58]. The evaporation process is very rapid, usually lasting less than 0.3s. Therefore, we ignore the drying process and assume that the aerosol cloud forming in the course of exhalation consists of fully evaporated droplets of fixed diameters.

**The exhaled aerosol forms a cloud in the shape of a cone.** All the released aerosol is mixed to a control volume of a breathing cone after the exhalation.

**Steady-state flow velocity field** The fluid flow is assumed to be fully developed under a certain constant mass flow rate at the inlets of an indoor ventilation system. Thus, the mean flow velocity field  $\mathbf{U}$  and the turbulent diffusivity is steady-state.  $\chi_t$  produced by a RANS model are assumed to be constant [52].

**The fate of virions in aerosol cloud.** The viability and, plausibly, infectious potential of the virions contained within an aerosol particle have been measured to decrease by half in about 1.2 h [39, 49]

**Incompressible flow with constant fluid properties** The constant flow properties are assumed in analyzing the transport of airborne.

In order to characterize the infection potential of the airborne aerosol, we introduce the infectivity field  $C(\mathbf{x}, t)$ . This is scalar field representing the local value of the number of active virions per unit volume of air. Its evolution in a turbulent flow is described by the convection-diffusion equation:

$$\frac{\partial C}{\partial t} + \mathbf{U} \cdot \nabla \mathbf{C} = \nabla \cdot (\chi_t \nabla \mathbf{C}) - \sigma \mathbf{C} + \mathbf{S} \quad (2.3)$$

where  $\mathbf{x}$  is space location,  $t$  is time,  $\sigma$  is the rate of decay of virus viability.  $S(\mathbf{x}, t)$  is the intensity of the source, which can be derived from the experimental data on expiratory droplet formation and virus load [44]. Time-independent velocity fields  $\mathbf{U}(\mathbf{x})$  and turbulent diffusivity  $\chi_t(\mathbf{x})$  of air flow are precomputed using a CFD solver, where steady-state equations describing the conservation of mass, momentum and energy as well as the equations for characteristics of turbulence in the form of a RANS model are solved.

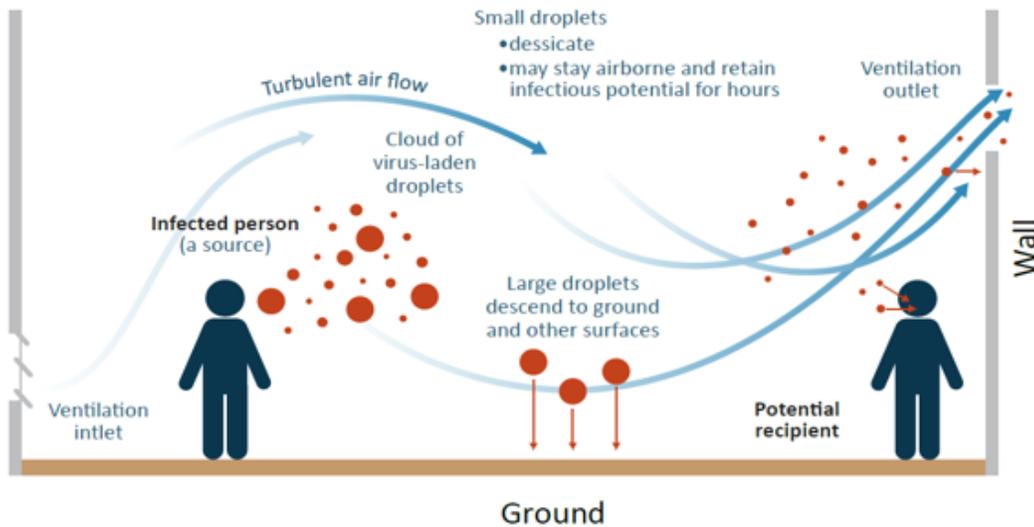


Figure 2.1: Schematic representation of various modes of respiratory infection transmission.

The exhaling forms a breathing cone-shaped cloud in front of the infected person illustrated in Figure 2.2. Previous studies have shown that the geometry of the aerosol breathing cone varies greatly depending on the person, climatic conditions, and exhalation behaviors. Our study defines a generic exhaling cone with a high density along the cone's axis while the aerosol gradual reduction towards its edges. Along with cone's axis, the peak value decreases exponentially with the distance from the mouth. We hypothesize that the defined breathing cone intensity change proportionally with the different exhaled intensities under different exhaling activities.

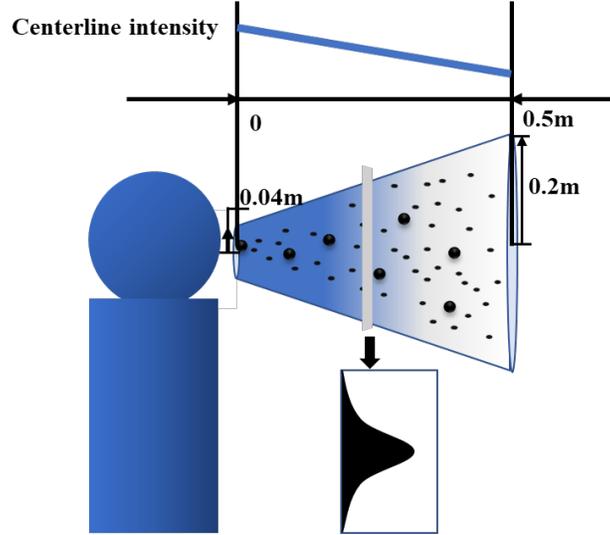


Figure 2.2: Breathing cone. Spatial distribution of the source intensity  $B(\mathbf{x})$  is shown.

Based on the experimental data, the particle concentration exhaled by an adult is: from 2.4 to  $5.2 \text{ cm}^{-3}$  per cough or 0.004 to  $0.223 \text{ cm}^{-3}$  per second. The total number of the droplet expelled ranged from 947 to 2085 per cough and 112 to 672 per second [58].

### 2.1.3 Consistent Steady-State Representation of Large-Scale Linear Systems

The equations have to be solved by CFD solvers, which convert (2.2) or (2.3) into the form of state-space representation that consists of millions of ordinary differential equations. The set of ordinary differential equations is built for the values of interests, i.e., values of temperature for (2.2) and values of infectivity field for (2.3) at the points of a discretization grid, in the consistent form:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.4)$$

where  $x(t) \in \mathbb{R}^n$  is the solution vector of grid-point values, the constant sparse matrix  $A \in \mathbb{R}^{n \times n}$  is the discretization coefficients matrix of the ODE system from (2.2) or (2.3), which incorporates the terms of the governing equations. We assume the asymptotically stable systems, so  $A$  is a

negative-definite matrix. The input matrix  $B \in \mathbb{R}^{n \times m}$  can be a vector for a single-input system ( $m = 1$ ) or a multi-column matrix for a multiple-input system ( $m \geq 2$ ). Each column vector in  $B$  represents the spatial distribution of the system's input, which is defined as a volumetric heat source in (2.2) or volumetric source of infectivity in (2.3). The system (2.2) or (2.3) is solved by a general CFD software, e.g., STAR-CCM+, Ansys Fluent, or OpenFOAM, as a 'black box' such that the system matrix  $A$  is not explicitly available. The systems can be approximated via reduced-order modeling, which is described shortly.

### 2.1.4 Boundary Condition as an Input

For an ODE system where a Dirichlet boundary condition is imposed, it specifies the values that a solution needs to take along the boundary of the domain. Such a boundary condition with a fixed or time-varying value often plays an important role in the dynamics of the system and can be considered as an input.

As an example, we consider the conjugate heat transfer of a battery pack, where the inlet temperature of the coolant is applied. The temperature of the inlet coolant is one of the essential parameters governing the conjugate heat transfer process of the system. Therefore, it would be useful to treat it as an input control parameter in (2.4). The Dirichlet boundary condition is represented by discretization coefficients  $\gamma$  in the system matrix:

$$\dot{x}(t) = Ax(t) + Bu(t) + \gamma \tag{2.5}$$

where  $B$  can be an input matrix that  $B = b_1$ . As we apply the inlet coolant temperature as an input control parameter,  $\gamma$  can be represented as an additional input vector such that  $\gamma = b_2 u_2$ . In this case, the system (3.4) can be represented as  $\dot{x}(t) = Ax(t) + \begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$  where the boundary condition is represented by  $b_2 = \gamma$  multiplied by a transient signal  $u_2$ , i.e., transient boundary condition temperature  $T_{BC}$ .

The spatial coefficients of boundary conditions  $\gamma$  need to be explicitly computed in order to

construct the boundary condition as an input control parameter. An additional extruded layer of Finite Volume cells at the inlet is created. The computation of  $\gamma$  mimic the convective heat flux at the inlet by volumetric heating within the extruded cells. Considering that the Finite Volume Method (FVM) is used for the CFD solver, the discretized form of the energy equation is:

$$\frac{d}{dt}(\rho T \mathbf{U}) + \sum_f [\rho T (\mathbf{U} \cdot \mathbf{a})]_f = \sum_f \left[ \frac{k}{C_p} \nabla T \cdot \mathbf{a} \right]_f + \left( \frac{\dot{q}}{C_p} \cdot Volume \right) \quad (2.6)$$

where  $\rho$ ,  $k$  and  $C_p$  are the density, heat conductivity and specific heat in the finite volume,  $T$  and  $\dot{q}$  are the temperature and volumetric heat source stored at the center of a finite volume cell.  $\mathbf{U}$  is the velocity field vector. The vector  $\mathbf{a}$  represents the surface area vector. *Volume* is the cell volume.

In (2.6), the convective flux term at a boundary face is expressed by:

$$[\rho T_i (\mathbf{U} \cdot \mathbf{a})]_f = \dot{m}_f \cdot T_{BC} \quad (2.7)$$

where  $\dot{m}_f$  is the mass flow rate at the boundary face.  $T_{BC}$  is the temperature value as boundary condition at the finite volume.

The diffusive flux term in (2.6) at a boundary face is:

$$\left[ \frac{k}{C_p} \nabla T_i \cdot \mathbf{a} \right]_f = \frac{k}{C_p} (T_{BC} - T_i) \cdot \vec{\alpha} \cdot \mathbf{a} \quad (2.8)$$

where  $\vec{\alpha} = \frac{\mathbf{a}}{a \cdot ds}$  and  $dS = x_f - x_0$  is the distance between face center and cell center and  $T_i$  is the temperature at the cell center.

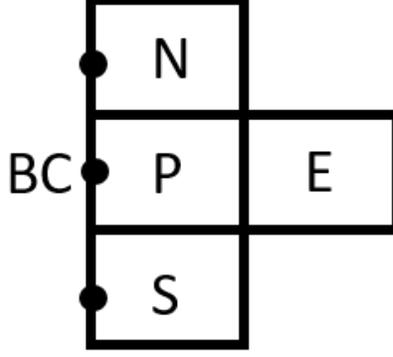


Figure 2.3: Boundary cells illustration

We see that the inlet temperature contributes to the free term of the discretization equation as illustrated in Figure 2.3, if all the Finite Volume cells adjacent the boundary have the same thickness, the contributions are the same for all boundary cells and can be rearranged as:

$$\begin{aligned}
 T_{BC} \frac{(C_{conv} + C_{cond}) \cdot A_{BC}}{V_{BC}} &= T_{BC} (C_{conv} + C_{cond}) \cdot \delta_{extrudedMesh}^{-1} \\
 &= T_{BC} \left( \dot{m}_f + \frac{k}{C_p} \right) \cdot \delta_{extrudedMesh}^{-1} = F = constant
 \end{aligned} \tag{2.9}$$

where  $C_{conv}$ ,  $C_{cond}$  are the coefficients of convective (2.7) and diffusive (2.8) flux term from discretization equation at a boundary cell.  $A_{BC}$  and  $V_{BC}$  are the boundary surface area and the volume of the boundary cell.  $\delta_{extrudedMesh}$  is the thickness of the extruded layer.

Evidently, the same contribution to the discretization equation is achieved by the internal heating with volumetric density  $F$  within each extruded cell. We choose this interpretation and add the heating as an input parameter while setting  $T_{BC}$  in the CFD problem to zero.

## 2.2 Reduced-Order Modeling Methodologies

### 2.2.1 Krylov-Subspace Method

The method utilizes projection onto the Krylov subspace to reduce the FOM system (2.4) into a system of lower order  $r$ . As discussed, e.g., in [26, 28, 29, 30, 34, 38, 59], the resulting ROM

matches the first  $r$  coefficients of the Taylor series expansion of the transfer function of the FOM. The coefficients are called ‘moments’ in the reduced-order model, whereas the model reduction based on the Krylov subspace is also called moment-matching model order reduction [38].

The Krylov subspace of order  $r$  is defined as [59]:

$$\kappa_r(\tilde{A}, \tilde{b}) = \text{span} \left\{ \tilde{b}, \tilde{A}\tilde{b}, \dots, \tilde{A}^{(r-1)}\tilde{b} \right\}, \quad (2.10)$$

where  $\tilde{A}$  is a constant matrix and  $\tilde{b}$  is a starting vector generating the subspace. For the purpose of deriving an ROM for conjugate heat transfer problems, which approximates the original system’s behavior in the low frequency domain, the appropriate selection is  $\tilde{A} = A^{-1}$  and  $\tilde{b} = A^{-1}B$  [27]. On the other hand, the ROM is generated in a broader range of frequency for airborne transmission analysis, the details will be discussed shortly.

The model reduction is achieved by projection of the FOM solution vector  $x(t)$  on the Krylov subspace [26]. With an orthonormal matrix  $V$ , whose column vectors span the subspace, this is expressed as

$$x \approx Vx_r, \quad (2.11)$$

where  $x_r$  is the vector of the approximate solution in the Krylov subspace.

The system (2.4) is approximated as

$$V\dot{x}_r = AVx_r + Bu. \quad (2.12)$$

The ODEs for  $x_r(t)$  (the actual ROM) are obtained by the Petrov-Galerkin projection using another orthonormal matrix  $W \in \mathbb{R}^{n \times r}$ . Following [29], one-sided Arnoldi algorithm is utilized, which implies  $W = V$ , so  $W^T V = I$ , where  $I$  is the identity matrix. Left-multiplying (2.12) by  $W^T$  we obtain

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t), \quad (2.13)$$

where  $A_r = W^T A V \in \mathbb{R}^{r \times r}$  and  $B_r = W^T B \in \mathbb{R}^{r \times m}$ .

The first  $r/m$  moments of the transfer function of the ROM obtained in this manner match the first moments of the full system [30, 29]. Since  $r \ll n$ , the reduced order system (2.13) can be solved with a much smaller computational effort. Once the system is solved, an approximation of the solution state vector in the full-order space can be reconstructed as  $x(t) = Vx_r(t)$ .

The two-sided Arnoldi algorithm, where  $W \neq V$ , can potentially improve the moment-matching property [38]. However, this is true only when the order of the outputs is considerably smaller than the order of the original system [38]. Since the temperature distribution in the entire battery pack is of primary interest in our problem, the output consists of  $n$  values. Therefore, only the one-sided Arnoldi algorithm is employed in this study.

## 2.2.2 Parametric Reduced-Order Modeling

The parametric reduced-order modeling (pROM) is presented in this section. The model allows us to generate ROM for an arbitrary value of the system parameter entering the system matrix  $A$ , e.g. the mass flow rate of coolant, and input matrix  $B$  using interpolation from the reduced order models already developed for several values of the parameter [60].

We firstly introduce the FOMs with  $k$  different parameters, which are obtained from several mass flow rates:

$$\dot{x}_i = A_i x_i + B_i u. \quad (2.14)$$

where parameter-dependent matrices  $A_i \in \mathbb{R}^{n \times n}$  and  $B_i \in \mathbb{R}^{n \times 1}$  are spatial discretization coefficient in system matrices and heat input, respectively.

To start with the construction of pROM,  $k$  FOMs are reduced independently with its individual projection matrices  $W_i, V_i$ . Thus, we have  $k$  developed ROMs, hereafter called local ROMs:

$$W_i^T V_i \dot{x}_{r,i} = W_i^T A_i V_i x_{r,i} + W_i^T B_i u \quad (2.15)$$

where the reduced state vectors  $x_{r,i}$  in each ROMs belong to different vector spaces since they are

projected by different local projection matrices, i.e.,  $x_i \approx V_i x_{r,i}$ . To interpolate between them, we need to transform the local ROMs into a common subspace. The transformation is achieved by applying the singular value decomposition (SVD) to the local projection matrix dataset  $V_{all} = \begin{bmatrix} V_1 & V_2 & \dots & V_k \end{bmatrix}$ . The first  $r$  columns of the orthogonal matrix  $U$  from the SVD form the transformation matrix  $R \in \mathbb{R}^{n \times r}$ . After  $R$  is formed, each local projection matrix  $V_i$ ,  $W_i$  can be transformed into a common reduced-order space, where every local ROMs have a consistent interpretation. Each local projection matrix is transformed by the matrix  $R$ , so that the transformed local ROM gives:

$$M_i T_i^{-1} \dot{x}_{r,i}^* = M_i W_i^T A_i V_i T_i^{-1} x_{r,i}^* + M_i W_i^T B_i u \quad (2.16)$$

where  $M_i = (W_i^T R)^{-1}$  and  $T_i = R^T V_i$ .  $x_{r,i}^*$  is the transformed state vector, i.e.,  $x_{r,i} = T_i^{-1} x_{r,i}^*$ .

Now, the transformation matrices  $M_i$  and  $T_i$  are computed to allow the transition from each coordinates of local ROMs vector  $x_{r,i}$  to a modified coordinate system  $x_{r,i}^*$  that share the same basis  $R$ . The pROM can be constructed by interpolation with the weighted coefficient  $\varpi_i$ . Given a pROM with parameter  $p$ , the weighted coefficients are obtained such that  $\sum_{i=1}^k \varpi_i(p) = 1$ . Various interpolation methods can be applied for the weighted coefficients such as linear or quadratic interpolation. For example, when two local ROMs are selected to construct the pROM, linear interpolation can be applied in calculating the weighted coefficients.

The parametric reduced-order model can be constructed as:

$$\sum_{i=1}^k \varpi_i M_i T_i^{-1} \dot{x}_r^* = \sum_{i=1}^k \varpi_i M_i W_i^T A_i V_i T_i^{-1} x_r^* + \sum_{i=1}^k \varpi_i M_i W_i^T B_i u \quad (2.17)$$

The pROM ODE is:

$$\dot{x}_{pmor}^*(t) = A_{pmor} x_{pmor}^*(t) + B_{pmor} u(t) \quad (2.18)$$

where the ROM system matrix  $A_{pmor} = (\sum_{i=1}^k \varpi_i M_i T_i^{-1})^{-1} (\sum_{i=1}^k \varpi_i M_i W_i^T A_i V_i T_i^{-1})$  and  $B_{pmor} = (\sum_{i=1}^k \varpi_i M_i T_i^{-1})^{-1} (\sum_{i=1}^k \varpi_i M_i W_i^T B_i)$ . The state vector of the full-order system is

reconstructed from parametric reduced-order state vector as  $\sum_{i=1}^k \varpi_i V_i T_i^{-1} x_{pmor}^*(t)$ .

The results of pROM will be presented later which shows excellent approximation with original high-fidelity FOMs.

### 2.2.2.1 Time-Varying Parameters Using the pROM Framework

We developed the applications with time-varying parameters when the system matrix  $A(t)$  and input matrix  $B(t)$  change with time under the current framework of the pROM. The application aims to predict the batteries' thermal behavior under a time-varying mass flow rate signal. Since the mass flow rate is changing over time, the state-space representation of the system becomes:

$$\dot{x}(t) = A(t)x(t) + B(t)u \quad (2.19)$$

where  $A(t)$  is the system matrix that changes with time. We can apply the parametric model to approximate the system matrix by interpolation (2.17).

The ROM for the time-varying MFRs is:

$$\dot{x}_r(t) = A_r(t)x_r(t) + B_r(t)u \quad (2.20)$$

where  $A_r(t)$  and  $B_r(t)$  are interpolated based on MFRs. The procedure is a generalization of the interpolation procedure described in section 2.2.2. For every value of the time step  $t_j = t_0 + j\Delta t$ , we use the instantaneous mass flow rate  $p(t_j)$  to compute the interpolation coefficients  $\varpi_i(t_j)$  and then the matrices  $A_r(t_j)$  and  $B_r(t_j)$ . In this case, the parametric model can be applied to approximate the system matrix by interpolation. The discussion and limitation of the implementation will be shown shortly.

## CHAPTER 3

# Algorithms

We introduce the indirect Arnoldi algorithms for single-input systems in 3.1 and multiple-input systems in 3.2. The frequency-shift Arnoldi-type algorithm is introduced in 3.3. The algorithm developed for parametric reduced-order modeling is presented in 3.4. Finally, we present the change of variable method in 3.5 for the systems, in which inhomogeneous Dirichlet boundary conditions are applied.

The proposed reduced-order modeling (ROM) approach and the developed algorithms for single-input and multiple-input systems in the current chapter have been successfully implemented in our GM-funded project on conjugate heat transfer in the battery pack of an electric vehicle [61]. Further development of the algorithms for airborne respiratory transmission systems is carried out by the frequency-shift algorithm.

### 3.1 Indirect Arnoldi Algorithm for Single-Input Systems

The indirect version of the Arnoldi algorithm proposed in [29] and presented in **Algorithm 1** is used for problems with a single input, i.e., with  $m = 1$ . The version is different from the original Arnoldi algorithm because it computes the reduced system matrix as  $A_r = V^T X$  rather than  $V^T A V$ . The orthogonalization matrix  $X = A V$  is constructed simultaneously with  $V$ . It contains the information from the system matrix  $A$  and makes direct access to  $A$  unnecessary. As one can see in **Algorithm 1**, the column vector  $x_i$  of  $X$  is generated as the basis vector  $v_{i-1}$

orthonormalized through the modified Gram-Schmidt process with respect to the previous basis vectors in  $X$ .

The key (and the only computationally demanding) element of the algorithm is the calculation of the matrix products  $\tilde{v}_1 = A^{-1}b_1$  and  $\tilde{v}_i = A^{-1}v_{i-1}$ ,  $i = 1, \dots, r$ . They can be obtained without direct access to  $A$  by applying the CFD solver to steady-state heat transfer problems, discretization of which corresponds to  $A\tilde{v}_1 = b_1$  and  $A\tilde{v}_i = v_{i-1}$ . For example, the initialization step of **Algorithm 1** includes a CFD solution of the steady-state version of (2.4) with the source term  $-b_1$ . The computed values of temperature at the grid points form the vector  $\tilde{v}_1$ . The following steps require similar CFD solutions of heat transfer problems with source terms determined by the previously generated vector  $v_{i-1}$ .

The basis vectors  $v_i$  are computed by taking the recursive orthogonalization of  $A^{-1}v_{i-1}$  to the previously generated orthonormalized basis vectors through the modified Gram-Schmidt process. The basis vectors  $x_i$  are computed and orthonormalized to satisfy  $x_i = Av_i$ , accordingly. Once  $V$  and  $X$  are computed, the ROM matrix is formed by  $A_r = V^T X$ .

---

**Algorithm 1:** Indirect Arnoldi algorithm for single-input systems

---

1) Initialization:

$$\tilde{v}_1 = A^{-1}b_1; \tilde{x}_1 = b_1;$$

$$v_1 = \frac{\tilde{v}_1}{\|\tilde{v}_1\|}; x_1 = \frac{\tilde{x}_1}{\|\tilde{x}_1\|};$$

2) Arnoldi iterative process:

**for**  $i = 2 : r$  **do**

a)  $\tilde{v}_i = A^{-1}v_{i-1}$ ;

b)  $\tilde{x}_i = v_{i-1}$ ;

c) Orthogonalization:

**for**  $j = 1 : i - 1$  **do**

|  $h = \tilde{v}_i v_j; \tilde{v}_i = \tilde{v}_i - hv_j; \tilde{x}_i = \tilde{x}_i - hx_j$ ;

**end**

d) Normalization:

$$v_i = \frac{\tilde{v}_i}{\|\tilde{v}_i\|}; x_i = \frac{\tilde{x}_i}{\|\tilde{x}_i\|};$$

**end**

3) Set  $V$  and  $X$ :

$$V = [ v_1 \quad \dots \quad v_r ]; X = [ x_1 \quad \dots \quad x_r ];$$

4) Construct  $A_r$ :

$$A_r = V^T X;$$


---

## 3.2 Indirect Arnoldi Algorithm for Multiple-Input Systems

Several Krylov-subspace methods have been developed to create ROMs for systems with multiple inputs ( $m \geq 2$ ). Successful attempts include, e.g., the block Arnoldi method [37] and the two-sided Arnoldi method for multiple-input and multiple-output systems [38]. The block Arnoldi method is analogous to the Arnoldi algorithm, but operates with the block matrix  $A^{-1}B$ . It efficiently utilizes the block Arnoldi recursion through the orthogonalization process and basis deflation [37]. Compared with that, the two-sided Arnoldi method constructs the matrices  $V$  and  $W$  from the input and output Krylov subspaces of any order  $r$  independently of the number of inputs and outputs.

The common drawback of these two methods is that they require access to the system matrix  $A$ . To overcome this problem, a new method presented by **Algorithm 2** is developed in this work. It uses the one-sided Arnoldi approach and is suitable for large-scale multiple-input systems, for which the system matrix is unavailable.

The column vectors in  $B = \begin{bmatrix} b_1 & \cdots & b_m \end{bmatrix}$  are utilized as starting vectors. At the initialization stage, the algorithm computes the first  $m$  orthogonal basis vectors  $v_i$  by recursive orthogonalization of the steady-state solution  $\tilde{v}_i = A^{-1}b_i$  with respect to the previously generated orthonormal basis vectors. Note that the starting vectors in  $B$  must be linearly independent to ensure the orthogonality. In the following Arnoldi iterative process (see **Algorithm 2**), every newly generated basis vector  $v_i$  is computed through orthogonalization of the steady-state solution  $\tilde{v}_i = A^{-1}v_{i-m}$  with respect to the previously orthonormalized basis vectors followed by normalization. It is required that the newly generated basis vector is linearly independent of the existing basis vectors in  $V$ . If this condition is not satisfied, the deflation procedure [38], by which the newly computed vector and its derivative recursive vectors are deleted, is utilized. Practically, the near linear dependency of the generated basis is identified as the situation, in which the norm of the newly computed orthogonal vector becomes smaller than  $\varepsilon = 10^{-3}$ .

Since the system matrix  $A$  is not explicitly available, the orthogonalization matrix  $X = AV$ , such that  $A_r \equiv V^T AV = V^T X$ , is computed by the algorithm. Every new column vector  $x_i$  of  $X$

is the results of orthonormalization of  $\tilde{x}_i = A\tilde{v}_i$ . Throughout the process of orthogonalization and normalization, the relationship of the basis vectors,  $x_i = Av_i$ , is kept valid as:

$$\begin{aligned}
 Av_i &= A \frac{A^{-1}v_{i-m} - V^{i-1} \begin{bmatrix} h_1 & h_2 & \cdots & h_{i-1} \end{bmatrix}^T}{\|\tilde{v}_i\|} \\
 &= \frac{x_{i-m} - X^{i-1} \begin{bmatrix} h_1 & h_2 & \cdots & h_{i-1} \end{bmatrix}^T}{\|\tilde{v}_i\|} = x_i,
 \end{aligned} \tag{3.1}$$

where  $h_i$  is the coefficient of the Gram-Schmidt orthogonalization and  $V^{i-1}$ ,  $X^{i-1}$  denote the already constructed matrices containing the first  $i - 1$  vectors of  $V$ ,  $X$ . The **Algorithm 2** thus computes  $A_r$ , equivalent to the matrix  $A_r$  found by the conventional one-sided Arnoldi algorithm.

---

**Algorithm 2:** Indirect Arnoldi algorithm for multiple-input systems

---

1) Choose starting vectors  $b_i$  (linearly independent):

$$B_{n \times m} = [ b_1 \quad \dots \quad b_m ]$$

2) Initialization of starting vectors:

**for**  $i = 1 : m$  **do**

$$\tilde{v}_i = A^{-1}b_i; \tilde{x}_i = b_i;$$

**if**  $i \neq 1$  **then**

**for**  $j = 1 : i$  **do**

$$h = \tilde{v}_i v_j; \tilde{v}_i = \tilde{v}_i - hv_j; \tilde{x}_i = \tilde{x}_i - hx_j;$$

**end**

**end**

$$v_i = \frac{\tilde{v}_i}{\|\tilde{v}_i\|}; x_i = \frac{\tilde{x}_i}{\|\tilde{v}_i\|};$$

**end**

3) Arnoldi iterative process with deflation:

**for**  $i = m + 1 : r$  **do**

a)  $\tilde{v}_i = A^{-1}v_{i-m};$

b)  $\tilde{x}_i = v_{i-m};$

c) Orthogonalization:

**for**  $j = 1 : i - 1$  **do**

$$h = \tilde{v}_i v_j; \tilde{v}_i = \tilde{v}_i - hv_j; \tilde{x}_i = \tilde{x}_i - hx_j;$$

**end**

d) Deflation:

**if**  $\|\tilde{v}_i\| < \varepsilon$  **then**

$$m = m - 1;$$

**if**  $m \neq 0$  **then**

| Continue;

**else**

| Break;

**end**

$$v_i = \frac{\tilde{v}_i}{\|\tilde{v}_i\|}; x_i = \frac{\tilde{x}_i}{\|\tilde{v}_i\|};$$

**end**

4) Set  $V$  and  $X$ :

$$V = [ v_1 \quad \dots \quad v_r ]; X = [ x_1 \quad \dots \quad x_r ];$$

5) Construct  $A_r$ :

$$A_r = V^T X;$$

---

### 3.3 Frequency-Shift Arnoldi Algorithm

We introduce a new algorithm base on Krylov subspace method: Frequency shift Arnoldi Algorithm. The algorithm establishes a more accurate approximation for Krylov subspaces by the selection of non-zero frequencies, i.e.,  $\sigma \in \mathbb{C}$ , in describing the moment matching [26].

By defining the Krylov subspace of shifted frequency, the ROM approximates characteristic, under the selections of frequency points, from system's response within a broad range of frequency, i.e.,

$$\begin{bmatrix} \sigma_1 & \sigma_2 & \cdots & \sigma_n \end{bmatrix}. \quad (3.2)$$

We define the Krylov subspaces as:

$$\kappa_r \left( (\tilde{A} - \sigma_i \mathbf{I}), \tilde{b} \right) = \text{span} \left\{ \tilde{b}, (\tilde{A} - \sigma_i \mathbf{I})\tilde{b}, \dots, (\tilde{A} - \sigma_i \mathbf{I})^{(r-1)}\tilde{b} \right\}, \quad (3.3)$$

where  $\sigma_i$  represents the selected frequency shift points that are distributed within a range of frequency. The Taylor series expansion of moment at shifted frequencies (3.2) involves shifted moments given by (3.3). In the expansion of the Taylor series, one can choose frequency shift points  $\sigma$  according to system's characteristics to improve the approximation accuracy. For example, we define  $\sigma$  at high-frequency range that helps ROM captures the system's high-frequency dynamics. Compared with that, ROM misses part of the high-frequency characteristic if ROM is built based on SIMO algorithm in the previous sections. Note that systems with high-frequency response can be approximated more accurately by frequency-shift algorithm.

In algorithm 3, we define a total of  $n$  frequency points,  $\sigma_s$ . Per each  $\sigma$ , we construct the orthogonalized vector set following the Arnoldi process that spans (3.3). Each of the vector set has  $m+1$  ortho-normalized vectors, which indicate up to  $m^{\text{th}}$  moments are matched between ROM and FOM. The vectors from  $n$  vectors sets are organized from  $i=1$  to  $n$  per each moment to form vector set  $\hat{V}$ . As a final step, we get the projection matrix  $V$  by applying modified Gram-Schmidt orthogonalization to vector candidates set  $\hat{V}$ . Each vector in  $\hat{V}$  are explicitly orthogonalized against all the previous basis vectors. Note that  $V$  is orthonormal, and its columns form a basis that spans every Krylov subspace in a defined frequency shift.

The basis vectors  $x_i$  are computed as  $x_i = Av_i$  because one-sided Arnoldi algorithm is used. Once  $V$  and  $X$  are computed, the ROM system matrix is formed by  $A_r = V^T X$ .

---

**Algorithm 3:** Frequency-shift Arnoldi algorithm for single-input systems

---

$n$  sampling points are selected:

$$\begin{bmatrix} \sigma_1 & \sigma_2 & \cdots & \sigma_n \end{bmatrix}$$

1) Initialization:

**for**  $i = 1 : n$  **do**

$$\begin{array}{|l} \tilde{v}_i = (A - \sigma_i \mathbf{I})^{-1} b_1; \\ \hat{v}_i = \frac{\tilde{v}_i}{\|\tilde{v}_i\|}; \end{array}$$

**end**

2) Generation of Krylov subspace at each frequency shift by Arnoldi iterative process:

**for**  $k = 1 : m$  **do**

**for**  $i = 1 : n$  **do**

        a) Getting CFD solution:

$$\tilde{v}_{kn+i} = (A - \sigma_i \mathbf{I})^{-1} \hat{v}_{(k-1) \times n + i};$$

        b) Orthogonalization:

**for**  $j = 1 : k$  **do**

$$\begin{array}{|l} h = \tilde{v}_{kn+i} \hat{v}_{(j-1) \times n + i}; \\ \tilde{v}_{kn+i} = \tilde{v}_{kn+i} - h \hat{v}_{(j-1) \times n + i}; \end{array}$$

**end**

        c) Normalization:

$$\hat{v}_{kn+i} = \frac{\tilde{v}_{kn+i}}{\|\tilde{v}_{kn+i}\|};$$

**end**

**end**

3) Set  $V$  and  $X$ :

a) Apply Gram-Schmidt Orthogonalization to vectors set:

$$\hat{V} = \begin{bmatrix} \hat{v}_1 & \hat{v}_2 & \cdots & \hat{v}_{(m+1)n} \end{bmatrix};$$

b) Get matrices:

$$V = \begin{bmatrix} v_1 & \cdots & v_{(m+1)n} \end{bmatrix};$$

4) Construct  $A_r$ :

$$A_r = V^T A V;$$

---

### 3.4 Parametric Reduced Order Modeling Algorithm

The construction of the parametric reduced-order model is executed following the algorithm 4.

Note that  $\varpi_i$  are the weighting coefficients for interpolation as described in Section 2.2.2.

---

**Algorithm 4:** Parametric Model Order Reduction algorithm

---

1) Choice of  $R$ :

Apply Singular Value Decomposition (SVD) to the local projection matrix collection  $V_{all}$ .

$$V_{all} = \begin{bmatrix} V_1 & V_2 & \dots & V_k \end{bmatrix};$$

Choosing  $R$  as the first  $r$  columns of  $U$  from the SVD of  $V_{all}$ .

$$V_{all} = U\Sigma V^T$$

(where  $U$  is a  $k \times k$  real orthonormal matrix,  $\Sigma$  is a  $k \times n$  rectangular diagonal matrix with non-negative real numbers on the diagonal, and  $V$  is an real unitary matrix.)

2) Calculation of matrices  $M$  and  $T$  for local ROMs:

**for**  $i = 1 : n$  **do**

$$\left| \begin{array}{l} \text{a) } T_i = R^T V_i; \\ \text{b) } M_i = (W_i^T R)^{-1}; \end{array} \right.$$

**end**

3) Construction of pMOR based on weighing coefficients  $\varpi_i$ :

$$\sum_{i=1}^k \varpi_i M_i T_i^{-1} \dot{x}_r^* = \sum_{i=1}^k \varpi_i M_i W_i^T A_i V_i T_i^{-1} x_r^* + \sum_{i=1}^k \varpi_i M_i W_i^T B_i u;$$

4) Reconstruction of pMOR :

$$C_{pmor}^* = \sum_{i=1}^k \varpi_i V_i T_i^{-1};$$

$$y_r^* = C_{pmor}^* x_{pmor}^*$$

---

### 3.5 Change of Variable Method

The change of variable method can be applied when the input vector  $u(t)$  in system (2.4) has time-invariant components. For example, if the system with  $m = 2$

$$\dot{x}(t) = Ax(t) + \begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

has a time-varying signal  $u_1$  and a constant signal  $u_2$ , it can be rewritten as

$$\dot{x}(t) = Ax(t) + Bu(t) + \gamma, \quad (3.4)$$

where  $\gamma = b_2 u_2 = \text{const}$  and  $Bu(t) = b_1 u_1(t)$ . The solution of (3.4) can be obtained by the change of variable:

$$x(t) = x^*(t) - A^{-1}\gamma \quad (3.5)$$

where  $x^*(t)$  is the solution of the system (3.4) with  $\gamma = 0$  and  $-A^{-1}\gamma$  is the steady-state response of the system to the constant input  $\gamma$ . Substituting (3.5) into (3.4) results in a single-input system (2.4) for  $x^*$ .

The modification can be used to exclude the constant input from the process and, thereby, reduce the number of inputs  $m$ , thus reducing the computational cost of the ROM creation. The steady-state response  $-A^{-1}\gamma$  can be found a posteriori as an FOM solution of the steady-state problem with the single constant input  $\gamma$ .

Practically, the change of variable method is useful for ROM approximation of heat transfer with Dirichlet boundary conditions on temperature, for example, for BTMS with prescribed temperature at the inlet of the cooling channels (see section 4.3). In that case,  $x^*(t)$  is obtained as the ROM approximation of the solution of (2.4) with zero inlet temperature, while  $-A^{-1}\gamma$  is the full steady-state solution obtained with true temperature boundary values and zero internal heating rate, which needs to be computed only once.

## CHAPTER 4

# Reduced-Order Modeling for Conjugate Heat Transfer in a Battery Pack

The work described in this section is presented in detail in [61]. The proposed methods are applied to a large-scale CFD model of conjugate heat transfer in a battery pack. After an ROM is derived using **Algorithm 1**, **Algorithm 2**, **Algorithm 4** and the change of variable method, the accuracy of its prediction is tested in a series of scenarios of unsteady behavior. In each test, the model system of equations (2.13) is solved with appropriate input and initial conditions. The solution  $x_r(t)$  is converted into a full-order approximation of temperature field using (2.11). The outcome is compared with the the temperature field obtained in unsteady FOM CFD solution.

A comment is in order concerning the rather high heating rates and, respectively, internal temperature variations observed in the tests. The typical heat generation in an operating automotive battery cell of the type considered below varies in the range between 1 and 3 W. Stronger heating, sometimes exceeding 10 W is found in exceptional situations during extreme conditions (e.g., charging and discharging of 4 C and higher currents). Such exceptional situations are intentionally used for several test cases in our work because they are particularly challenging for accurate modeling and, therefore, particularly suitable for validation of the proposed ROM method.

The physical model and its CFD discretization are described in section 4.1. The accuracy of ROM and measures of error are discussed in section 4.2. In section 4.3, the results for single-input systems are presented using two different input signals. The performance of the ROMs of multiple-input systems is discussed in section 4.4.

## 4.1 Battery Pack Model

A model battery pack is used to test the method. While retaining the typical geometry and physical properties of an automotive battery pack, the model does not correspond to any specific manufactured product. Details of battery design not directly relevant to heat transfer and BTMS are ignored. In particular, the processes of Joule dissipation, phase change, and mixing are not explicitly considered. Their cumulative effect on the internal energy balance within the system (see, e.g., [56]) is simulated as volumetric heat generation treated as an input in our analysis.

As illustrated in Figure 5.5, the pack includes nine battery cells enclosed in aluminum casing and attached to dual coolant channels. In Figure 5.5, the leftmost casing is displayed as partially transparent for illustration purposes. The cells are cooled by heat transfer to the cooling liquid flowing in the channels.

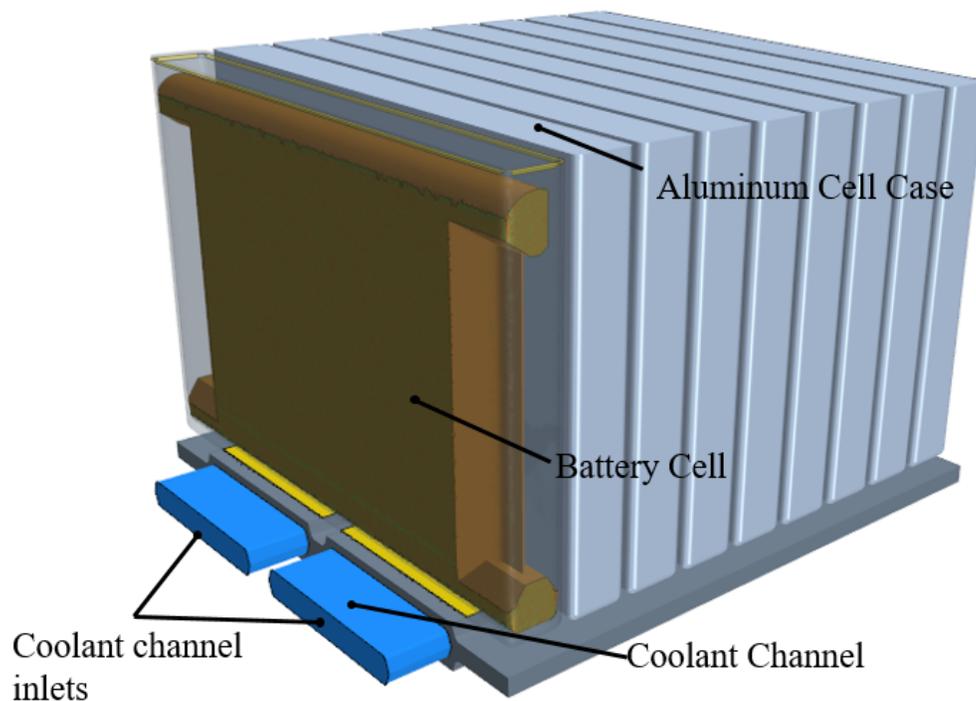


Figure 4.1: Schematic of the battery pack with dual coolant channels.

The parameters of the model are summarized in Table 4.1. Each battery can be viewed as representing an NMC-graphite cell with an approximate capacity of 10 Ah. The physical properties

of the pack materials listed in Table 4.1 correspond to a typical design [62]. Aluminum casing and water as a coolant are assumed. The interiors of the batteries are assumed to be uniform, with constant density and specific heat and constant anisotropic thermal conductivity.

Table 4.1: Model specifications.

CFD Model mesh size	up to 880,862, 1,071,863 or 1,457,717	
CFD Model Specifications	Laminar	
	Segregated Flow	
	Segregated Fluid Enthalpy	
Model Components Properties	Solid: Aluminum	$\rho = 2702.0 \text{ kg/m}^3$
		$C_p = 903.0 \text{ J/kg-K}$
		$\kappa = 237.0 \text{ W/m-K}$
	Solid: Cell Bulk	$\rho = 2560.0 \text{ kg/m}^3$
		$C_p = 975 \text{ J/kg-K}$
		Anisotropic $\kappa$ (W/m-K): $0.95 \times 30.8$ (through/in-plane)
	Liquid: $H_2O$	$\rho = 997.561 \text{ kg/m}^3$
		$C_p = 4181.72 \text{ J/kg-K}$
		$\kappa = 0.620271 \text{ W/m-K}$
Boundary Conditions	Inlet	Temperature: $15 \text{ }^\circ\text{C}$
		Mass Flow Rate: $0.01 \text{ kg/s}$
	Outlet	Pressure Outlet: $p = 0 \text{ Pa}$
	Pack Exterior	Wall

The boundary condition on the solid part is zero heat flux (an adiabatic wall). At the inlet of the cooling channels, uniform velocity corresponding to the mass flow rate of  $0.01 \text{ kg/s}$  is imposed. This induces a laminar flow in channels. A constant value of temperature is imposed at the inlet of the cooling channels. The inlet temperature value of  $15 \text{ }^\circ\text{C}$  is used in the examples below.

The commercial solver STAR-CCM+ [36] is used for the full-order CFD solution. The heat transfer equations (2.1) and (2.2) are spatially discretized in the entire domain on an unstructured finite-volume grid adapted to the battery geometry (for example, thin finite-volume cells are used to resolve the walls of the aluminum casing). Three grids differing from each other by the base cell size have been tested: with 880,862, 1,071,863, and 1,457,717 cells. A grid sensitivity study conducted in the form of transient simulations of the response of the pack to constant-rate heating of all batteries have shown no significant difference between the results obtained with the three grids. In particular, the grid-related changes of the average temperatures of individual batteries and the temperatures recorded at four probe points have not exceeded 0.5% of the maximum variation of temperature within the pack during the entire process. Based on these data and with the goal of better demonstration of the capabilities of the order reduction method, the FOM on the largest grid of 1,457,717 cells is used for ROM development and ROM-FOM comparison in the analysis presented in the rest of this paper.

In order to impose the Dirichlet boundary condition corresponding to the constant cooling temperature at the inlet, the change of variable method described in section 3.5 is used. The vector  $-A^{-1}\gamma$  is calculated as the solution at zero internal heat sources and inlet temperature of 15 °C. Due to the boundary conditions of zero normal temperature gradients applied at all the other boundaries of the model, no calculations are, in fact, necessary in our case. The solution is simply the uniform temperature field  $T = 15^\circ\text{C}$ .

The FOM simulations are performed on a workstation with 20 cores installed on two Intel Xeon Silver 4114 processors. The ROM solutions are carried out on a personal computer using two cores.

## 4.2 Error of ROM Approximation

As discussed in section 3.1, the ROM approximation is derived to match the first  $r/m$  moments of the FOM transfer function. The higher the reduced order  $r$ , the more moments are matched,

yielding more accurate approximation. The downside of using higher  $r$  is the higher computational costs of development and application of the ROM. The physical nature of the conjugate heat transfer suggests that the dynamic characteristics of the FOM are dominated by the first few moments, so large values of  $r$  may, in fact, be unnecessary. One anticipates existence of an optimal order  $r$ , such that its further increase does not significantly improve the accuracy of the ROM approximation. The tests presented below fully confirm this expectation.

An important question arising in the tests is that of an appropriate measure of the accuracy of an ROM approximation. The error of the approximation is defined as the vector

$$\varepsilon = x(t) - Vx_r(t) \quad (4.1)$$

that shows the difference between the FOM solution  $x(t)$  and the approximate solution  $Vx_r(t)$  reconstructed from the ROM at every mesh point. The accuracy of the approximation is evaluated in our work by the root-mean-square and infinity norms of the vector:

$$\|\varepsilon\|_2 = \left( \frac{1}{n} \varepsilon^T \varepsilon \right)^{1/2}, \quad (4.2)$$

$$\|\varepsilon\|_\infty = \max |\varepsilon|. \quad (4.3)$$

### 4.3 Single-Input Cases

A single-input problem appears in the situation when all 9 battery cells generate heat according to the same schedule. To explore the ROM performance, models of different reduced order  $r$  are developed and applied to two characteristic scenarios: with a step input signal, and with a realistic rapidly changing input. The initial temperature distribution is uniform at  $T = 15^\circ\text{C}$ .

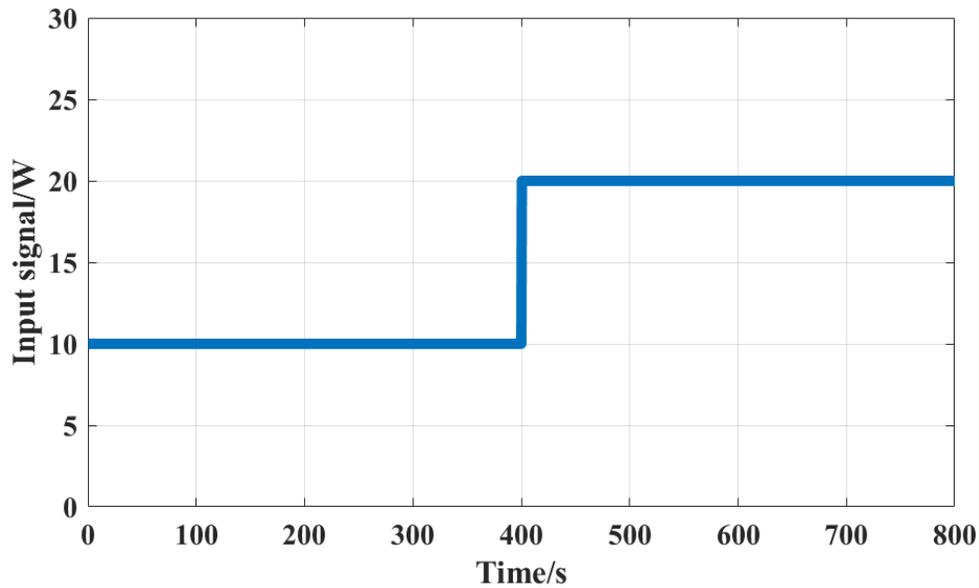


Figure 4.2: Heat input signal for the test case 1 – single input in the form of a stepwise heating rate.

In the first case, each battery cell produces 10 W at  $0 \leq t \leq 400$ s and 20 W after that (see Figure 4.2). As the first illustration, we show how the order  $r$  of the ROM affects the simulated evolution of temperature at one probed point inside the domain. The point's location is shown in Figure 4.3. The results obtained for  $r = 1, 3, 10,$  and  $15$  are presented in Figure 4.4.

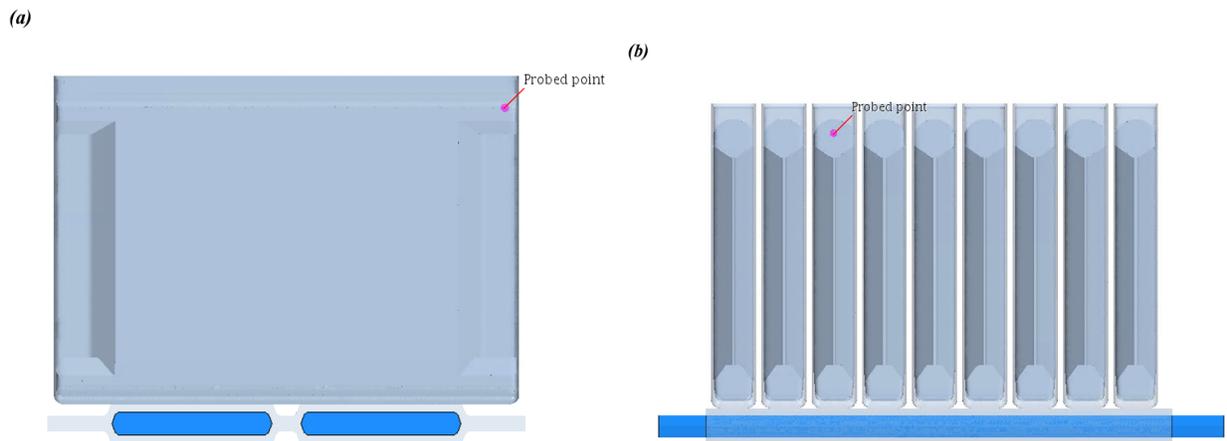


Figure 4.3: Location of the point within the battery pack, at which temperature is probed. (a) Front view; (b) Side view.

We see that the ROM of order one is clearly inaccurate. Much better accuracy is obtained by the models with  $r \geq 3$ . Some small-amplitude, but still unrealistic fluctuations of temperature are shown by the ROMs with  $r = 3$  and  $r = 5$  soon after the stepwise change of the input, approximately at  $t = 2s$  and  $402s$ . Very accurate results (no discernable difference with the FOM curve) are produced by the ROM with  $r = 15$ . Evidently, the data at a single probed point cannot comprehensively reflect the approximation accuracy. Thus, the norms (4.2) and (4.3) of the error are computed and shown in Table 4.2.

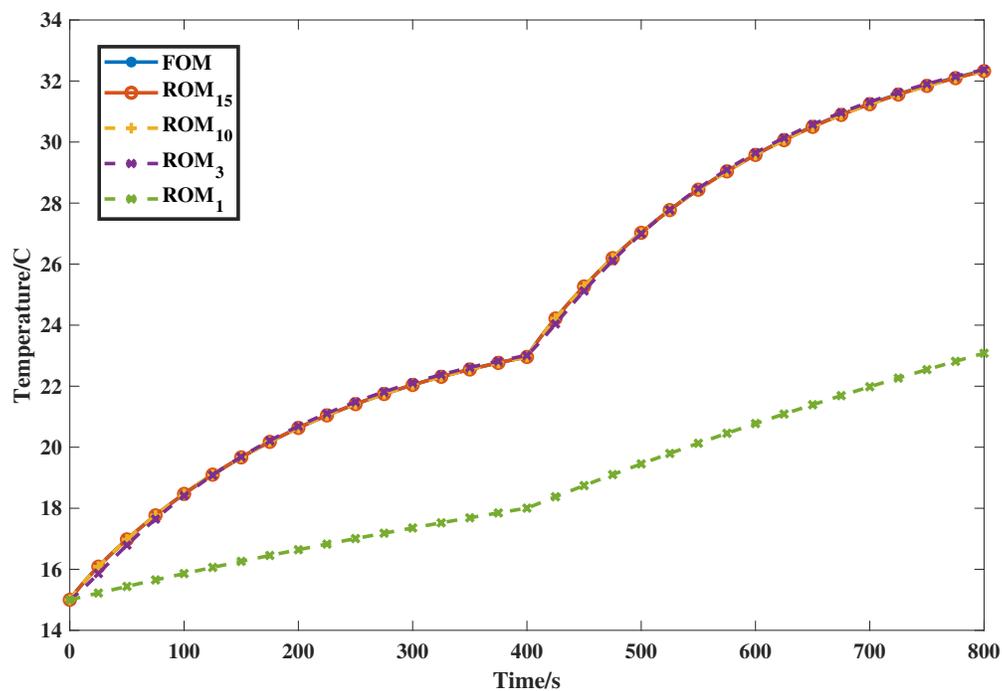


Figure 4.4: Comparison between ROM and FOM solutions for the test case 1 (see Figure 4.2). Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of  $r$  are shown.

Table 4.2: The norms (in °C) of the error at various time instants for the test case 1 (see Figures 4.2, 4.4 and 4.5).

r	402s		410s		800s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
1	0.28265	0.42378	0.29965	0.45800	0.46674	0.66457
2	0.00534	0.01702	0.01679	0.03740	0.03301	0.05755
3	0.00571	0.01564	0.00563	0.01929	0.01045	0.03697
5	0.00382	0.01704	0.00336	0.01823	0.00754	0.03727
8	0.00369	0.01712	0.00400	0.01766	0.00758	0.03726
10	0.00379	0.01684	0.00425	0.01761	0.00757	0.03726
12	0.00391	0.01696	0.00423	0.01756	0.00757	0.03726
14	0.00394	0.01691	0.00421	0.01758	0.00757	0.03726
15	0.00395	0.01691	0.00421	0.01759	0.00757	0.03726

We see that the approximation error decreases with  $r$  and plateaus at a low level at fairly small  $r$ , so that further growth of  $r$  does not improve the accuracy. As an example, at  $t = 800s$ , the error remains nearly constant with  $\|\varepsilon\|_2 \approx 0.02^\circ\text{C}$  and  $\|\varepsilon\|_\infty \approx 0.2^\circ\text{C}$  at  $r \geq 8$ . At  $t = 402s$  and  $410s$ , the saturation occurs at  $r \approx 12$ . We conclude that the ROM with  $r = 15$  is certainly sufficient for a reliably good approximation of the FOM in this case.

The temperature distributions in the cross-section of the battery pack at  $t = 3, 402$  and  $800s$  are presented in Figure 4.5 for the FOM and the ROM of order 15. Despite the fact that small difference can be observed between the ROM and FOM contours, for example, at the top of the aluminum casing of the leftmost cell and at the coolant channel near the outlet (see Figure 4.5a and 4.5b), the distributions of temperature computed by the ROM are almost identical to those by the FOM. This is also true in the distributions at  $t = 3s$ , i.e. right after the stepwise heating increase and at the later stages of the evolution characterized by strong influence of the cooling system.

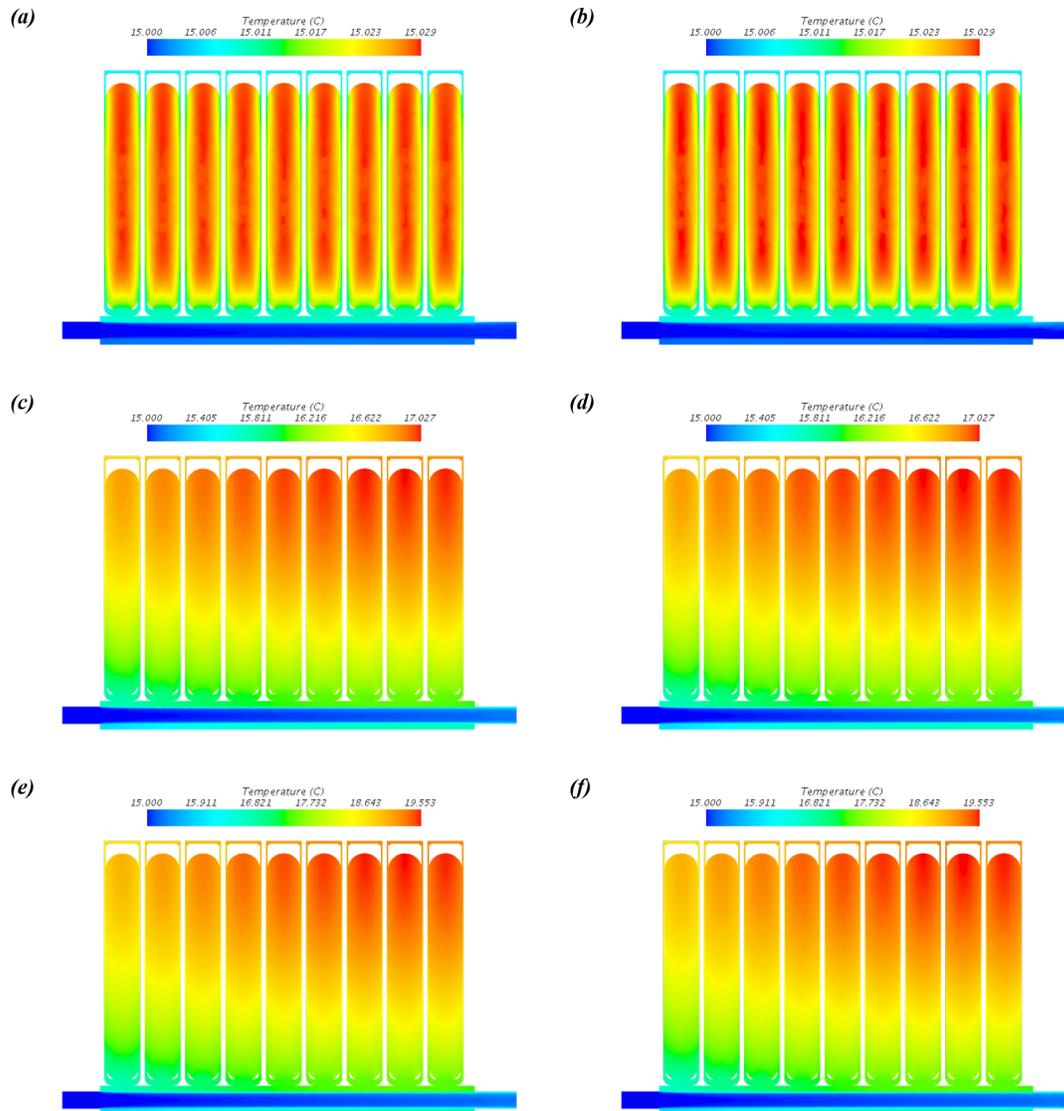


Figure 4.5: Temperature observed in the middle cross-section of battery pack for the test case 1. The heating input signal is shown in Figure 4.2. The results of ROM with  $r = 15$  and FOM at different instants are presented. (a) FOM  $t=3s$ ; (b) ROM  $t=3s$ . (c) FOM  $t=402s$ ; (d) ROM  $t=402s$ . (e) FOM  $t=800s$ ; (f) ROM  $t=800s$ .

An even greater confidence in reliability and accuracy of the ROM model is provided by the results obtained for a dynamic heat input. The transient heat input signal (see the dashed line in Figure 4.6) reflects the typical behavior of a battery pack in a driven electric vehicle.

The probed point signals of temperature presented in Figure 4.6 show excellent agreement

between the FOM and the ROM with  $r = 15$ .

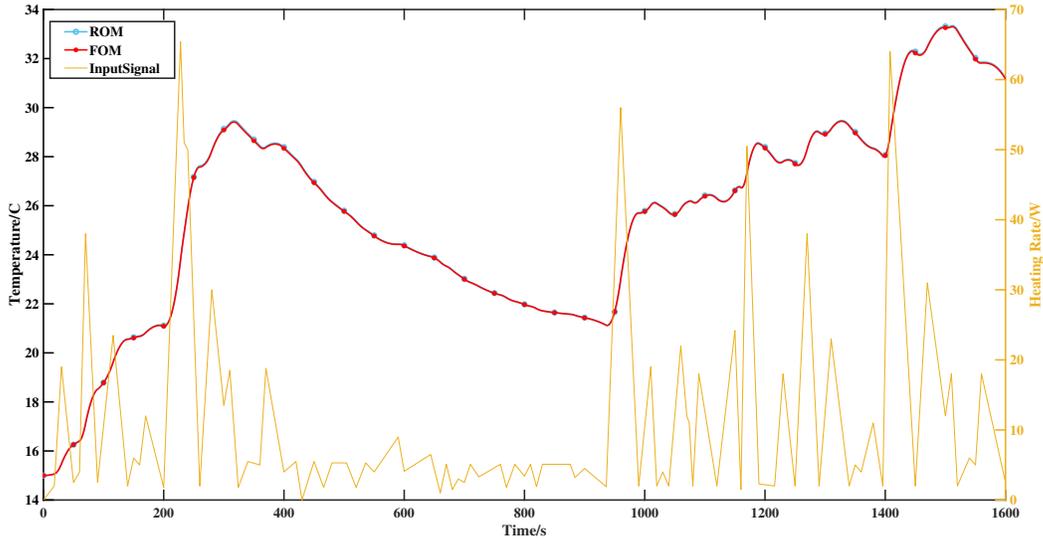


Figure 4.6: Results for the test case 2 with a realistic transient single input. The heat input signal (yellow line, right-hand side axis) and the temperature values at the probed point obtained in the FOM and the ROM with  $r = 15$  are shown.

For further verification, the norms of the error vector calculated at different time instances are shown in Table 4.3. We see that both the norms remain small. The largest error  $\|\varepsilon\|_{\infty} = 0.14175^{\circ}\text{C}$  is observed at  $t = 250\text{s}$  immediately after a sharp peak of the heating input.

Table 4.3: The norms (in  $^{\circ}\text{C}$ ) of the error at  $r = 15$  for the test case 2.

Time/s	$\ \varepsilon\ _2/^{\circ}\text{C}$	$\ \varepsilon\ _{\infty}/^{\circ}\text{C}$
250	0.01236	0.14175
800	0.01295	0.06311
950	0.03703	0.06773

We also compared ROM and FOM in the frequency domain using the Bode plot. The system's Bode diagrams for the signal at the probed point obtained by the FOM and ROMs with different values of  $r$  are shown in Figure 4.7. We see that the ROM can approximate FOM accurately in the low-frequency interval. When the frequency is increased, the accuracy of the approximation is

slightly reduced but remain acceptable. We also see that the accuracy of the ROM approximation increases with growing  $r$  and becomes certainly sufficient at  $r = 15$ .

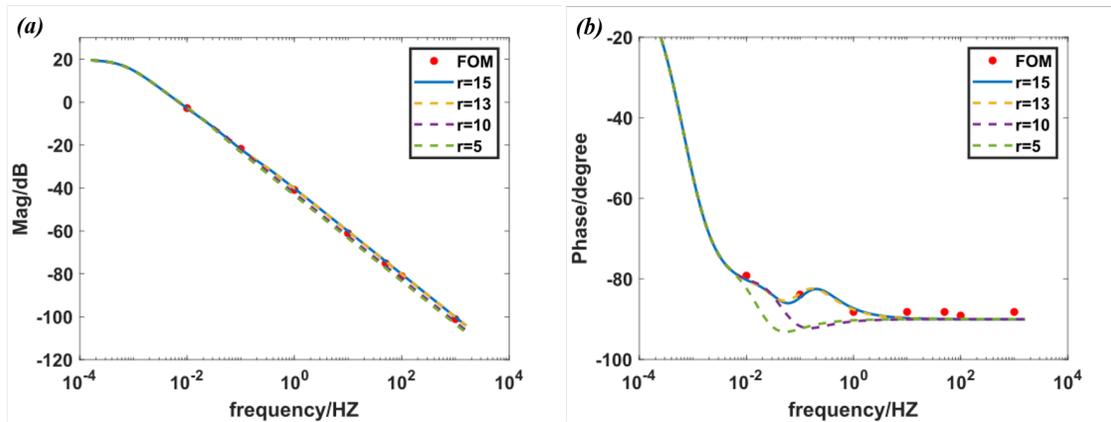


Figure 4.7: The Bode diagrams of the FOM and the ROM at different values of  $r$  for a single-input system. (a) Magnitude plot; (b) Phase plot.

The ROM dramatically saves the computation effort of modeling. The only significant computational expense is for the construction of ROM. Once this task is completed, the ROM can be used for multiple simulations of the system's evolution over long periods of time. As an example, data for the computational costs in our work are presented in Table 4.4. We see that the FOM transient simulation requires 18 hours of parallel computation on a workstation. The ROM with  $r = 15$  takes about 40 hours to develop, while the simulation of the same 1500s of evolution in the test case 1 requires only 20s.

Table 4.4: The computation cost of the ROM creation and transient simulations in the single input case compared with the cost of the FOM simulations.

	ROM	FOM
Transient simulation (1500 s)	20 s/2 cores	64509s(18 hours)/20 cores
ROM creation ( $r = 15$ )	143043 s (39.7 hours)/20 cores	

## 4.4 Multiple-Input Cases

In this chapter, we present the result of the implementation of the multiple-input algorithm to the model of a battery pack. Two multiple-input systems are considered: with two and nine independently controlled heating rates in individual battery cells. The first case is used for illustration purposes. The second case demonstrates the potential of the approach for practical applications.

ROMs of the orders up to 20, with each vector in the input matrix  $B \in \mathbb{R}^{n \times 2}$  generating up to 10 basis vectors of the Krylov subspace, was developed for the first case. No linear dependency that would trigger the deflation mechanism was detected during the model generation.

The heating scenario used to test the ROM is illustrated in Figure 4.8. In the input matrix  $B \in \mathbb{R}^{n \times 2}$ , the input signal  $b_1$  controls the signal of one battery cell at the center of the pack. The input signal  $b_2$  controls the heating of the remaining eight cells. In the scenario, all the nine battery cells generate 10 W per cell uniformly for the first 500 seconds. The signals bifurcate at  $t = 500s$  with input signal  $b_1$  jumping to 25W per cell and input signal  $b_2$  dropping to 5W per cell for the remaining time.

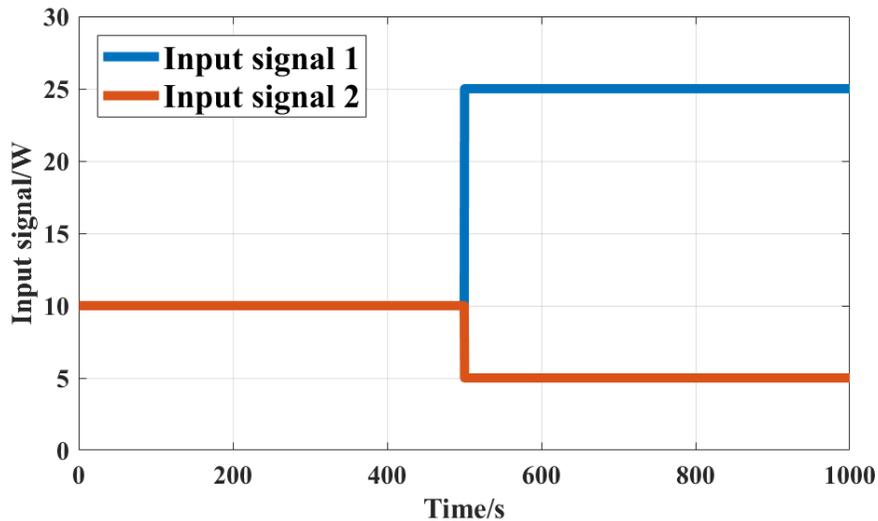


Figure 4.8: Heating rates for the test case 3 – multiple inputs with two stepwise heating rates.

The temperature distributions obtained by the FOM and the ROM with  $r = 20$  for different time moments are nearly identical (see Figure 4.9).

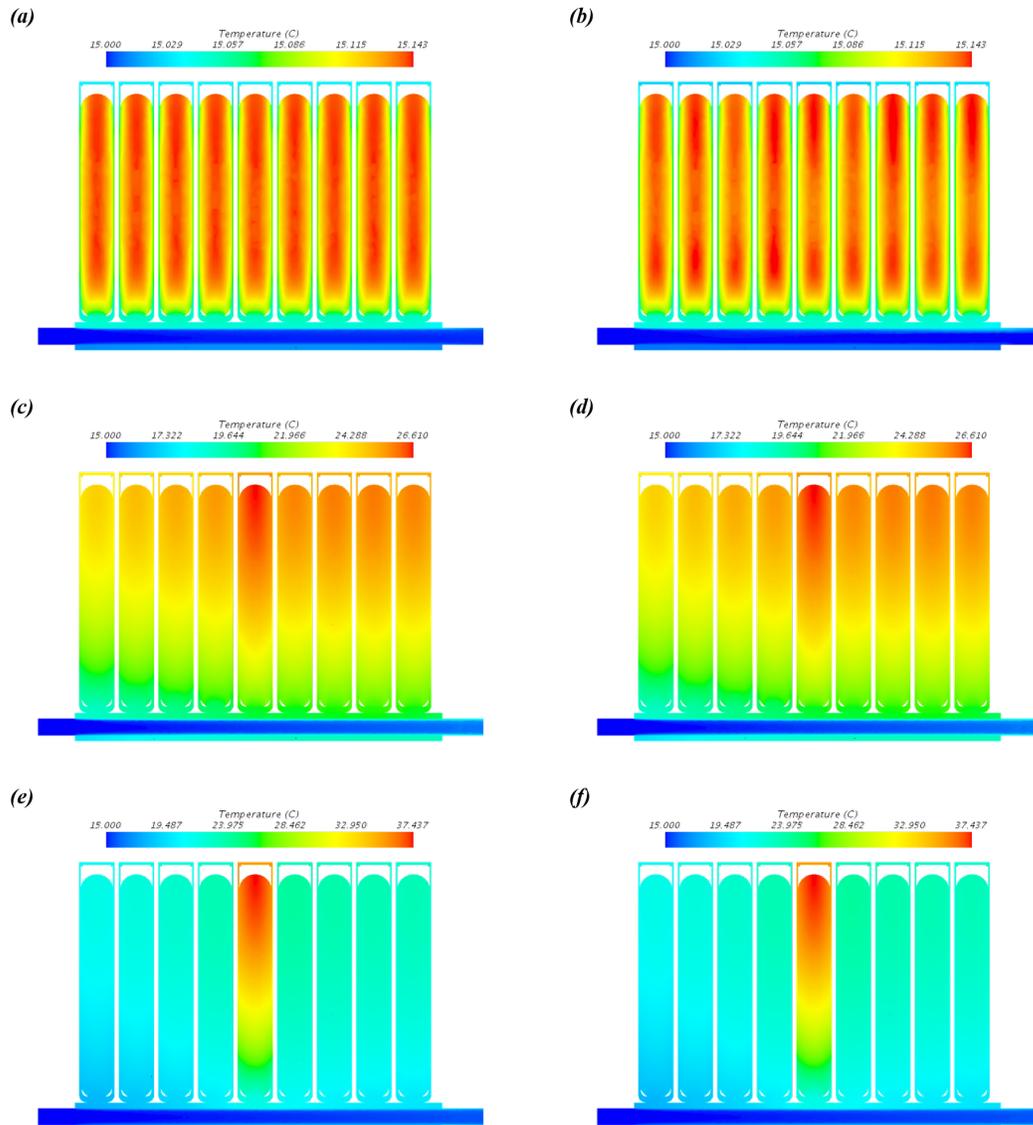


Figure 4.9: Comparison between the ROM results with  $r = 16$  and the FOM results for the test case 3 (see Figure 4.8 for input signals). Temperature distributions in the middle cross-section of battery pack are shown. (a) FOM  $t=3s$ ; (b) ROM  $t=3s$ . (c) FOM  $t=510s$ ; (d) ROM  $t=510s$ . (e) FOM  $t=1000s$ ; (f) ROM  $t=1000s$ .

The temperature signals at the probed point (see Figure 4.3) are plotted in Figure 4.10. They clearly show that the accuracy of the approximation increases with  $r$ . The ROM of the cumulative order 16 or higher accurately reproduces the dynamics of the FOM. Significant deviations between the ROM and FOM are found at lower reduced orders, such as  $r = 1, 2, 3$  and 5.

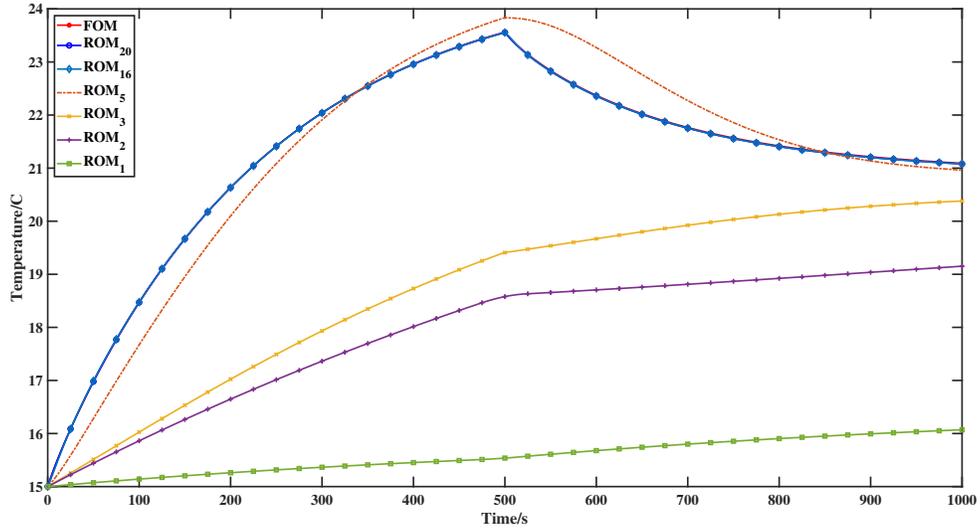


Figure 4.10: Comparison of ROM and FOM solutions for the test case 3 (see Figures 4.8 and 4.9.) Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of  $r$  are shown.

The norms of the error,  $\|\varepsilon\|_2$  and  $\|\varepsilon\|_\infty$ , are presented in Table 4.5. The ROMs of order 10 and higher appear to be sufficient for accurate representation of the FOM. However, in order to reliably reproduce the fast dynamics of the original system possibly not reflected by our test, ROM of a higher order, e.g., with  $r = 16$  can be considered for practical use.

Table 4.5: The norms of the error vector (in °C) for the test case 3 (see Figures 4.8–4.10).

r	502s		510s		1000s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
1	6.71094	10.04838	6.62344	9.91439	4.40738	6.71387
2	1.35025	1.92200	1.32306	1.95377	0.35767	1.21754
4	0.08832	0.19034	0.12001	0.30500	0.11812	0.37082
6	0.01700	0.09508	0.03376	0.16518	0.01229	0.07855
8	0.01127	0.09550	0.02395	0.11719	0.01000	0.07782
10	0.00683	0.09302	0.00618	0.09133	0.00985	0.07769
12	0.00684	0.09342	0.00576	0.09263	0.00989	0.07771
15	0.00628	0.09384	0.00522	0.09291	0.00989	0.07771
16	0.00544	0.09382	0.00535	0.09305	0.00989	0.07770
20	0.00525	0.09371	0.00531	0.09309	0.00989	0.07770

The computational cost data for the test case 3 are shown in Table 4.6. Generation of the ROM with  $r = 20$  takes 42.1 hours on 20 CPU cores. Once the ROM is generated, it takes only 26 seconds to compute a transient simulation of 1000s of evolution on a personal computer the task which would require 20.6 hours on a workstation using the FOM. The computational time for simulation is reduced by a factor of  $10^3$ .

Table 4.6: The computation cost of the ROM creation and transient simulations compared with the cost of the FOM simulation of transient process in the test case 3 lasting 1000 seconds.

	ROM	FOM
Transient simulation (1000s)	26s/2 cores	74244s(20.6 hours)/20 cores
ROM creation ( $r = 20$ )	151709s (42.1 hours)/20 cores	

In the second multiple-input case, we constructed an ROM for the realistic situation, in which all nine battery cells were controlled independently so  $B \in \mathbb{R}^{n \times 9}$ . We built ROMs of the orders  $r \leq 50$ . The deflation was not triggered during the execution of multiple-input algorithm.

The model was verified using the input signals shown in Figure 4.11. The heating rate of all the battery cells was the same 10 W, during the first 400 seconds. After that, the square wave signals changing every 50s were applied to the battery cells as indicated in Figure 4.11.

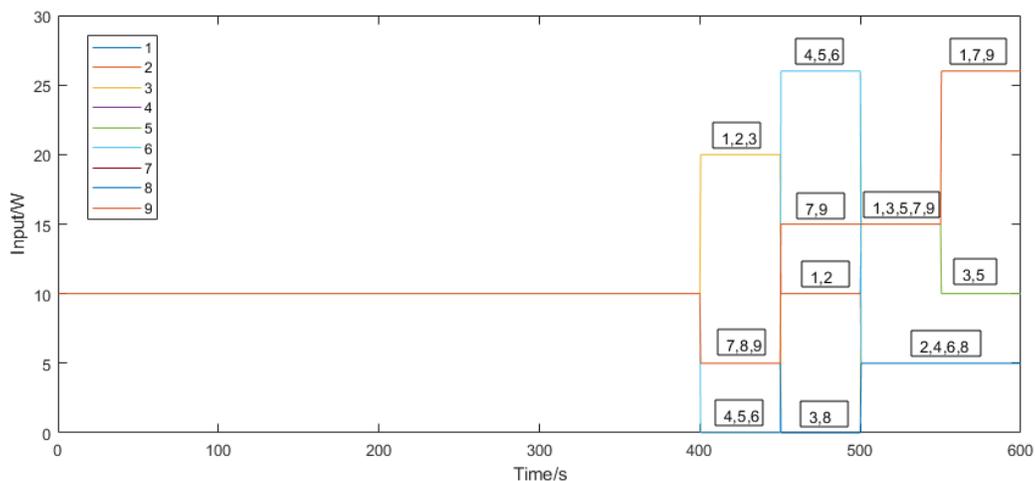


Figure 4.11: Heating rate input of the battery cells for the test case 4 – multiple inputs. The numbers indicate the heating rates applied to individual cells.

We first compare the FOM and ROM temperature distributions in the cross-section of the battery pack. We see that the ROM accurately calculates the highly variable temperature field at various stages of the evolution. As an example, the difference between the highest temperature values predicted by the two models does not exceed  $0.007^{\circ}\text{C}$  at  $t = 400s$ ,  $0.015^{\circ}\text{C}$  at  $t = 505s$ , and  $0.006^{\circ}\text{C}$  at  $t = 600s$ .

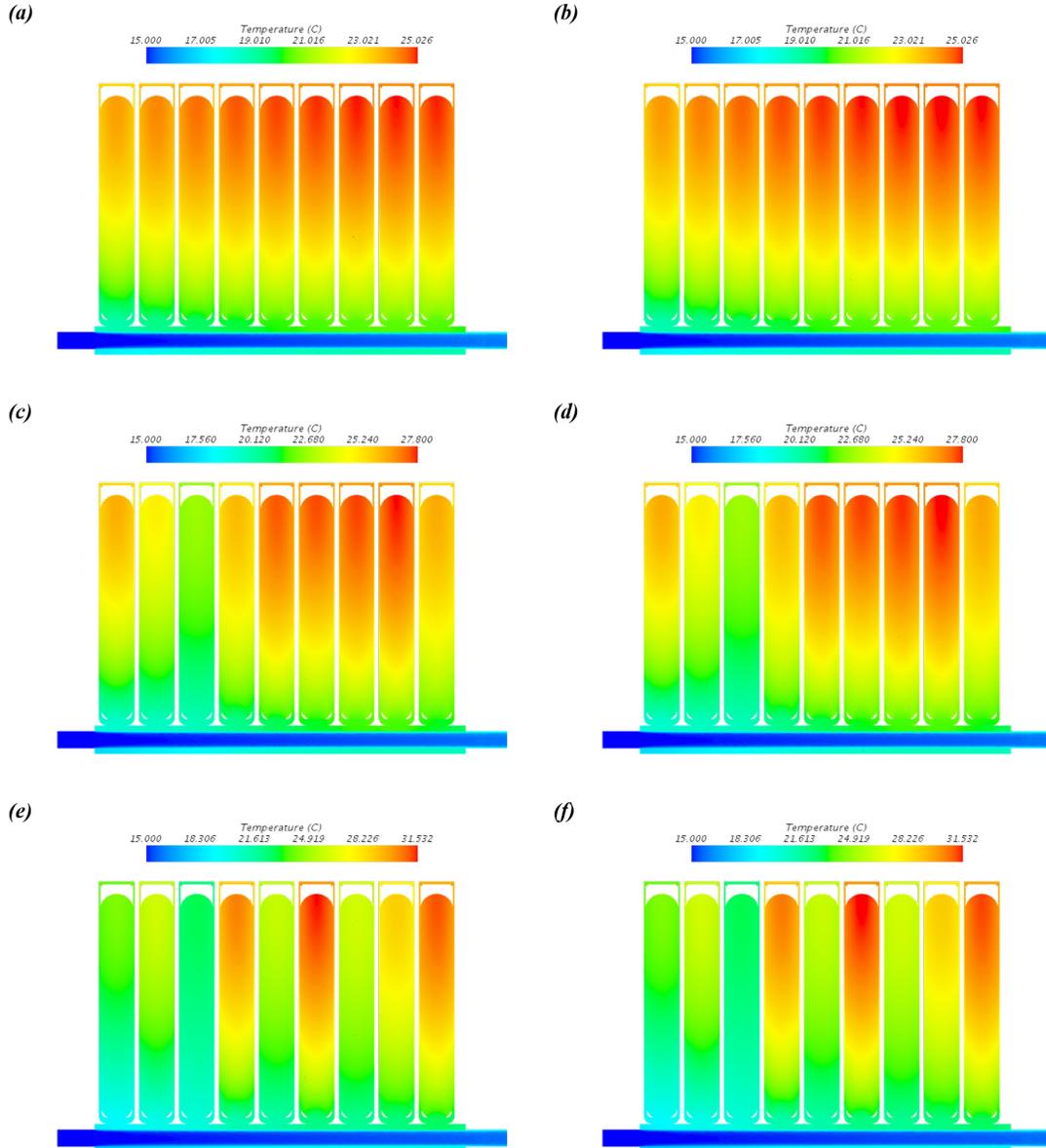


Figure 4.12: Comparison between the ROM results at  $r = 50$  and the FOM results for the test case 4 (see Figure 4.11 for input signals). Temperature distributions in the middle cross-section of the battery pack are shown. (a) FOM  $t=400s$ ; (b) ROM  $t=400s$ . (c) FOM  $t=505s$ ; (d) ROM  $t=505s$ . (e) FOM  $t=600s$ ; (f) ROM  $t=600s$ .

Further comparison is made using the temperature signal at the probed point (see Figure 4.13). The FOM data and the ROM data obtained with  $r$  in the range between 10 and 99 are shown in Figure 4.13. We see that the ROM with  $r = 10$  fails to reproduce the dynamic of the system.

Though the ROM with  $r = 20$  greatly improves the approximation accuracy, a deviation from the ROM signal can still be observed during the various stages of the transient process, e.g., at 0-100 s and 400-600 s. The ROMs with  $r = 50, 80,$  and  $99$  indistinguishably follow the FOM's transient dynamics at the probed point.

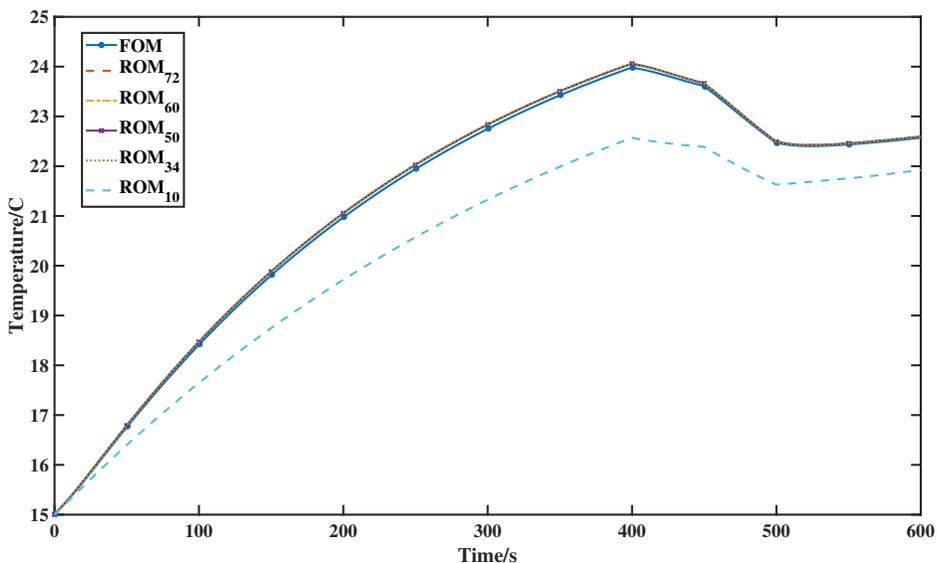


Figure 4.13: Comparison between the ROM and FOM results for the test case 4 (see Figures 4.11 and 4.12). Transient temperature signals at the probed point obtained in the FOM and the ROMs with different values of  $r$  are shown.

The norms of the error vectors obtained in the test case 4 are shown in Table 4.7. The data confirm poor accuracy of the ROM at  $r = 10$ . The accuracy, as it is reflected by the norms of the error, is good at  $r = 34$ . Further increase of  $r$  does not lead to decrease of the norms. We conclude that  $r = 34$  is sufficient in this test case, although a higher value, e.g.,  $r = 50$  may be necessary as a safety measure to accurately approximate the features of the stack behavior not reflected in the test.

The construction of the ROM with  $r = 50$  takes around 150 hours of the 20-CPU-cores parallel computation. The computational time required to complete the ROM simulation of transient behavior, such as that in the test case 4, remain exceptionally small, as in the previous cases considered in this paper.

Table 4.7: The norms (in °C) of the error vector for the test case 4 (see Figures 4.11, 4.12 and 4.13).

r	455s		505s		600s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
10	0.93124	1.79201	1.07766	1.98469	0.99904	2.25001
34	0.05344	0.12464	0.06379	0.12266	0.06040	0.14394
50	0.05382	0.12344	0.06181	0.11407	0.06019	0.14342
60	0.05384	0.12458	0.06177	0.11302	0.06019	0.14376
72	0.05444	0.11962	0.06171	0.11485	0.06017	0.14371

## 4.5 Boundary Condition as Input

In this section, we present the result of the implementation of the boundary condition as an input. The Dirichlet boundary condition of the ODE system, i.e., the inlet temperature of the coolant channel in the battery pack, is considered. It is controlled to be one of the time-varying variables of  $u(t)$  in (2.4).

The heat source vector  $B = \gamma$  is generated such that the internal heating represents the Dirichlet boundary condition within one extruded cell at the coolant inlet (as shown in Figure 4.14). The implementation of the method does not follow the formula (2.9) directly. The reason for that is the possibility of deviation of the internal STAR-CCM+ solver from the FV formulas (2.7) and (2.8). The implementation is based on the linearity of the problem and follows the following approach. We solve the steady-state heat transfer problem with  $T_{BC} = 0$  and some volumetric heating density  $F$  in the extruded cells, e.g.,  $10^9 W/m^3$ . The solution vector is a uniform field with a temperature equal to  $d$  Celsius degrees. By linearity, the volumetric heating of extruded cells with density  $F/d$  is, then, identical in its effect on the system to the inlet temperature  $T_{BC}$  of 1 Celsius degree. Vector  $B$ , which has elements equal to  $F/d$  in the extruded cells and zero elsewhere, can be considered as an input vector in the state-space representations used to develop ROM, i.e., from (2.4).

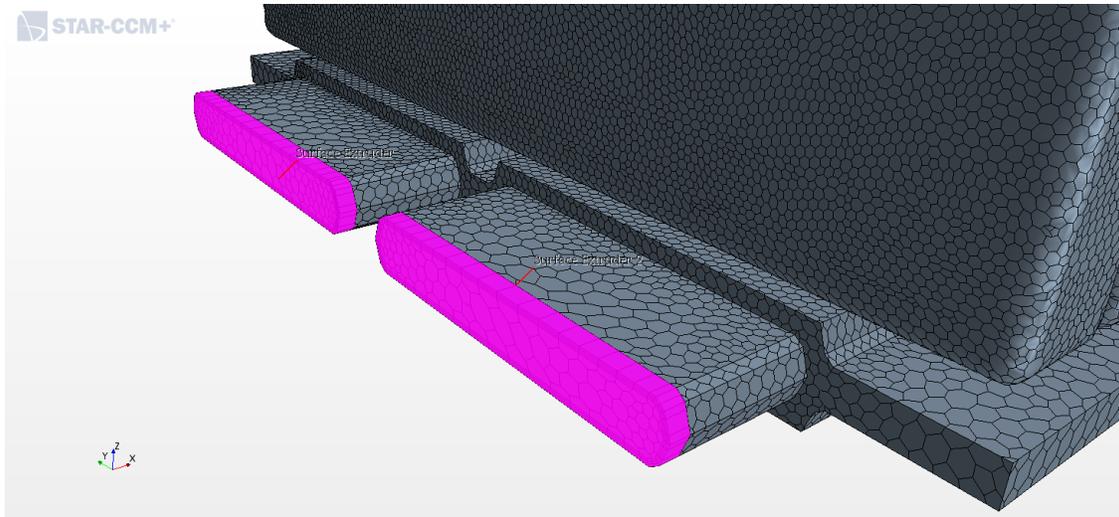


Figure 4.14: Extruded layer method for BC as input. The extruded layer is marked in pink.

We present a ROM with the boundary condition implemented as an input in a single-input system. Note that the boundary condition as input can always be implemented for the multiple-input systems for practical demand.

To explore the performance of the boundary condition as an input method, we present an ROM of order 20 using single-input algorithm 1 for the validation. It is validated using a transient inlet temperature as the boundary condition. The inlet temperature starts at temperature of 25 °C for 100 seconds and step-changes to 5 °C until 200 seconds (as shown in Figure 4.15). The initial temperature distribution is uniform at  $T = 15^{\circ}\text{C}$ .

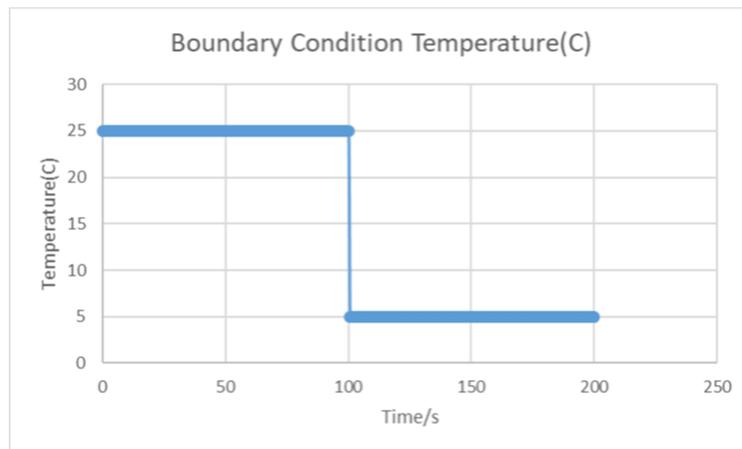


Figure 4.15: Inlet temperature signal for boundary condition as input validation.

The temperature signals at the probed point (see Figure 4.3) are plotted in Figure 4.16. They clearly show that the accuracy of the approximation using the boundary condition as input. The ROM of order 20 reproduces the dynamics of the FOM. Derivation takes place after the step-change of the signal, i.e.,  $t = 102s$ . Other than that, the approximation achieved by the boundary condition as the input indicates a great accuracy. To be more clear about the approximation of the method, the norm of the error,  $\|\varepsilon\|_2$  and  $\|\varepsilon\|_\infty$ , are presented shortly in Table 4.8.

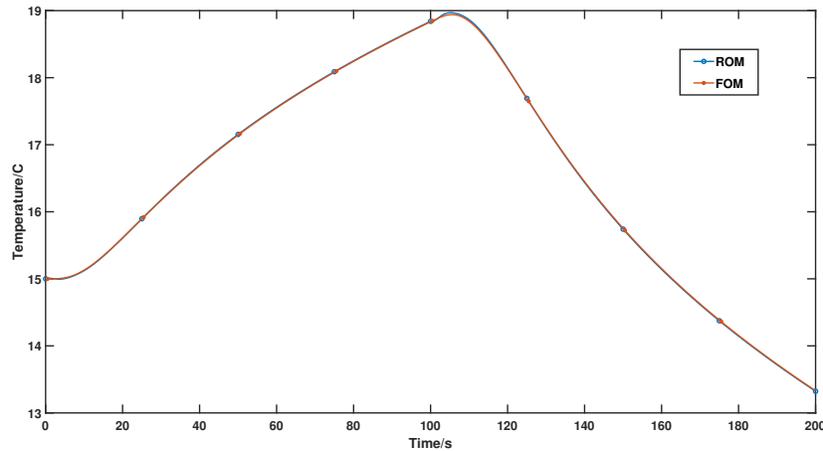


Figure 4.16: Comparison of ROM and FOM solutions for the boundary condition as input. Transient temperature signals at the probed point obtained in the FOM and the ROM of order 20 are shown.

Compared with the results shown in previous sections for single-input or multiple-input cases, the norms of the error is rather large with respect to the approximation to the system’s dynamics. We expect the inaccuracy come from accumulated error by imposing the volumetric heat source to the model. The artificially defined ‘zero’ boundary temperature yield very small error near the inlet. This error can potentially accumulated as during the ROM iterations. However, the approximation level is still a relatively good approximation to the FOM with norm of error equals to  $0.373319^{\circ}\text{C}$  while the maximum error among the whole temperature does not exceed to  $0.846873^{\circ}\text{C}$ . As the transient process goes further, the norms of the error vector restores to excellent similarity to the FOM. We expect the boundary condition as the input method to have overall excellent performance in applying the reduced-order modeling.

Table 4.8: The norms of the error vector (in °C) for the boundary condition as the input (see Figures 4.16).

r	102s		110s		200s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
20	0.373319	0.846873	0.060261	0.202178	0.007995	0.089852

## 4.6 Parametric Reduced-Order Modeling

In this section, we present the implementation of the parametric reduced-order modeling. As illustrated in section 2.2.2, given several local reduced-order models, a parametric reduced-order model can be calculated by interpolating the system matrices. The new ROM obtained in such way can be used to predict a dynamic behavior of the system within a certain range of parameters, e.g., mass flow rates for conjugate heat transfer or flow rate in an inlet of a ventilation system for airborne transmission. We illustrate the results of the parametric ROM for the conjugate heat transfer of a battery pack.

We firstly generate three local reduced-order models at different mass flow rates (0.01, 0.03, 0.05 kg/s). With the pre-computed local ROMs, the transformation matrix  $R$  is computed by singular value decomposition. Then the local ROMs are transferred into a common reduced subspace for the interpolation. The parametric ROM is created according to the **Algorithm 4**.

The parametric ROM valid for the mass flow rate  $p$  in the range from 0.01 to 0.05 kg/s is constructed by the means of quadratic interpolation. The quadratic interpolation weighted factors are solved based on  $\sum_{i=1}^k \varpi_i(p) = 1$  and three local quadratic points, i.e.,  $\varpi_1(0.01) = 1$ ,  $\varpi_2(0.03) = 1$  and  $\varpi_3(0.05) = 1$ . The weighted coefficients  $\varpi_i$  are:

$$\begin{bmatrix} \varpi_1(p) \\ \varpi_2(p) \\ \varpi_3(p) \end{bmatrix} = \begin{bmatrix} 1250 & -100 & 15/8 \\ -2500 & 150 & -5/4 \\ 1250 & -50 & 3/8 \end{bmatrix} \begin{bmatrix} p^2 \\ p \\ 1 \end{bmatrix} \quad (4.4)$$

where  $p$  is the mass flow rate for interpolation.

The model is tested for two mass flow rates  $p = 0.02, 0.04 \text{ kg/s}$ . In the test case, each battery cell produces  $10 \text{ W}$  constantly for  $400$  seconds. The local ROMs of order  $15$  found in section 4.3 to provide accurate results for the single-input cases are used.

The results obtained at  $p = 0.02 \text{ kg/s}$  are shown in Figure 4.17 and 4.18. Similar results are observed at the mass flow rate of  $0.04 \text{ kg/s}$ . We see that the parametric ROM accurately predicts the temperature field at various stage of the evolution.

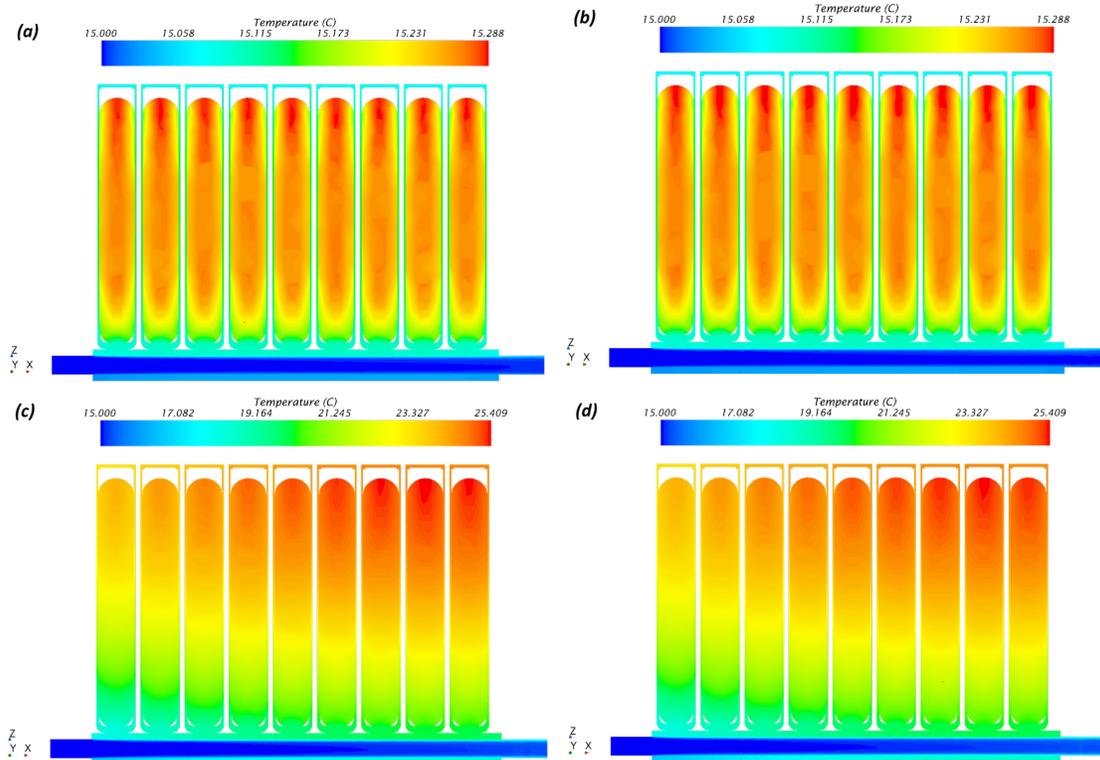


Figure 4.17: Comparison between the parametric ROM results at  $\text{MFR} = 0.02 \text{ kg/s}$  and the FOM results. Temperature distributions in the middle cross-section of battery pack are shown. (a) FOM  $t=5\text{s}$ ; (b) parametric ROM  $t=5\text{s}$ . (c) FOM  $t=400\text{s}$ ; (d) parametric ROM  $t=400\text{s}$ .

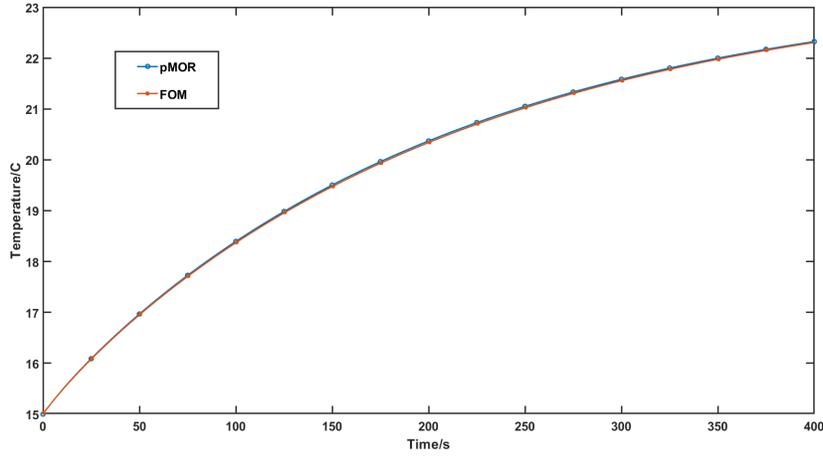


Figure 4.18: Comparison of parametric ROM and FOM solutions at mass flow rate =  $0.02\text{kg/s}$ . Transient temperature signals at the probed point obtained in the FOM and the ROM of order 15 are shown.

To further examine the accuracy of the parametric ROM, the norms of the error are calculated (see Table 4.9). At  $t = 5\text{s}$ , the maximum difference between parametric ROM and the FOM  $\|\varepsilon\|_\infty$  is  $0.0146^\circ\text{C}$  while the rms difference  $\|\varepsilon\|_2$  is  $0.002^\circ\text{C}$ . The norms of the error between parametric ROM and FOM indicate the excellent approximation has been achieved using the parametric ROM.

Table 4.9: The norms of the error vector (in  $^\circ\text{C}$ ) for parametric ROM at mass flow rate =  $0.02\text{kg/s}$  (see Figures 4.17 and 4.18).

r	5s		400s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
15	0.002	0.0146	0.0314	0.1233

## 4.7 Systems With Time-Varying Parameters

The implementation of parametric reduced-order modeling for systems with time-varying parameters, i.e., time-varying mass flow rates at the inlet, is presented in this section. The applicability of the reduced-order modeling is extended into broader setting. We drop the assumption

of a steady-state flow velocity field. The system with time-varying parameters is described using decoupled momentum equation and heat equation. The energy equation is now a linear equation with time-varying three-dimensional flow velocity field  $\mathbf{U}(\mathbf{x}, t)$  and constant flow properties. The system is solved in two steps: (i) solve the momentum equation to obtain the flow velocity field  $\mathbf{U}$ ; (ii) substitute  $\mathbf{U}$  into the heat equation and solve it.

Instead of the previously considered ROM with constant system matrix  $A$ , the ROM with time-varying parameters has a time-dependent system matrix  $A$  and input matrix  $B$  as shown in (2.19). It contains the discretization coefficients matrix of the ODEs system which express the diffusive and adjective terms at given time-varying parameter. Until then, the time-varying system matrix  $A(t)$  is determined using interpolation.

We simulate the battery pack with time-varying MFRs and constant heat input by parametric reduced-order modeling. The parametric ROM is essentially an ODE system. It is found by interpolating the system matrices of the local reduced-order models obtained at specific parameters. The approach is tested at the mass flow rate  $p(t)$ :

$$p(t) = 0.03 + 0.01 * (\sin(2 * \pi * t/200) + \sin(2 * \pi * t/100))(kg/s). \quad (4.5)$$

The ODEs are interpolated at every time step based on the instantaneous mass flow rate  $p(t)$  (see Figure 4.19). The weighted coefficients for the quadratic interpolation are computed as in (4.4). Then, the system matrix  $A(t)$  and input matrix  $B(t)$  are interpolated as a pROM of parameter  $p$  using (2.16). In this way, the FOM that computes partial differential equations (PDE) with decoupled momentum and heat equations are simplified to an ODE system.

The results are compared with FOM for validation. In the pROM, we use three local ROMs with  $r = 15$  at MFR= 0.01, 0.03 and 0.05kg/s. The temperature at the inlet and initial condition at the domain are 15°C. We apply a constant heat input of 10 W for all battery cells. While in the FOM, we choose implicit unsteady solver with segregated flow. At each time step, the momentum and energy equations are solved with 400 inner iterations. The residual of energy reaches below  $10^{-7}$ ; the residual of momentum and continuity reaches below  $10^{-14}$ .

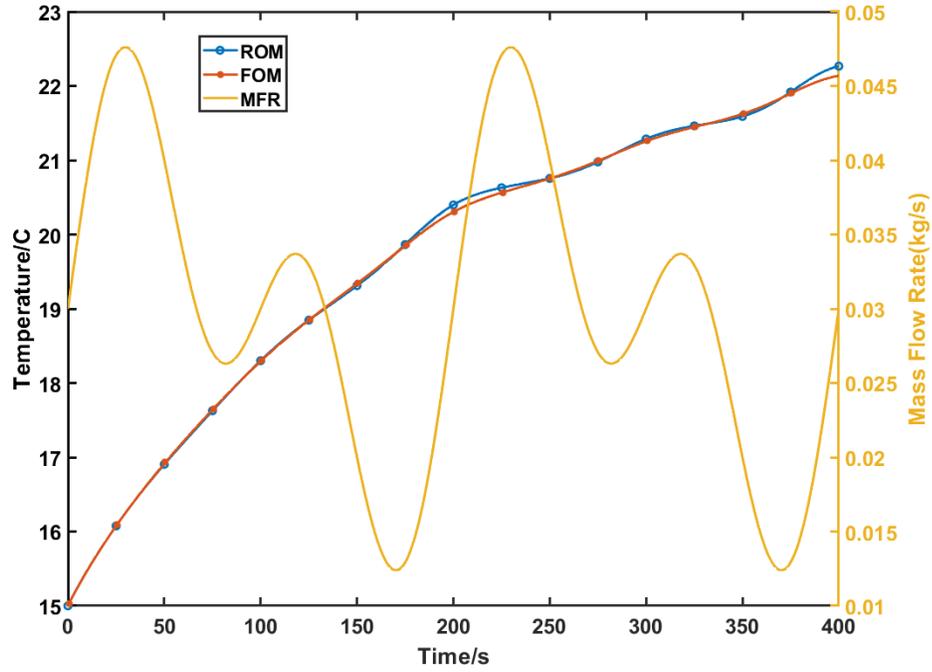


Figure 4.19: Comparison of parametric ROM and FOM solutions under transient mass flow rates. Temperature signals at the probed point obtained in the FOM and the parametric ROM are shown.

The results show a clear deviation between FOM and ROM when the change of the MFR is rapid. The  $\|\varepsilon\|_2$  and  $\|\varepsilon\|_\infty$  norms evaluated at different time instants are shown in Table 4.10. It is observed that the parametric ROM fails to capture the dynamics of the original FOM when the MFR undergoes rapid changes at  $t=200$  and  $t=400$ s.

Table 4.10: The norms of the error vector (in °C) for parametric ROM at time-varying mass flow rates (see Figures 4.19).

Time/s	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
200	0.038173	0.177921
250	0.022414	0.096755
300	0.018023	0.046944
350	0.029274	0.115097
400	0.052249	0.244504

The results show us a clear limitation of the current parametric ROM technique. The limitation comes from the description of system matrix  $A(t)$  in (2.19). The parametric ROM uses interpolation between approximations obtained for the system matrices based on steady-state velocity field. However, the velocity field, by nature, is unsteady at transient MFR signal. The parametric ROM fails to represent the resulting dynamics of heat transfer.

FOM simulation performed with small time step  $\Delta t = 10^{-4}s$  clearly illustrate the difference between the unsteady and steady-state velocity fields in the cooling channel. Specifically, the ‘reverse flow’ near the wall is clearly observed once the MFR is changed (see Figure 4.20).

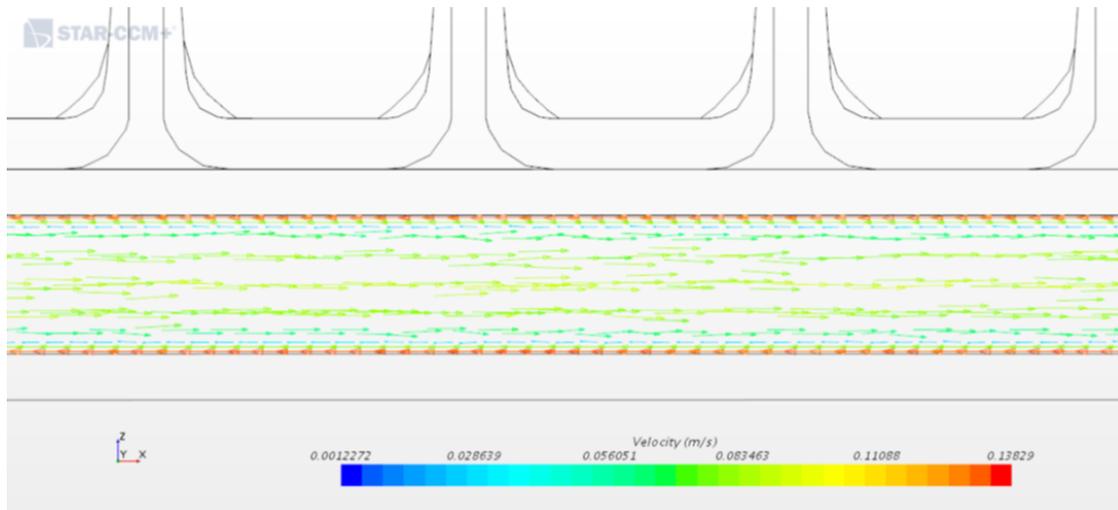


Figure 4.20: Reverse flow developing in the cooling channel with time-dependent mass flow rate

The interpolated parametric ROM cannot take this effect into account. The use of a fully developed flow field by the interpolation leads to ‘over-reaction’ of the heat transfer behavior of the system. Therefore, it is not surprising that the time-varying MFRs by the current parametric ROM technique based on interpolation does not show the results as accurate as in the previous sections.

## 4.8 Conclusion

A method for development of reduced order models (ROMs) of conjugate heat transfer is presented. The method is based on the Krylov subspace approach and the Arnoldi algorithm, and can

be applied to single input and multiple input systems. The method utilizes a full order (FOM) CFD solution of a steady-state problem, but does not require access to the discretization matrix. This makes the method especially practical for industrial applications, where commercial CFD software is used and the discretization matrix is usually inaccessible.

As a demonstration and validation study, the method is applied to conjugate heat transfer in a simplified model of a typical automotive electric battery pack. The ROMs of various orders are derived and applied to reproduce transient behaviors in four distinct scenarios characterized by strong temporal variations and spatial gradients of temperature. The results of the FOM CFD simulations of the same scenarios based on the computational grid of more than  $10^6$  finite-volume cells are used for validation. The analysis shows that ROMs of small order ( $r$  between 5 and 50 depending on the number of independent inputs and the type of transient behavior) accurately reproduce the evolution of the entire temperature field. It is noted that the physical parameters in the FOM will still need to be validated by experiments to guarantee the accuracy of the ROM when compared with the actual system.

Significant, but feasible amount of computations is required to derive an ROM for a given heat transfer system. Once the ROM is developed, the computational costs of its application to prediction of transient behaviors is exceptionally low, three or more orders of magnitude lower than the time required for achieving the same goal using a full CFD solution. This opens the opportunity of using the ROMs for real-time on-board control of battery thermal management systems and for the design optimization studies requiring multiple simulations. Another possible extension of the developed ROM for heat transfer is to couple it with an electrothermal model of battery cells for real-time simulation of both electric and thermal behaviors of a battery pack including its thermal management system. In such a coupled simulation, the electrothermal model of battery cells will calculate the heat generation based on current input and the predicted cell temperature from the ROM. The ROM, on the other hand, will take the predicted heat generation from the electrothermal cell model as input.

An attempt to extend the method to the case of time-varying system parameters using matrix

interpolation has been also made. The reduced-order modeling approach based on the framework of pROM is proposed for systems with time-varying parameters. The original decoupled PDE system is approximated by an ODE system using matrix interpolation. The resulting ROM is shown to capture the dynamics of the system. However, the quality of the results is, admittedly, lower than in the case of time-independent parameters. This is attributed to the effect of not fully developed fluid velocity under time-varying mass flow rates. To the best of our knowledge, the choice of weighted function for pROM or adapting the reduced-order modeling technique that can potentially improve the quality of the results. This is left for future investigations.

## CHAPTER 5

# Reduced-Order Modeling for Airborne Transmission of Respiratory Infections

This Chapter presents the reduced-order modeling of airborne transmission of respiratory infections in indoor environments with forced convection. Firstly, a computational experiment comparing the results of the RANS-based Eulerian approach with the RANS-based Lagrangian approach of [1] is presented in section 5.1. Then, the development of ROM for ventilated indoor environment is presented in section 5.2. The optimized ROM is applied to assess the risk of infection as presented in section 5.3.

### 5.1 Comparison Between Eulerian Approach and Lagrangian Approach Applied to Aerosol Transport

We investigate aerosol transport to predict airborne transmission in indoor ventilation systems. The results are compared with previous works analyzing the airborne transmission with the Lagrangian method [1]. The goal is to confirm the reliability of our approach.

The geometry of the classroom model, as well as the iso-contour at time instance of 100s are shown in Fig 5.1. The classroom model in the numerical study of aerosol transport [1] is applied in the current work. A typical medium-size classroom is represented by a room of 9m x 9m x 3m. Desks with simplified human bodies are located as shown in Fig. 5.1. A total of 9 students and one instructor are located in each of their sites in the face of a lecture on the stage. Each simplified geometry of the student has a rectangular mouth surface ( $0.06 \times 0.03 \text{ m}^2$ ) where the inlet boundary condition is applied.

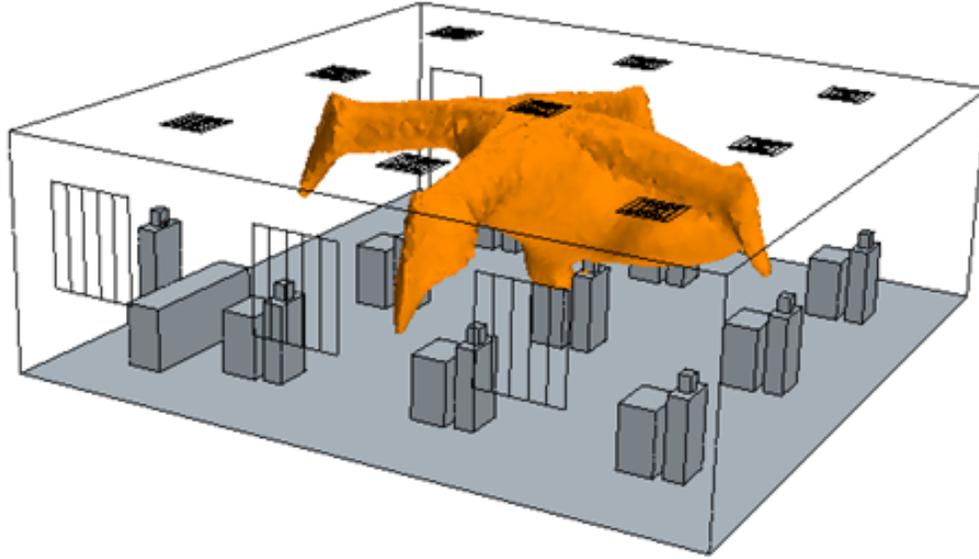


Figure 5.1: Model geometry and iso-surface of infectivity field  $C=10$  at  $t=100s$ .

The ventilation system follows the standard of ASHRAE 62.1 for recommended indoor air quality. In the ventilation system, a total of 5 air supply diffusers and 4 air recycling diffusers are settled on the top of classroom. The Cubic Feet per Minute (CFM) is approximated to be  $\sim 1230$  for adequate ventilation according to the recommended configuration [1]. The inlet boundary condition for air supply diffusers follows the settings in [1] in order to have an identical base indoor ventilation flow field. Specifically, the airflow injects with a vertical air velocity of  $0.395m/s$  at an angle of  $37^\circ$  from the horizontal surface, and the diffuser area gives  $0.294m^2$ . The model mesh contains a total of 3,041,512 cells with a minimum cell size of 0.5 cm and maximum cell size of 10 cm with gradual transition, maximum skewness of 0.823 (a mean value of 0.593), and maximum aspect ratio of 3.21 (a mean value of 1.43) as presented in Fig. 5.2 [1].

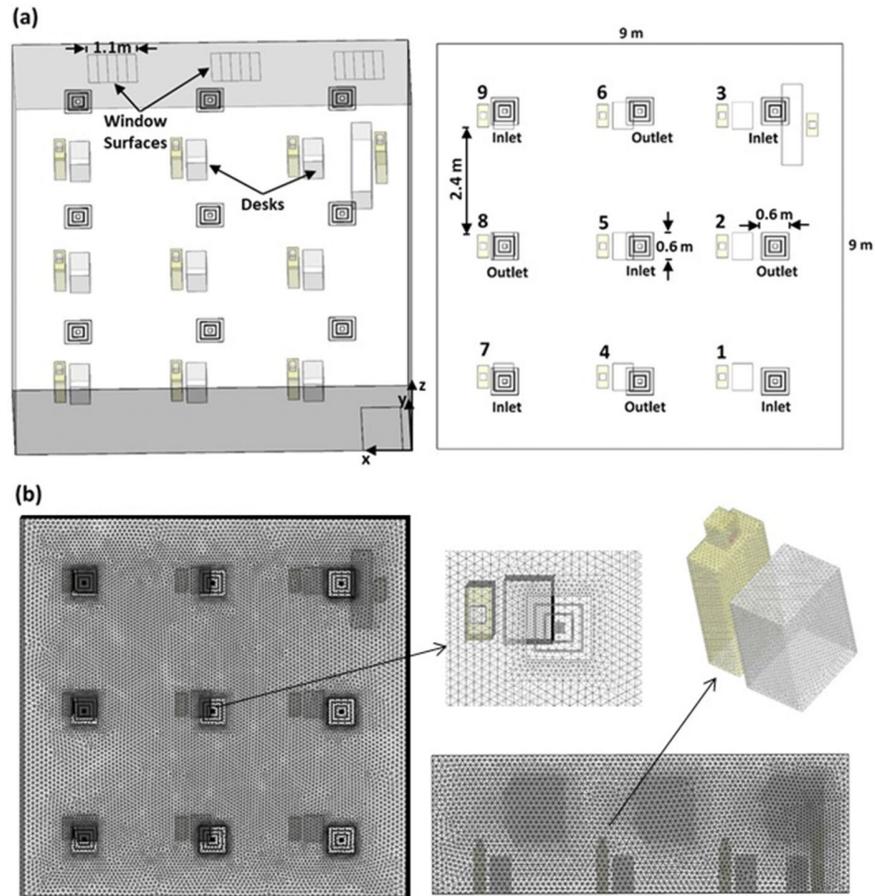


Figure 5.2: Illustration of (a) the classroom model and (b) the computational mesh used in the CFD simulations. The image from [1] is represented with permission of AIP Publishing.

The indoor ventilation flow field is solved from continuity and momentum equations in the beginning using Reynolds Averaged Navier-Stokes (RANS) incompressible solver of ANSYS FLUENT. The Re-Normalization Group (RNG)  $k - \epsilon$  model is used to simulate the turbulence flow. The SIMPLE algorithm is used in computing the flow fields. First order scheme is employed for pressure interpolation while the convection and viscous terms are discretized using a second-order scheme. The computed airflow fields (shown in Fig. 5.3) are frozen once the solution is converged.

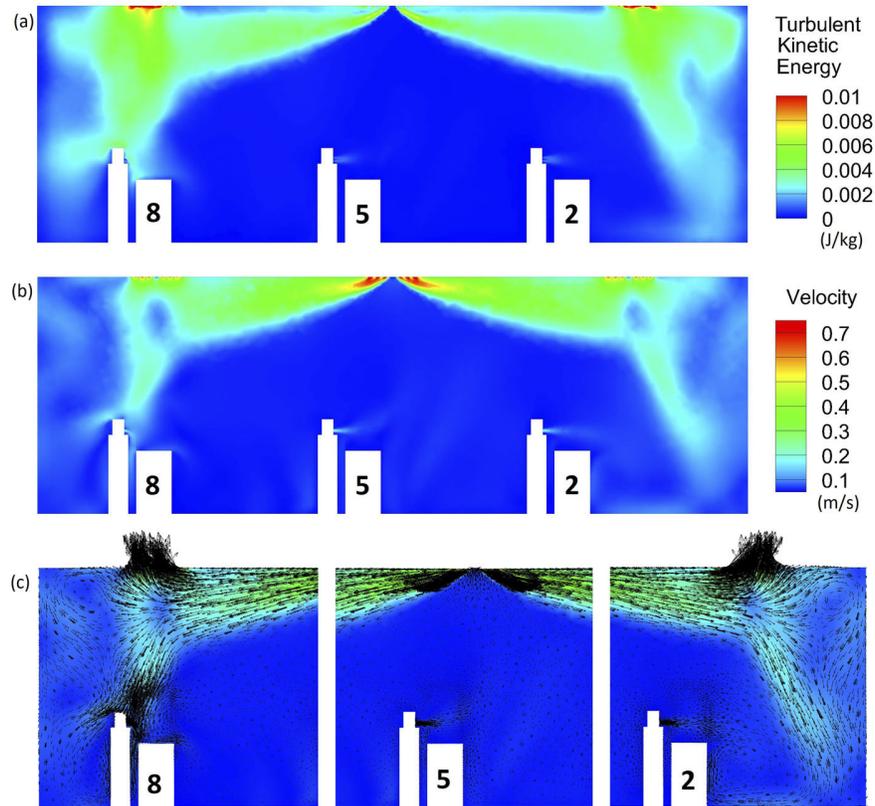


Figure 5.3: (a) Turbulent kinetic energy, (b) velocity magnitude distribution, and (c) velocity vectors across a slice going through students 2, 5, and 8. The imagine from [1] is represented with permission of AIP Publishing.

The temporal transport behavior of the infectivity field is modeled in an Eulerian framework. The second order convection scheme is used to converge the transport equation 2.3 for the infectivity field. The Newman boundary conditions (zero flux for  $C$ ) are applied to all walls, surfaces of tables, and humans. Constant zero values of  $C$  is applied at the inlets.

We study the scenario by assuming student 5 in the center of the classroom carried the infectious aerosol by exhaling activities such as coughing, and laughing. The transmission mode with the base of steady-state flow fields is studied numerically through the Eulerian approach. In this way, the airborne transmission can be assessed using the CFD solver. We first compare the prediction of aerosol transmission between the Eulerian and Lagrangian approaches.

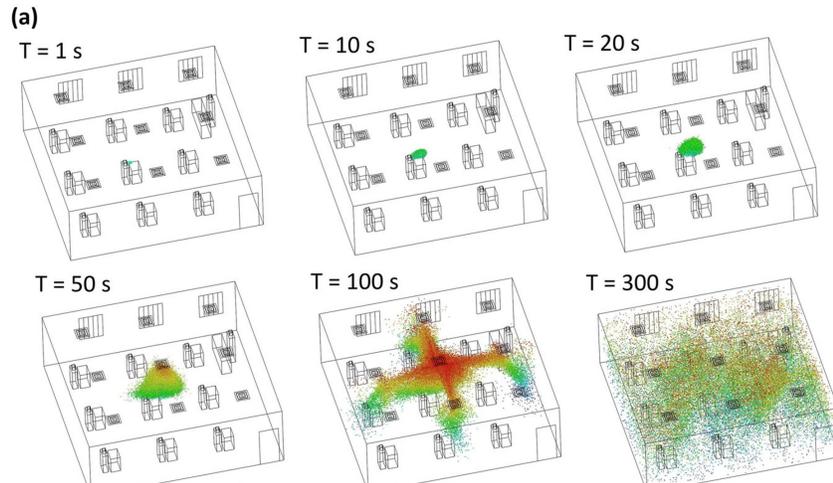


Figure 5.4: Distribution of  $1 \mu\text{m}$  aerosol particles in the classroom at different points in time for the (a) student 5 source. The imagine from [1] is represented with permission of AIP Publishing.

It has been observed from [1] in simulating the aerosol transport from initial diffusion of the aerosol particles ( $t=20\text{s}$ ) to the convected transport of aerosol particles ( $t=100\text{s}$ ) (see Fig. 5.4). An exhaling process is modeled using a single-release impulse source from student 5. The distribution of the released particles are modeled under the framework of the Lagrangian approach. The result in [1] predicts the spatial distribution of the aerosol particle of  $1 \mu\text{m}$  followed by the initial one-single impulse.

For a comparative study, a transient signal for the transport scalar source imitating the Lagrangian particle seeding in [1] is used. The signal for the source intensity is shown in Figure 5.5.

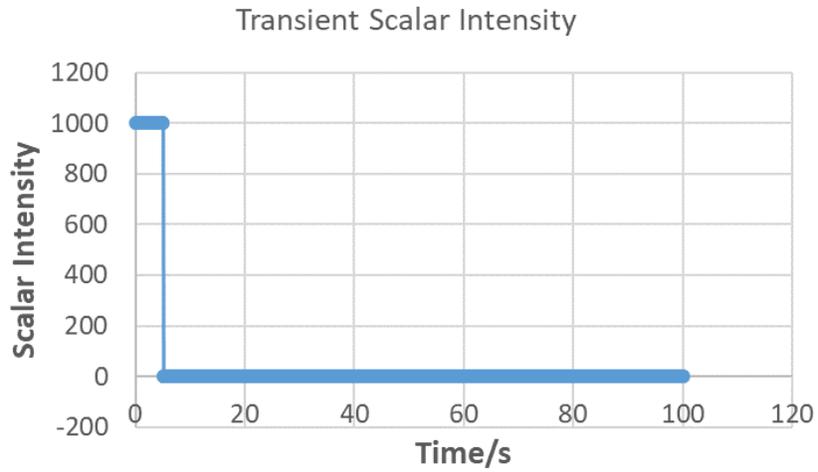


Figure 5.5: Transient Scalar Intensity signal of the aerosol cloud.

The source initiates a cloud of aerosol with the infectious virus in a cubic box in front of student 5 (See Fig 5.6) with an intensity of 1000 for the first 5 seconds (as illustrated in Figure 5.6).

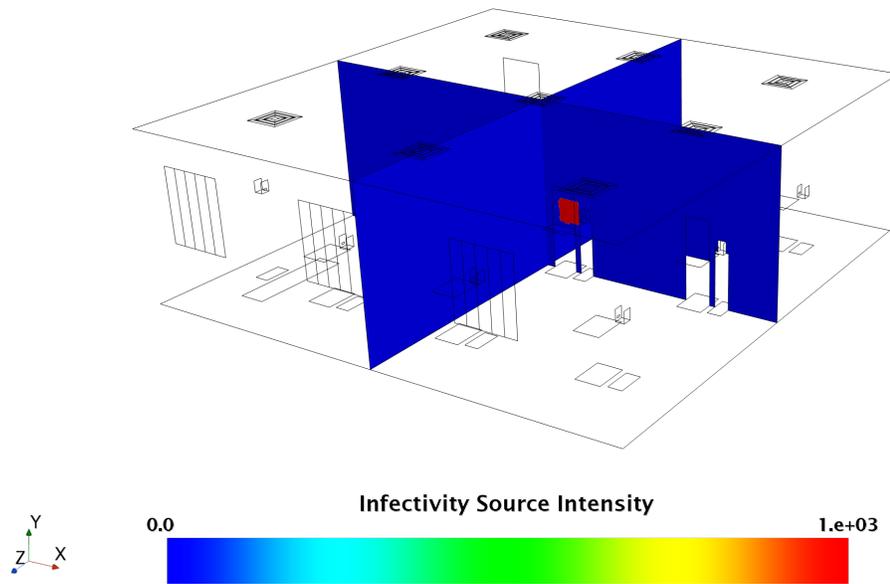


Figure 5.6: Initial scalar intensity source cloud (Source is marked in red box with  $0.38 \times 0.4 \text{ m}^2$  in area and 0.3m in height).

The results obtained using the Eulerian methods are presented in Figures 5.7, 5.8.

At 20 seconds, the aerosol scalar is still in the initial diffusion stage (See Figure 5.7). The area outside the aerosol cloud for the scalar source remains uniform 0. In the area of the scalar source, the aerosol source gradually spreads around due to the concentration gradient of the aerosol scalar. However, it can be observed that aerosol scalar transport is dominated by diffusion due to the low degree of airflow in the space where student 5 is located. The results predicted using the Eulerian approach (see Figure 5.7) are consistent with the results obtained by the Lagrangian approach.

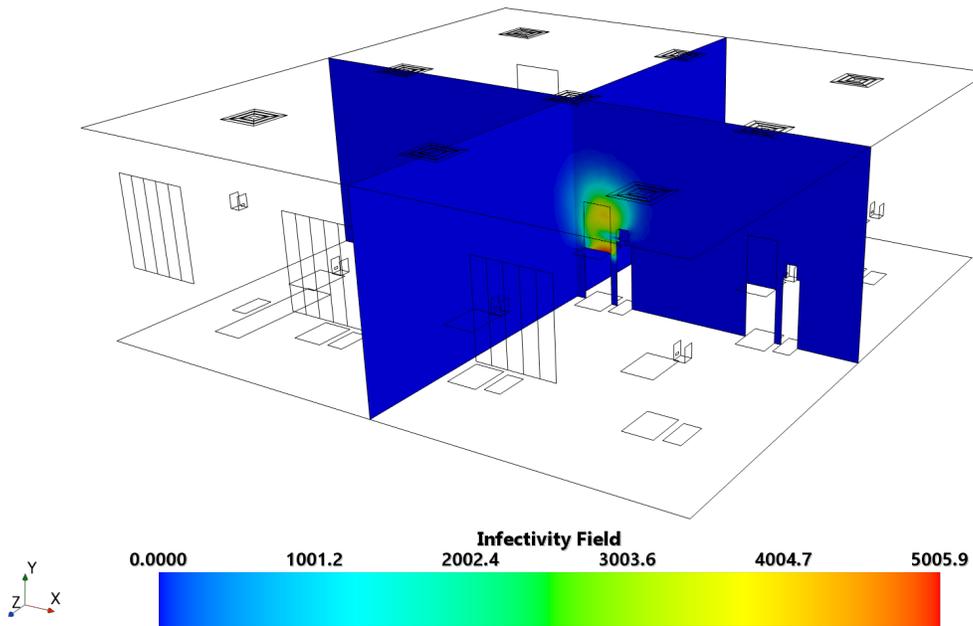


Figure 5.7: The spatial distribution of infectivity field C at  $t=20s$ .

From 50 to 100 seconds of the transient process using the Eulerian approach, the propagation mode of aerosol gradually changes to the co-leading propagation of diffusion and forced convection modes. As shown in Figure 5.8, the scalar is transported more efficiently with turbulent fluid motions. With the transport of scalar intensity, the most intense part of the aerosol field gradually shifts to the top of student 5, and the distribution of scalar is mainly concentrated in the positions of students 4, 6, and 8. In addition, through comparative study with the result from [1], simulation of the given transient process using the Eulerian approach shares the same characteristic of aerosol scalar distribution.

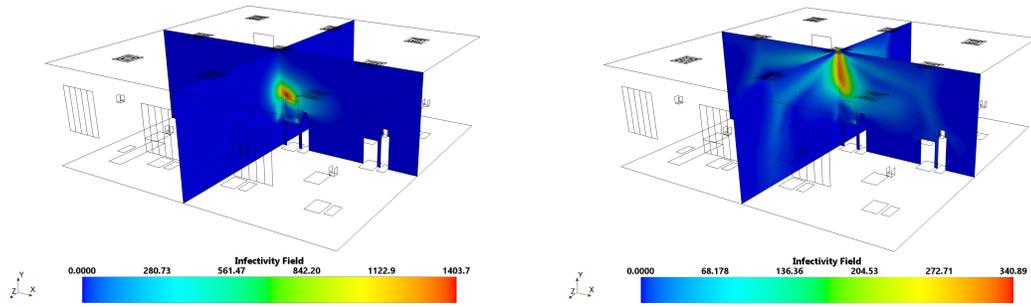


Figure 5.8: Infectivity field C at t=50s (on the left) and 100s (on the right)

It has been observed that a high degree of agreement has been reached between the two methods. We acquired the particles distribution at t=50s and 100s from [1]. Next, we evaluate the spatial distribution of the aerosol scalar with respect to the percentage of the aerosol in divided sections over the total amount of aerosol.

The line plots with respect to the percentage of amount of aerosol particle among total particles number (Lagrangian method) or passive scalar concentration (Eulerian method) along the classroom are compared in Fig 5.9. The probed line is picked along with the central direction of classroom ( $y=2m$ ,  $z=4.5m$ ). The intermedium of probed points along the probed line follows  $x=0.25 : 0.5 : 8.75$  m.

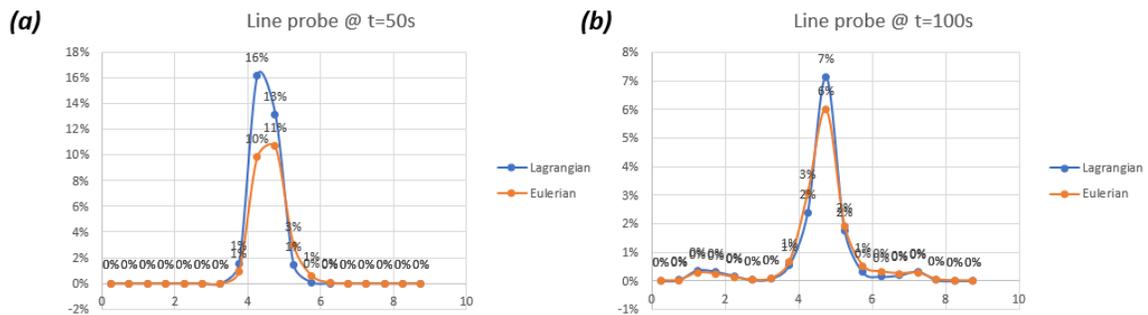


Figure 5.9: Line plot of total percentage of Infectivity Field C/Particle along central direction of classroom. The plot at t=50s is shown in (a); the plot at t=100s is shown in (b).

Along the selected direction, only one peak is shown in the line plot. Despite the difference in

percentage magnitude between Lagrangian and Eulerian approaches, the line plots at different time instants clearly illustrate the agreement between the two approaches. The peaks of the signal are found near the  $x=4.25, 4.75\text{m}$  in Fig 5.9. At  $t=100\text{s}$ , two sub-peaks at  $x=1.25$  and  $7.25\text{ m}$  are found along the central direction. Note that the difference in line plots magnitude between Lagrangian and Eulerian approaches is inevitable because the percentage magnitude depends on the size of the probed boxes, i.e., how many Lagrangian particles contains in a probed box. Again, the probed plots clearly illustrate a reasonable agreement can be achieved between the two approaches.

In Fig 5.10, we evenly divide the classroom model into 3 levels vertically as upper, middle, and lower with a height of 1 m. In each zone, the space is divided into 9 sections with a length and width of 3 m. In general, the entire classroom is divided into 27 sections. By calculating the percentage of aerosol particles in each section with respect to the total amount of indoor aerosol particles, aerosol concentration in each section can be evaluated and then compared between the two approaches. The identified distribution of aerosol spread in the room is matched statistically between Eulerian and Lagrangian approaches at  $t=50\text{s}$  and  $100\text{s}$  as shown in the Table 5.1 and 5.2, respectively.

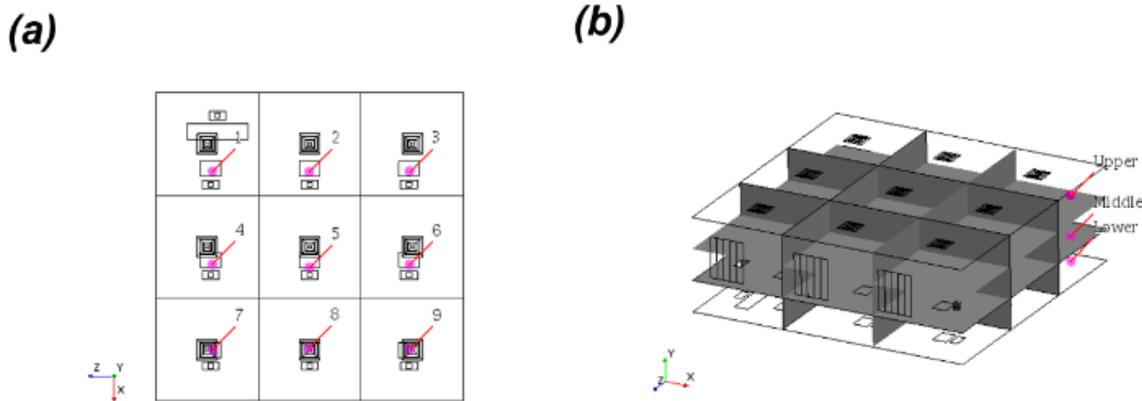


Figure 5.10: Schematic diagram of the partition of the classroom

At  $t=50\text{s}$ , the area with high aerosol concentration is located in the Middle and Upper zones' section 5. A reasonable agreement between Eulerian and Lagrangian methods has been achieved at

this moment. Since diffusion prior to  $t=50s$  dominates the transport of scalar, both the Eulerian and Lagrangian methods predict that most sections outside section 5 have low aerosol concentration.

Table 5.1: The quantitative study between Eulerian/Lagrangian methods at  $t=50s$

		$t=50s$								
Sections		Upper:1	Upper:2	Upper:3	Upper:4	Upper:5	Upper:6	Upper:7	Upper:8	Upper:9
Lagrangian Approach/%		0	0	0	0	19	0	0	0	0
Eulerian Approach/%		0	0	0	0	16	0	0	0	0

Sections		Middle:1	Middle:2	Middle:3	Middle:4	Middle:5	Middle:6	Middle:7	Middle:8	Middle:9
Lagrangian Approach/%		0	0	0	0	81	0	0	0	0
Eulerian Approach/%		0	0	0	0	80	0	0	1	0

Sections		Lower:1	Lower:2	Lower:3	Lower:4	Lower:5	Lower:6	Lower:7	Lower:8	Lower:9
Lagrangian Approach/%		0	0	0	0	0	0	0	0	0
Eulerian Approach/%		0	0	0	0	3	0	0	0	0

When the propagation process reaches a considerable period of time, in the results at  $t=100s$ , we can see that the aerosol particles have spread over the classroom, for which the transport is dominated by convection.

At  $t=100s$ , sections 2, 4, 6, and 8 pose a high risk of infection because these areas have relatively high aerosol concentrations as shown in Table 5.2. At the same time, in sections 1, 3, 7, and 9, the risk of infection is relatively low, as aerosols are rarely transmitted to this area predicted by both Eulerian and Lagrangian methods. Therefore, we can conclude that the aerosol propagation pattern/distribution predicted by the Eulerian method is highly consistent with the works in [1] using the Lagrangian method.

Table 5.2: The quantitative study between Eulerian/Lagrangian methods at t=100s

t=100s									
Sections	Upper:1	Upper:2	Upper:3	Upper:4	Upper:5	Upper:6	Upper:7	Upper:8	Upper:9
Lagrangian Approach/%	0	4	0	5	61	2	0	2	0
Eulerian Approach/%	0	3	0	3	58	3	0	3	0

Sections	Middle:1	Middle:2	Middle:3	Middle:4	Middle:5	Middle:6	Middle:7	Middle:8	Middle:9
Lagrangian Approach/%	0	2	0	3	14	1	0	2	0
Eulerian Approach/%	0	1	0	2	18	2	0	4	0

Sections	Lower:1	Lower:2	Lower:3	Lower:4	Lower:5	Lower:6	Lower:7	Lower:8	Lower:9
Lagrangian Approach/%	0	1	0	2	0	0	0	0	0
Eulerian Approach/%	0	1	0	1	0	0	0	1	0

In conclusion, we confirm, quantitatively, the similar prediction of the spatial distribution of aerosol particles by Lagrangian and Eulerian approaches.

## 5.2 ROM Generation and Optimization for Ventilated Indoor Environments

### 5.2.1 Convergence Issues: Regarding the Residual of Transport Equation Solution

We use commercial CFD solver, STAR-CCM+, to solve the transport equation (2.3) based on iterative approach. It is very important to have an accurate and reliable CFD solution while constructing a Krylov-based ROM. In other word, decent convergence is needed. The problem does not arise in dissuion-dominated processes, such as the heat transfer in a battery pack considered in Chapter 4. It may become acute, however, in processes with a stronger convection component such as the aerosol propagation discussed in this chapter.

Relative residuals and absolute residuals can be used as convergence criteria [52]. Both absolute or relative (to the  $0^{th}$  iteration) value of the residual are good measures of accuracy. If the

initial guess is inaccurate (e.g., we start with a zero scalar everywhere), relative value is more appropriate. If the initial guess is more accurate, and the residual is small from the beginning, using the relative value can sometimes overestimates the convergence error, so the absolute residual is a better choice.

The absolute residual value is a better option to evaluate convergence in our case. Each CFD solution of the ROM derivation algorithm (see section 3.3) starts with the previous solution as an initial guess. The initial value of the residual is, therefore, relatively small ( $\sim 10^{-4}$ ). We use absolute residual in all tests to guarantee an unbiased criterion for the final convergence level of each iteration of ROM.

STAR-CCM+, or similar commercial CFD software, solve the convection equation (2.2) using a  $2^{nd}$  order discretization scheme. The under-relaxation factor (URF) can be adjusted to ensure sufficient convergence. The price to pay by lowering the URF is usually the longer computation time needed to get a fully converged solution.

Additionally, it should be noted that the convergence is not only be judged by a sufficiently small residual. One also need to make sure that variables of the solution remain steady for a sufficiently long period of time.

To summarize, the CFD solution needs to be accurate enough in the ROM calculation process. The strategies and methods that can be used to guarantee the converged solution are adjusting the URF and ensuring the high confidence of the CFD solution by observing the stability of both residual and solution fields as well as the quality of the mesh. The convergence reaches to absolute residual value of  $\sim 10^{-12}$  during the ROM generation.

## 5.2.2 ROM Generation Using Single-Input Algorithm

In this section, we present the results, using the indirect one-sided Arnoldi algorithm 1 to construct ROM. The generated ROMs are assessed to determine the optimal reduced order. We use the Eulerian approach with the commercial CFD software, STAR-CCM+, to solve the discrete diffusion convection equation for the infectivity field  $C$ . Theoretically, as the order increases, the accuracy of the ROM increases accordingly due to the moment matching property of the Krylov

subspace method [29]. However, in the large-scale CFD models with large number of grid cells, the error of the CFD solver accumulates in the course of execution of the ROM-generation algorithm. As a result, there is a trade-off in the process of selecting the optimal reduced order for ROM. We quantify the approximation error by calculating the  $\|\varepsilon\|_2$  and  $\|\varepsilon\|_\infty$  of the difference between the full-order and reduced-order solutions.

The accuracy tests are performed for the source intensity illustrated in Fig 5.5. The intensity is constant 1000 for the first 5s within the source zone and zero otherwise. The simulations are carried out for 100s.

We observe the dynamics of the transient passive scalar signal monitored by a probe point in the model under different  $r$  (shown in Figure 5.11). The probed point is located in front of the source person, student 5. It can be observed that at low  $r$ , the transient signal does not closely follow the dynamics predicted by the FOM solution. But as  $r$  increases, the quality of the ROM prediction increases. It should be emphasized that the fluctuation of the signal keeps existing along with increase of reduced orders.

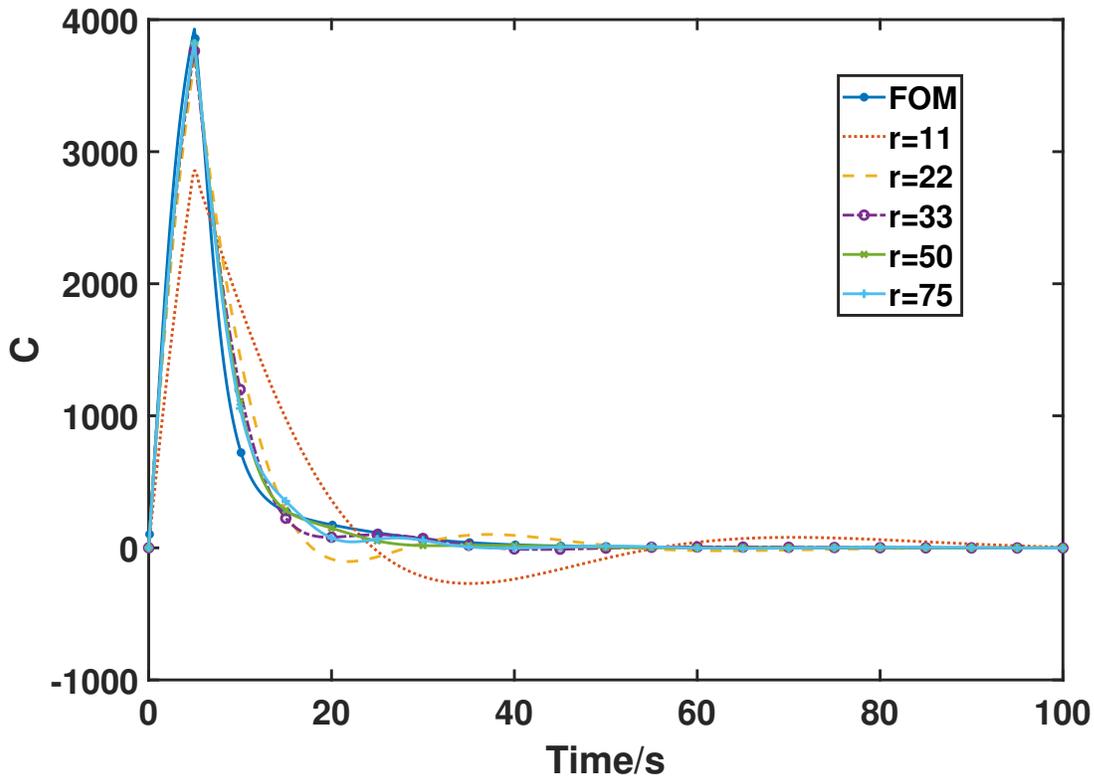


Figure 5.11: Transient signal at the probed point in front of student 5

Secondly, we compare the passive scalar line probed plot of FOM and ROM of different  $r$  in 50s ( $r=5, 15, 30, 60, 85$ ). The probed line is picked along with central direction of classroom ( $y=2.25\text{m}, z=4.5\text{m}$ ). As shown in Fig 5.12, the performance of ROM improves at larger  $r$ . At low order  $r$ , such as  $r=5, 15$ , the line probed signal of ROMs is obviously distorted and deviates significantly from the FOM line probe signal. As  $r$  increases, the line probed signal gradually starts to capture the correct behavior of the system, but there are still relatively large fluctuations locally. Taking  $r=30$  as an example, the probed line signal still has a mismatch near the location  $x=4.5\text{m}$ . Noticeable derivations are also found at  $x=1.5\sim 3.5\text{m}$ . Accuracy is achieved at high values of  $r$ , such as at  $r=60, 85$ . The probed line of the ROMs can already match the probed signal of the FOM very well.

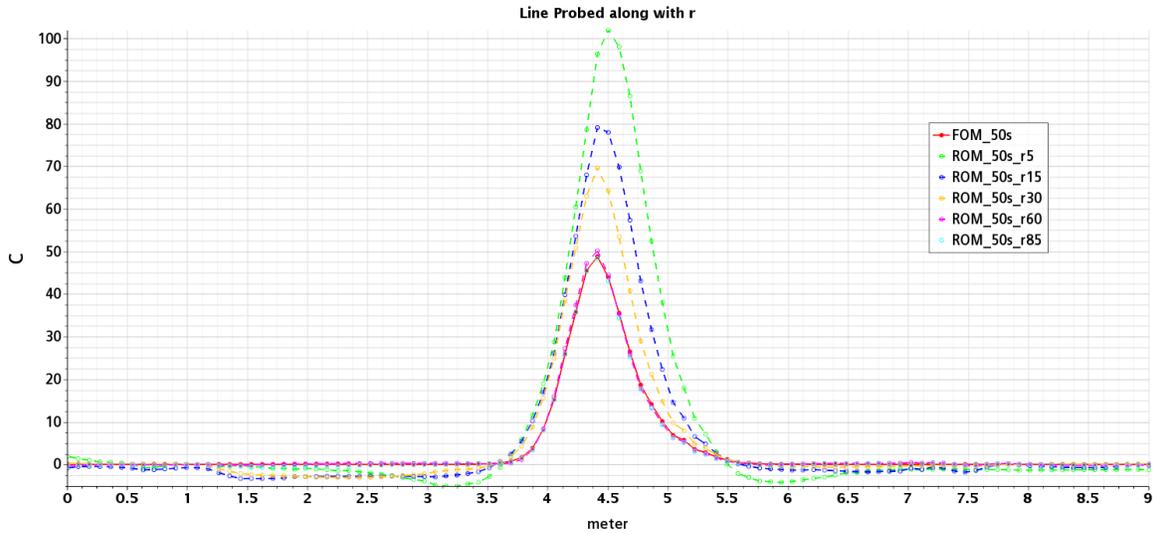


Figure 5.12: Distributions of  $C$  along the line at  $y=2.25\text{m}$ ,  $z=4.5\text{m}$  obtained by the FOM and by SIMO ROMs with various  $r$ .

We further compared the fit of the ROMs with the FOM at different  $r$  using the contour in the cross-section at  $z=4.5\text{m}$ . Their approximation degrees under the high frequency response ( $t=5\text{s}$ ) and low frequency response ( $t=50\text{s}$ ) of the system, respectively, we calculated the results in the corresponding ROM through contours with FOM.

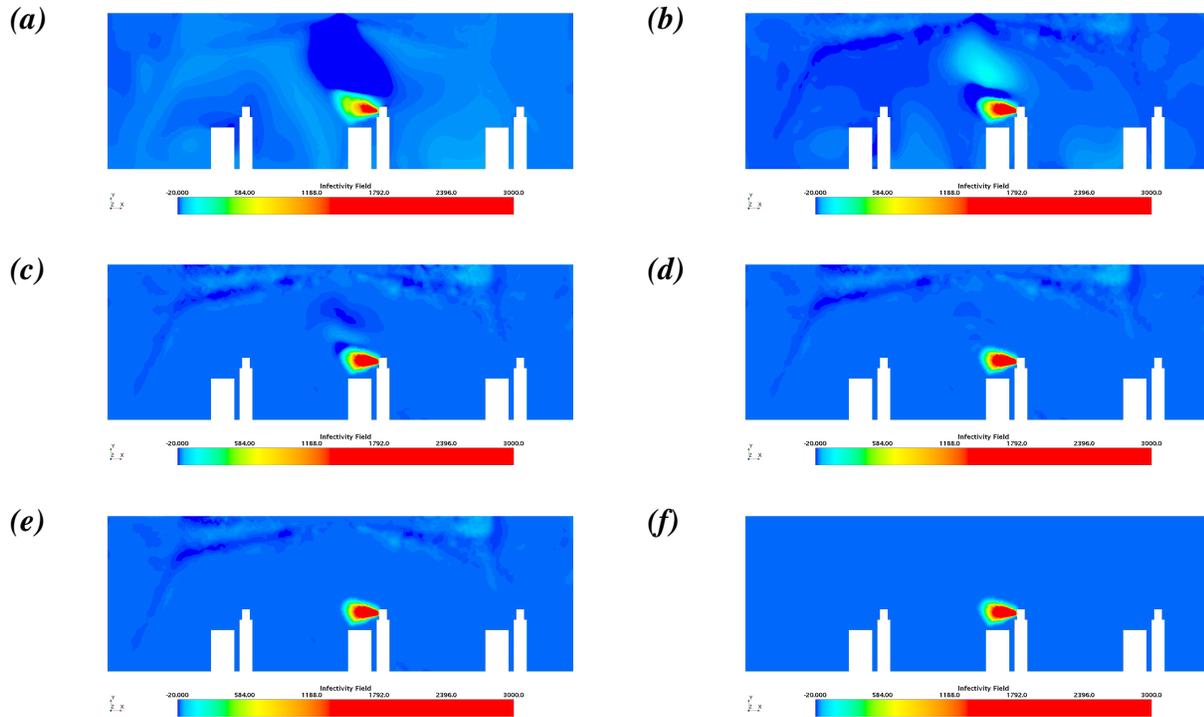


Figure 5.13: Contour plot at  $t=5s$  in the cross-section  $z=4.5m$  under the signal of 5.5. ROM of (a)  $r=5$ ; (b)  $r=15$ ; (c)  $r=30$ ; (d)  $r=60$ ; (e)  $r=85$ ; (f) FOM

In Figure 5.13, the low-order ROMs show obvious deviations from the FOM solution. As  $r$  increases to 60 and 85, the ROM becomes more accurate although some unphysical fluctuation in the upper part of the cross-section are still visible.

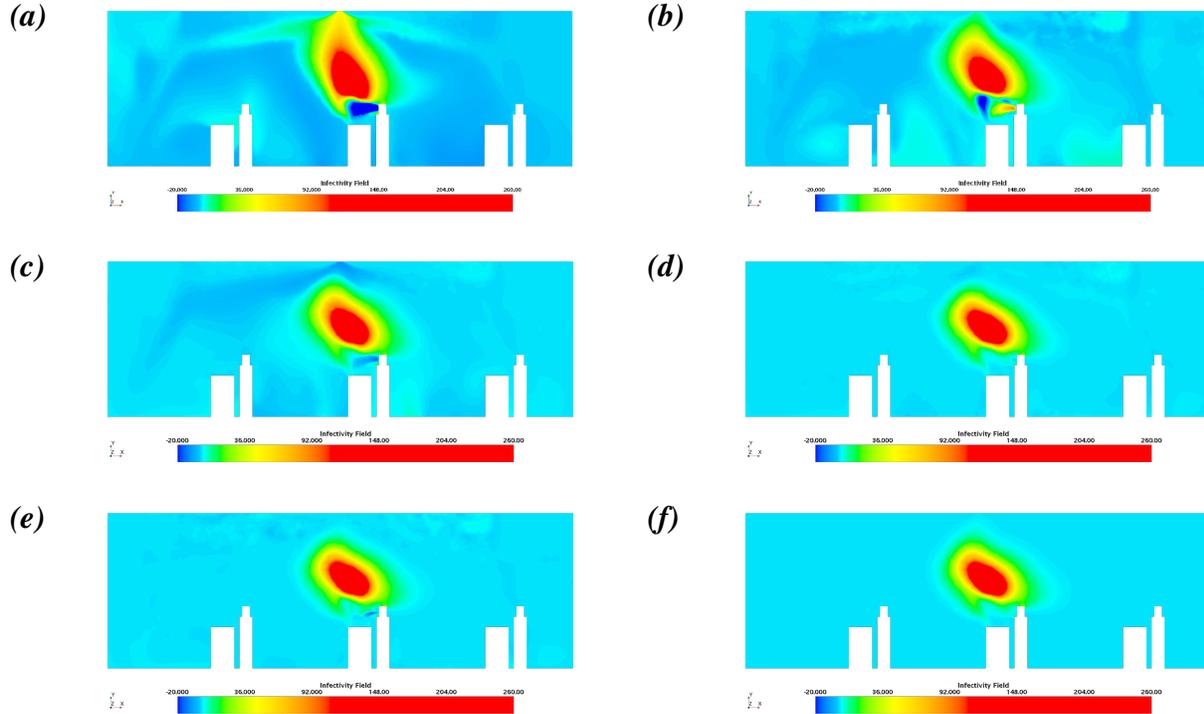


Figure 5.14: Contour plot at  $t=50s$  in the cross-section  $z=4.5m$  under the signal of 5.5. ROM of (a)  $r=5$ ; (b)  $r=15$ ; (c)  $r=30$ ; (d)  $r=60$ ; (e)  $r=85$ ; (f) FOM

We also compared the predictions of ROMs and FOM at 50s as shown in Figure 5.14. Good accuracy is found at high  $r$ . In the case of  $r=5$  or 15, the ROM predicts that the C circulates around the ventilation flow and the convection occurs more violently. In the meantime, obvious distortion in the field is observed near the ventilation exit and in front of student 5. When  $r=15$ , there is local obvious noise in the transport phenomenon in front of the source person. At  $r=30$ , although the ROM can basically simulate the transport pattern, we still observe that the transport prediction near the vent and in front of the source person has relatively large interference. Interestingly, ROM at  $r=60$  has better accuracy than at  $r=85$ . To confirm this observation, we further evaluated the simulation accuracy of ROM in a quantitative manner.

The results are presented in Table 5.3, where we show the  $\|\varepsilon\|_2$  and  $\|\varepsilon\|_\infty$  for the entire domain for different values of  $r$  at 5s and 50s. We observe that the simulation accuracy of ROM gradually increases with the increase of  $r$ . But as  $r$  rises to a certain value, the accuracy of ROM begins to

decrease. Therefore, we choose  $r=65$  as the optimal ROM in the case of the considered system.

Table 5.3: The quantitative evaluation for ventilation ROM using **Algorithm 1**.

r	5s		r	50s	
	$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$		$\ \varepsilon\ _2$	$\ \varepsilon\ _\infty$
1	4101.71614	0.033970722	1	420.0520974	0.004919662
2	3897.204849	0.0327505	2	464.5043782	0.005266173
3	3209.994665	0.027981335	3	261.2120021	0.004700886
4	2643.473463	0.024007687	4	177.4318848	0.00393388
5	2405.838849	0.022142363	5	160.9852151	0.00340241
6	2193.454966	0.020517757	6	231.9732018	0.003135373
7	2059.275084	0.019469939	7	273.1889522	0.003136924
8	1887.51468	0.018219563	8	301.2598654	0.003070044
9	1736.582754	0.017197164	9	337.2806877	0.003132623
10	1373.395574	0.014404866	10	274.3627911	0.002946727
11	1184.091838	0.01257946	11	186.1686172	0.002787503
12	1014.870606	0.010761379	12	136.9093779	0.002324464
13	900.0639051	0.009412445	13	124.8334285	0.001947135
14	809.0717399	0.008383301	14	120.0027065	0.001719401
15	716.5651398	0.007404798	15	109.6250602	0.001522216
16	629.4961751	0.006417915	16	112.3293944	0.001433778
17	481.0320972	0.005281978	17	114.3209319	0.001278743
18	579.3687363	0.005197281	18	99.00432428	0.001233456
19	449.6107714	0.00424985	19	76.66943915	0.001139877
20	373.4839549	0.003678724	20	58.95351351	0.001050186
21	348.0382623	0.003230563	21	50.05238558	0.000971481
22	357.9744149	0.003018883	22	50.64196978	0.00089657
23	365.7603587	0.002872882	23	53.76353273	0.000805993
24	374.760428	0.002792323	24	61.05185464	0.000734782
25	376.8608569	0.002781679	25	60.55166094	0.000702364
26	380.9151918	0.002777583	26	56.73706994	0.000673788
27	385.5845241	0.002791613	27	43.48603165	0.000603633
28	386.9239719	0.002796791	28	33.9478127	0.00056523
29	387.2511066	0.002796745	29	25.7624994	0.000538178
30	381.7347837	0.002751962	30	23.22064723	0.000506549
31	385.5874459	0.002775623	31	23.26875473	0.000473082
32	384.119255	0.002758119	32	20.12503697	0.000432877
33	379.1195754	0.00270728	33	21.05494061	0.000404867
34	378.1193029	0.002701546	34	24.4136724	0.000373739
35	375.1193819	0.002671556	35	24.74476856	0.00035428
36	370.8379458	0.002630417	36	23.76344851	0.000337107

37	360.3596259	0.002517002	37	19.89312445	0.000323574
38	361.9317091	0.002510891	38	16.12372054	0.000316971
39	356.3608593	0.002459875	39	14.65690652	0.000305186
40	353.5764519	0.002411337	40	13.65480811	0.000294066
41	349.5181615	0.002360237	41	12.39039554	0.000281039
42	340.9504808	0.002267651	42	10.77884304	0.000263454
43	339.2978647	0.002249151	43	9.566147923	0.000244988
44	336.2699749	0.002216663	44	8.870075914	0.000233505
45	333.1574472	0.002184976	45	9.549938831	0.000223595
46	327.463079	0.002143439	46	10.60079039	0.000214064
47	318.8379016	0.002057057	47	11.14635746	0.00020724
48	318.7863228	0.002071424	48	11.82151129	0.000194305
49	312.2281737	0.002028678	49	12.13707098	0.000189179
50	310.9706287	0.002040752	50	12.40439857	0.000184712
51	305.0965482	0.002014014	51	12.52483672	0.000182363
52	300.8735916	0.001994799	52	12.51498364	0.000178176
53	298.2618394	0.001991562	53	12.25911892	0.000173662
54	294.671046	0.001994129	54	11.81147078	0.000167723
55	291.4015874	0.001990397	55	11.20097912	0.000164592
56	287.254523	0.001982461	56	10.41117734	0.000157704
57	274.9814845	0.001930557	57	9.332362694	0.000163888
58	280.0147708	0.001975191	58	8.450156536	0.000152497
59	277.3655784	0.001972204	59	7.760303013	0.000149295
60	275.1197137	0.001975357	60	7.056458758	0.000150088
61	271.5996527	0.001971519	61	6.938409566	0.000150522
62	269.9444072	0.001969991	62	7.488890788	0.00015198
63	262.8344384	0.001943989	63	8.294343379	0.000158903
64	266.3766401	0.001978931	64	9.428823962	0.000158024
65	264.4725261	0.001980763	65	10.32634107	0.000162373
66	263.2362821	0.001982362	66	11.00605074	0.000165951
67	254.6469817	0.001955302	67	11.997259	0.000175456
68	259.4744452	0.00198699	68	13.04606974	0.000178997
69	258.4825942	0.001990735	69	13.76362921	0.000184953
70	256.2293271	0.001990671	70	14.62744158	0.000193826
71	255.3931855	0.001996734	71	15.55298206	0.000203092
72	243.0935102	0.001971095	72	16.77684116	0.000219302
73	252.3462542	0.002002973	73	17.6671766	0.000226228
74	250.7122946	0.00200422	74	18.70726266	0.000239174
75	245.5665462	0.001993861	75	19.91535501	0.000257106
76	248.6873329	0.002012259	76	21.0015391	0.000272937
77	246.790942	0.002009456	77	22.02492086	0.000287745
78	247.4924163	0.002016787	78	23.04102533	0.000302372
79	247.0724657	0.002017424	79	23.81176658	0.000312108

80	247.1386967	0.002014025	80	22.3660962	0.00029611
81	240.6176409	0.001993884	81	21.50051095	0.000288445
82	240.4322138	0.001993596	82	21.4791231	0.000288299
83	243.8656064	0.001999515	83	20.60844857	0.000280471
84	249.7060725	0.002009539	84	14.17542578	0.000227744
85	250.4318961	0.002007852	85	32.00092188	0.000281084

The ROM of order 65 is validated in predicting the transient dynamics of the ventilation system under the same transient signal as in Figure 5.5. It can be observed that the ROM can well predict the distribution of the infectivity field  $C$ . As an example, we compare the ROM of with  $r=65$  and FOM at  $t=100s$  in Fig 5.15.

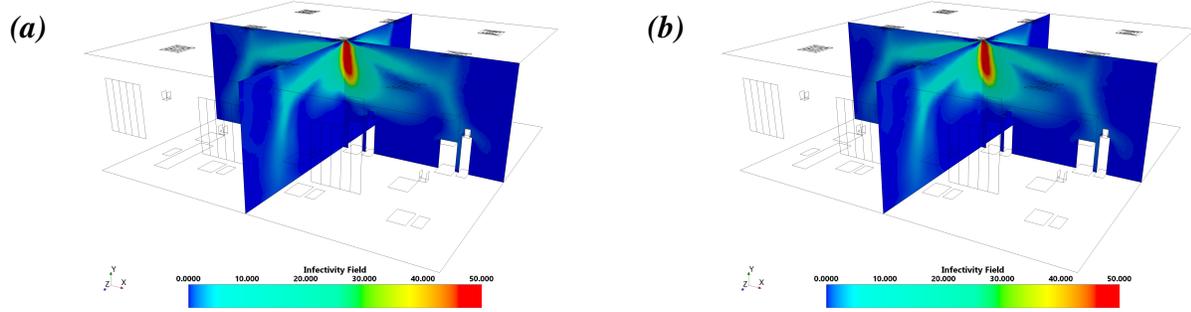


Figure 5.15: Infectivity field  $C$  at  $t=100s$ . (a) Optimal ROM, with  $r=65$ , using SIMO algorithm, and (b) FOM.

After the comparative study, ROM of  $r=65$  can be used as the optimal result, which can well simulate the aerosol distribution of the original large-scale model.

### 5.2.3 ROM Generation Using the Frequency-Shift Algorithm

In this section, we present the ROMs generated by frequency-shift Arnoldi algorithm, following with optimization of reduced order. We select a total of 11 sampling points as Sigma set in construction of ROM following the **Algorithm 3**. As presented in Table 5.4, the Sigma set is picked from a range of frequency to capture the characteristics of the system among different response rates, i.e., from low frequency to high frequency. For each of the 11 value Sigma, we develop ROMs of different orders using the frequency shift Arnoldi approach.

Table 5.4: The 11 Sigma points for frequency-shift Krylov method

	1	2	3	4	5	6	7	8	9	10	11
Sigma	0	0.001	0.002154	0.004642	0.01	0.021544	0.046416	0.1	0.215443	0.464159	1

### 5.2.3.1 Time Derivative Method Without Access to the System Matrix A

We apply a new method to construct the orthogonalization matrix  $X$  during the frequency-shift algorithm. The method discretizes the time derivative of the field using the 2nd order scheme. Technically, the method is validated to be able to produce an accurate approximation of the field. In our case, the orthogonalization matrix  $X$  is defined as  $X = (A - \sigma \mathbf{I})^{-1}V$ . The method calculates the time derivative of  $C$  to get the vectors of matrix  $X$ . Therefore, the vectors in  $X$  is defined as  $x_i = (A - \sigma \mathbf{I})^{-1}v_i$ .

As mentioned before, commercial CFD software, in general, does not give access to the system's matrix. We have developed an indirect method to get the necessary information. In one-sided Arnoldi algorithm, the orthogonalization matrix  $X$  is defined as  $AV = A[v_1 v_2 \dots v_r]$ . The proposed method calculates the 2<sup>nd</sup> order forward time derivative of passive scalar field in order to get  $x_i$ :

$$\frac{dC}{dt} = (A - \sigma \mathbf{I})C \approx \lim_{\Delta t \rightarrow 0} \frac{3C_0 - 4C_{0+\Delta t} + C_{0+2\Delta t}}{2\Delta t}, \quad (5.1)$$

where reasonable approximation of  $AV$  can be achieved with substantially small time step  $\Delta t$ .

One can validate the method by evaluating the  $b = Av_1$ . The 2nd norms of cell-by-cell error vector  $b - Av_1$  does not exceed  $10^{-6}$ . As a conclusion, time-derivative method is a reasonable way to access to the system matrix when  $A$  cannot be accessed directly.

### 5.2.3.2 ROM Optimization

The optimization of ROM developed using the frequency shift Arnoldi algorithm is presented in this section. We apply the same step-change signal as the one used for the Arnoldi algorithm optimization process (See Figure 5.5) to evaluate the ROM approximation.

The computed time-varying signals at the probed point are presented. The ROM of  $r=11$  matches the  $0^{th}$  order moment of the system. The transient signal that matches only one moment is off-track for completely missing the dynamics of the system after 5 seconds. As  $r$  goes to higher order, the transient signal at  $r=22, 33$  can follow the dynamics from FOM that produce the high-fidelity result of the system. However, we can observe the fluctuation of the signal at high-frequency response area. For example, the signal at  $t$  from 10s to 20s. As  $r$  increase, the ROM starts to closely follow the temporal dynamics of the system.

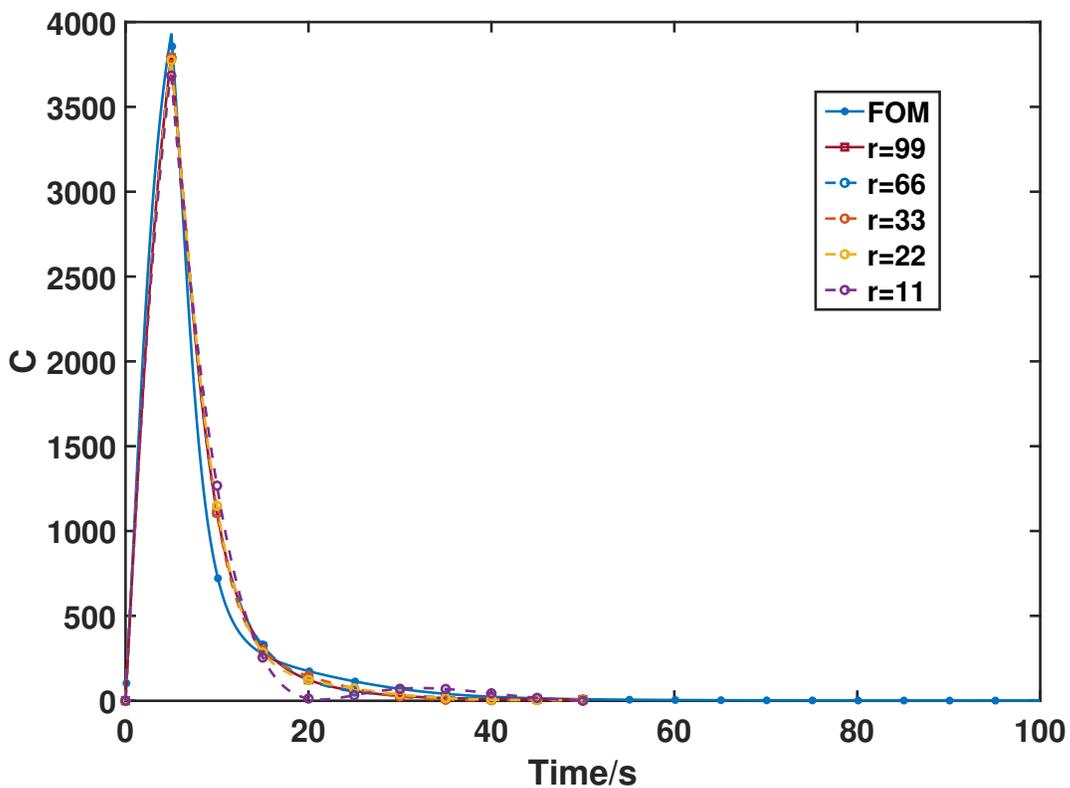


Figure 5.16: Results of frequency-shift ROMs for the source signal in Fig 5.5

Secondly, we compare the passive scalar line probed plot of FOM and ROM of different  $r$  at 50s ( $r=11, 22, 33, 66, 99$ ). The probed line is picked along with central direction of classroom ( $y=2.25m, z=4.5m$ ). As shown in Fig 5.17, the line probe signals of ROM moves closer to that of FOM at higher  $r$ . The accuracy is poor at low values of  $r$  ( $r=11, 22, 33$ ). As  $r$  increases, the line

probed signal can gradually capture the behavior of the system, but there is still a noticeable error, i.e., in the range of  $x=[4, 5]$  m. The results obtained at  $r=99$  are more accurate than the results of  $r=66$ .

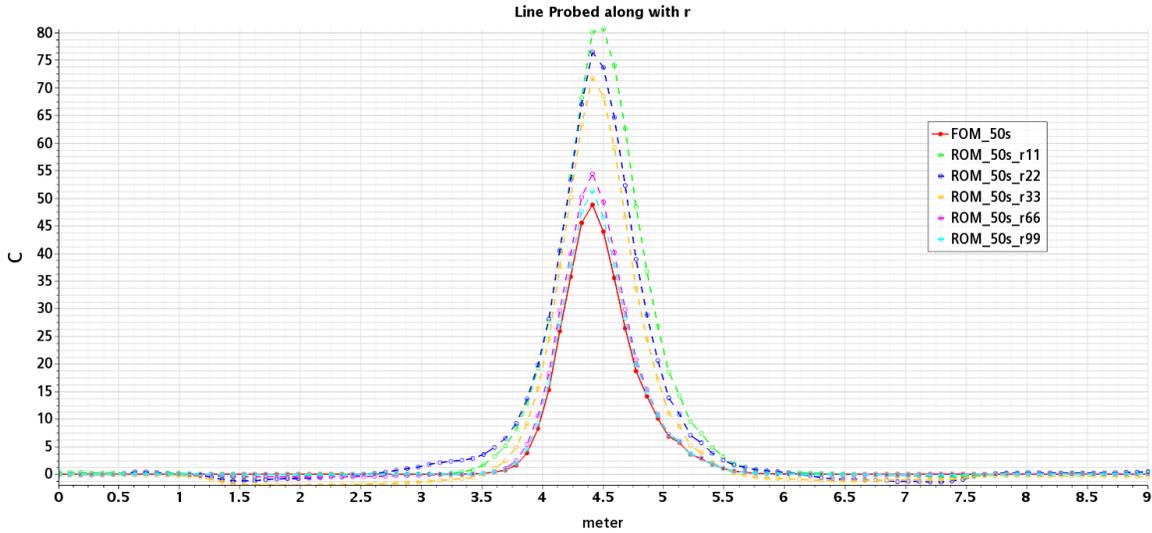


Figure 5.17: Distributions of  $C$  along the line at  $y=2.25\text{m}$ ,  $z=4.5\text{m}$  obtained by the FOM and by frequency-shift ROMs with various  $r$ .

We further evaluate the ROMs at different  $r$  comparing with the FOM contour plot of  $C$  in the cross-section at  $z=4.5\text{m}$ . The contours at  $t=5\text{s}$  and  $50\text{s}$  are plotted in order to assess the high-frequency and low-frequency response, respectively. At  $t=5\text{s}$ , ROMs of lower order presents significant errors. As shown in Figure 5.18, the ROM of  $r=11$  in (a) illustrate the basic structure of the field. The contour illustrates the high-concentration cloud of aerosol located in front of index person. However, the contour illustrates obvious deviations around the field. As  $r$  increases, the ROM can be more accurate, in the case of  $r=66$  or  $99$ , to approximate the high-frequency characteristic of the FOM.

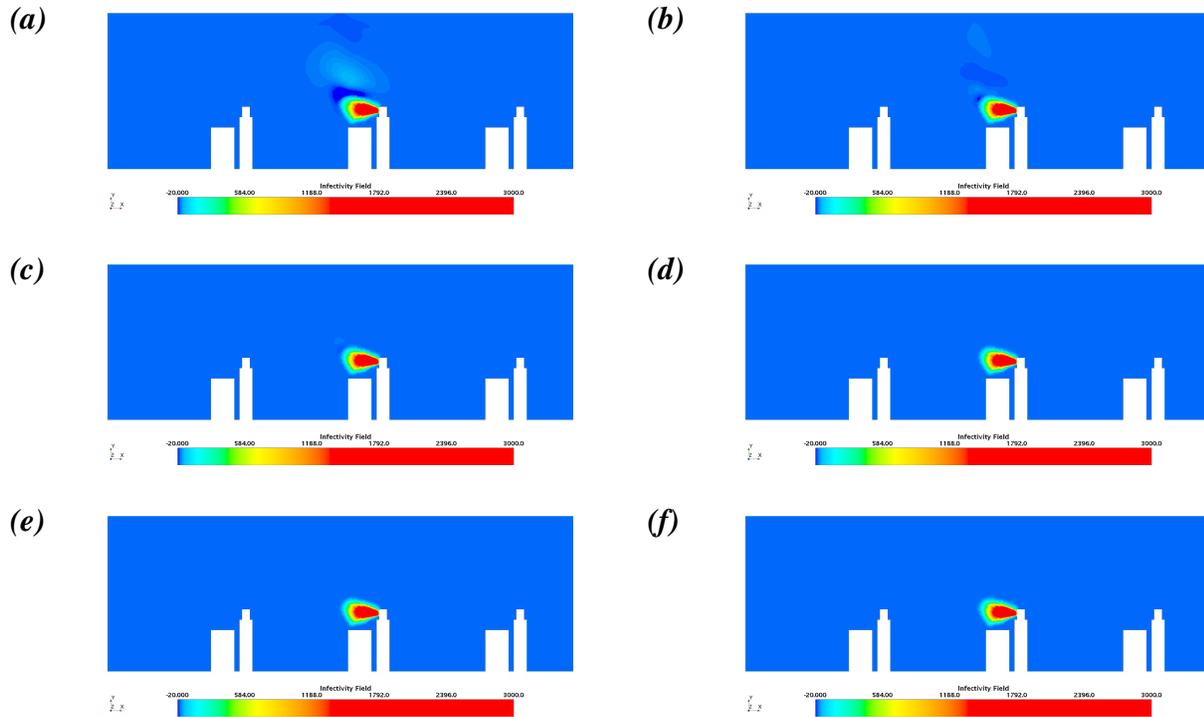


Figure 5.18: Contour plot at  $t=5s$  in the cross-section  $z=4.5m$  under scalar intensity signal 5.5. Frequency-shift ROMs of (a)  $r=11$ ; (b)  $r=22$ ; (c)  $r=33$ ; (d)  $r=66$ ; (e)  $r=99$ ; (f) FOM

The ROMs at  $t=10s$  of different  $r$  are shown in Figure 5.19. The ROMs of  $r=66$  and  $99$  shows a robust approximation of the FOM. As shown in (f) and (g), the distribution of the passive scalar is almost indistinguishable compared with FOM. The error vectors' contours between ROMs and FOM are presented in Figure 5.20. The error mainly locate near the ventilation flow between index person and ventilation inlets at  $r=11$ . As  $r$  increase to  $99$ , the fluctuation errors are largely removed outside the aerosol cloud area.

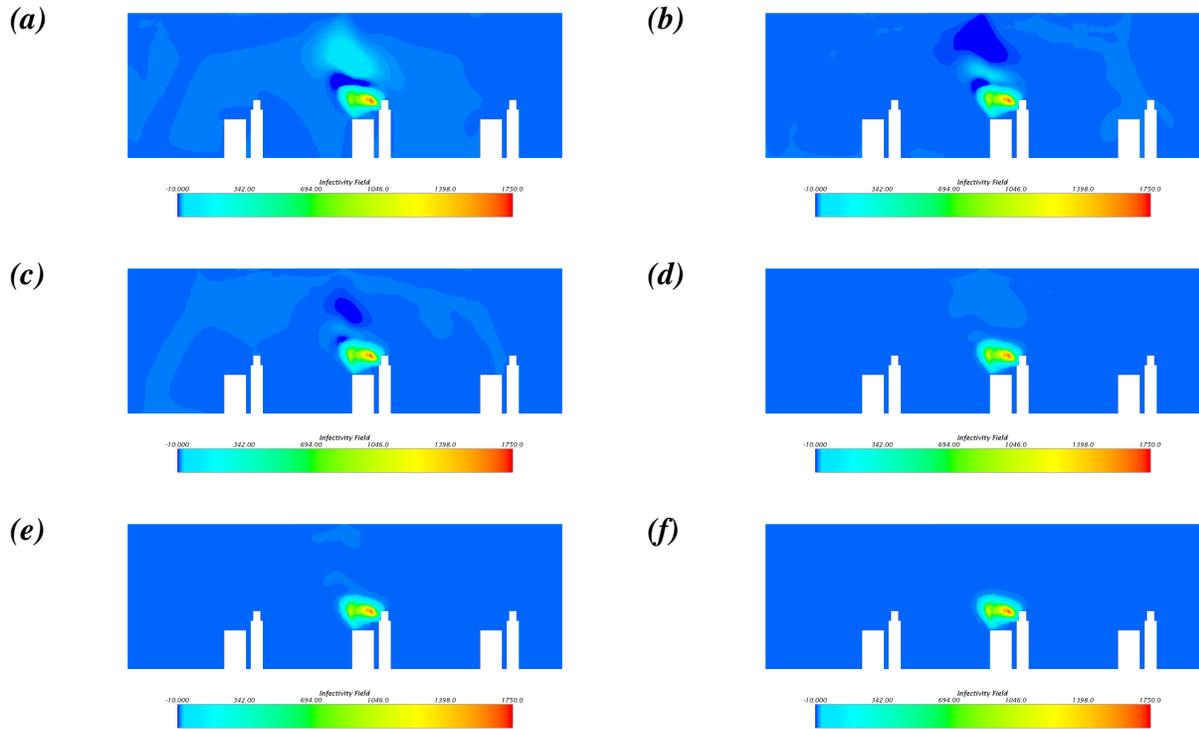


Figure 5.19: Contour plot at  $t=10s$  in the cross-section  $z=4.5m$  under scalar intensity signal 5.5. Frequency-shift ROMs of (a)  $r=11$ ; (b)  $r=22$ ; (c)  $r=33$ ; (d)  $r=66$ ; (e)  $r=99$ ; (f) FOM

Note that the ROMs using frequency-shift Arnoldi algorithm significantly reduce the error. As shown in Figure 5.13, the ROMs using conventional Krylov method show strong fluctuations of  $C$  in comparison with the FOM distribution even with optimal reduced order. The fluctuations are strongly reduced by using the frequency-shift Arnoldi algorithm.

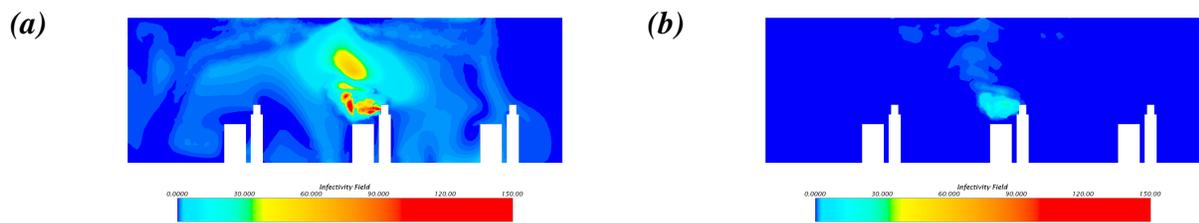


Figure 5.20: Contour plot of error vector at  $t=10s$ . ROM of (a)  $r=11$ ; (b)  $r=99$ .

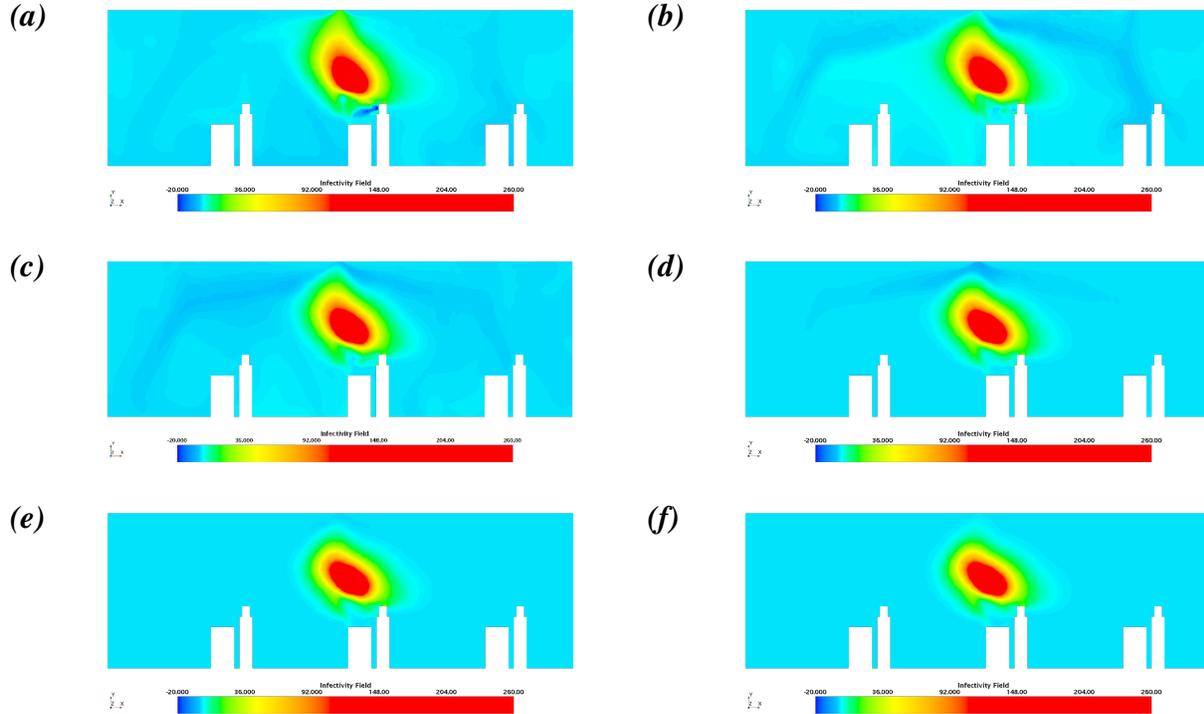


Figure 5.21: Contour plot at  $t=50s$  in the cross-section  $z=4.5m$  under scalar intensity signal 5.5. Frequency-shift ROMs of (a)  $r=11$ ; (b)  $r=22$ ; (c)  $r=33$ ; (d)  $r=66$ ; (e)  $r=99$ ; (f) FOM

Next, we compare the predictions of ROMs at 50s to evaluate the low-frequency response of the ROMs. The results are illustrated in Fig 5.20. Large unphysical fluctuations of  $C$  are predicted by the ROMs at lower reduced orders. For example, the ROM of  $r=11$  produces significant error near the boundary of aerosol cloud. The ROMs of  $r=22, 33, 66$  have obvious errors at the field near the inlets. It is found that only when  $r$  is high enough, the FOM can be accurately approximated (see Figs 5.21 d-f). It can be seen that when  $r=99$ , ROM simulates the slow-dynamic characteristics of the system accurately. To confirm this observation, we further evaluated the simulation accuracy of ROM using the norms of error vectors at different time moments.

The computed norms of the error vector at  $t=5$  and  $50s$  are shown in Table 5.5. It is observed that the ROMs using frequency shift algorithm perform more accurately than the ROMs generated by the SIMO algorithm. The 2nd and max norms of the error show a compatible approximation at  $r=33$  with previous ROM of optimal order, i.e.,  $r=65$ . Furthermore, ROMs produced by frequency

shift Arnoldi algorithm improve the approximation of high-frequency response. For example, the max norm of error at t=5s has been dramatically reduced at r=85 if compared with that using SIMO algorithm.

Table 5.5: The quantitative evaluation for ventilation ROM using **Algorithm 3**.

r	5s		r	50s	
	$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$		$\ \varepsilon\ _\infty$	$\ \varepsilon\ _2$
1	4140.738195	0.03429009	1	424.3104154	0.005002981
2	3872.915672	0.032636572	2	475.6279231	0.005371553
3	3868.398211	0.032625085	3	447.7609663	0.005155719
4	2984.428398	0.026558019	4	209.8513179	0.004580417
5	2055.412017	0.019686727	5	253.1895297	0.00306651
6	2059.551988	0.019695938	6	251.9027246	0.003041048
7	994.3827201	0.010891809	7	190.9258701	0.0030331
8	351.3838783	0.003417481	8	109.8881168	0.001629379
9	286.1558871	0.002476145	9	86.52083028	0.000957381
10	284.6152344	0.002477172	10	86.29178799	0.000957047
11	294.7784753	0.00253217	11	85.37398925	0.000957132
12	290.348287	0.002513407	12	89.04973739	0.00091977
13	280.5593175	0.002462447	13	90.32852677	0.000921084
14	280.717766	0.00245801	14	84.58507311	0.000985052
15	280.6759955	0.002454161	15	85.42174877	0.000978037
16	270.1835864	0.002394936	16	68.75056717	0.001103538
17	269.305744	0.002396156	17	77.85048094	0.000913818
18	322.2060712	0.002235286	18	63.97881905	0.000877883
19	269.1437511	0.001981614	19	48.60275106	0.000814185
20	196.9411118	0.001717461	20	46.11742955	0.000788997
21	87.04535113	0.000496735	21	38.96297114	0.000668758
22	95.21826627	0.0004599	22	36.72728765	0.000621495
23	95.12572062	0.000461065	23	36.92057896	0.000630628
24	94.45730934	0.000462076	24	36.49872509	0.000646658
25	94.50420003	0.000456035	25	36.53172436	0.000608468
26	94.56666133	0.000459529	26	35.60666628	0.000615415
27	95.09070737	0.000458524	27	34.88218678	0.000594917
28	94.85938324	0.000455672	28	47.83768379	0.000509329
29	94.78131777	0.000459864	29	51.20705135	0.000450591
30	92.97670777	0.0004637	30	34.84213546	0.000419225
31	102.7345595	0.000498162	31	29.79555923	0.000407504
32	96.13269183	0.000339518	32	29.63707332	0.000400093

33	74.16153765	0.000286972	33	29.94253344	0.000397919
34	74.8185961	0.000286509	34	29.9995258	0.000393284
35	73.18978949	0.000288978	35	29.81938558	0.000378538
36	75.691734	0.00028609	36	30.04034733	0.000368594
37	76.16985833	0.000285139	37	29.4139569	0.000373346
38	75.40586068	0.000282799	38	28.81070929	0.000362738
39	76.38162624	0.00028283	39	25.82486129	0.000355806
40	76.88557145	0.000286754	40	22.15686792	0.000347243
41	70.64468065	0.00028431	41	21.73414025	0.000340462
42	80.3399607	0.000285165	42	21.6523286	0.000339963
43	84.63160314	0.000281628	43	20.91215712	0.000339243
44	84.52769471	0.000282077	44	27.15305415	0.000335716
45	84.48766833	0.000282165	45	27.68647875	0.000329695
46	84.5982654	0.000282049	46	28.28085811	0.000326172
47	84.38433642	0.000282678	47	27.42632961	0.000336297
48	83.13373865	0.000283416	48	26.88153079	0.000331807
49	81.71833426	0.000284112	49	26.58917599	0.000328711
50	82.85945237	0.000280403	50	22.02096767	0.000333279
51	88.49075232	0.000286043	51	18.64539311	0.000303598
52	88.98159797	0.000284216	52	11.53393584	0.000222751
53	84.85617623	0.00025752	53	11.29946056	0.000214516
54	82.10516749	0.000236192	54	11.39499322	0.000211473
55	80.83618406	0.000229284	55	11.09211658	0.000211159
56	80.23495479	0.000229685	56	11.22642284	0.000209509
57	80.99212561	0.000229824	57	11.2933125	0.000207852
58	80.81120688	0.000230307	58	11.53847567	0.000206681
59	80.72245316	0.000230333	59	11.53824361	0.000206755
60	80.38113269	0.000230464	60	11.53868949	0.000206822
61	79.12503821	0.000231323	61	11.53736913	0.000207302
62	79.52041087	0.000231239	62	11.08537666	0.000209503
63	80.60808813	0.000231679	63	9.719092421	0.000190487
64	79.74282466	0.000232522	64	10.45560793	0.000166313
65	80.94356662	0.000226247	65	9.762007456	0.000161283
66	80.21971661	0.000224379	66	9.812793271	0.000160651
67	80.15292254	0.00022388	67	9.858599507	0.000160376
68	80.13184813	0.000223841	68	9.843595986	0.00016099
69	80.09164418	0.000223853	69	9.889235497	0.00016133
70	80.26324389	0.000223825	70	9.890472838	0.000161144
71	80.11301682	0.000223883	71	9.909891519	0.000160963
72	79.84987124	0.000223946	72	9.911772143	0.00016111
73	80.41588967	0.000223536	73	9.993808244	0.000154979
74	79.61406187	0.000223843	74	7.44763887	0.000147302
75	79.74200266	0.000222421	75	8.45632956	0.000136316
76	79.89825842	0.000223552	76	9.003205244	0.000130581

77	79.42457592	0.000218642	77	8.803988357	0.000128499
78	78.81811283	0.000218809	78	8.815514561	0.00012866
79	78.15710558	0.000219241	79	8.806006378	0.00012873
80	79.49446507	0.000218638	80	8.827649788	0.000127879
81	79.39284568	0.000218646	81	8.814389973	0.00012799
82	79.09378749	0.00021875	82	8.814828881	0.000128132
83	78.77479734	0.000218887	83	8.813601796	0.000128404
84	78.51413584	0.000220823	84	8.775443972	0.000123604
85	79.25587066	0.000219975	85	9.097768139	0.000121834
86	77.97167387	0.000219744	86	7.910817006	0.000108163
87	77.47846467	0.000220641	87	7.823645515	0.000107796
88	77.96779923	0.000219313	88	7.829633899	0.000109517
89	78.11194335	0.000219124	89	7.807124303	0.000109407
90	78.12878867	0.000219139	90	7.807935979	0.000109109
91	78.08862327	0.000219205	91	7.799255806	0.000108844
92	78.0487269	0.000219199	92	7.803915837	0.000109723
93	77.79960026	0.000219261	93	7.806329952	0.000109621
94	78.07206798	0.000219172	94	7.806841216	0.000109584
95	78.02977001	0.000219243	95	7.866997447	0.000108631
96	77.20928465	0.000218776	96	7.930122796	0.000108553
97	76.99507008	0.000219701	97	8.273007198	0.000101248
98	76.21877517	0.000220542	98	8.255143466	0.000100166
99	78.14531087	0.000220394	99	8.243749986	9.82E-05

Through Table 5.5, we found that the ROM have optimal accuracy with respect to high-frequency and low-frequency response at  $r=88$ . Thus, the ROM of  $r=88$  is the optimal approximation of the system. As  $r$  rises above 88, the accuracy of ROM remains approximately the same. All the further Ventilation-ROM cases will be verified and evaluated using the optimal frequency-shift ROM of  $r=88$ .

### 5.3 ROM Application

In this section, we apply the optimal ROM generated by frequency-shift Arnoldi algorithm to predict the aerosol transmission in an indoor classroom.

### 5.3.1 Exhalation Cycle

A time-dependent infectious aerosol released from an human source student in the classroom is the subject of current study. We define the source term  $S(\mathbf{x}, t)$  in (2.3) as  $S(\mathbf{x}, t) = \mathbf{B}(\mathbf{x})\mathbf{u}(t)$ , where  $B(\mathbf{x})$  is the localized spatial distribution, and  $u(t)$  is the time modulation. The spatial distribution is limited to the generic conical breathing cloud illustrated in Fig 2.2. The cloud is 0.5m long. The circle radii of the upper and lower bases are 0.2 and 0.04m, respectively. The source intensity gradually decreases along the central axis of the breathing cone. The source intensity follows a normal curve distribution within the cross-sectional direction.

The release of the aerosol within the cone is defined as a time-dependent modulation  $u(t)$ . The signal considered in our study is the combination of breathing cycle and coughing events of different magnitude. During breathing activities, the source person regularly exhales a volume of aerosol cloud from the mouth. The breathing cycle has a period of 4 seconds. According to the previous study of exhaling activities [48]. The first two seconds of the cycle are the exhalation phase. Following the exhalation phase, the next two seconds are the inhalation phase. We assume that the intensity of the source is described by a sinusoidal function during the exhalation phase and zero during the inhalation phase (see Figure 5.22 (a)).

The coughing cycle has period of 0.6s [48]. The time modulation  $u(t)$  during one cycle is taken from the experiments [48] (see Figure 5.22 (b)). Considering that coughing varies in intensity, we use the coughing cycles of different peak intensity magnitudes. As shown in Figure 5.22 (b), the coughing intensity peaks are defined as 200, 400, 600 and 1000. The cumulative source modulation signal used in the computational validation discussed below is shown in Fig. 5.23. In the most severe coughing situation, the peak value of the breathing cone intensity is  $\sim 1000$ .

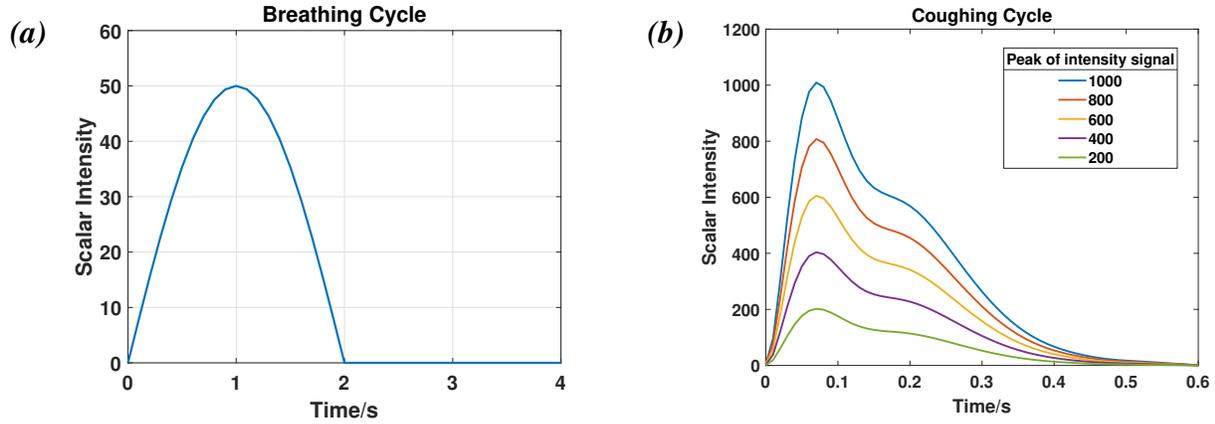


Figure 5.22: Breathing and coughing cycles used in simulations. Source modulations  $u(t)$  corresponding to breathing cycles (a) and coughing cycles (b) are shown.

### 5.3.2 ROM Validation Under the Designed Coughing Cycle

The validation study based on the scenario when the infections aerosol particles are generated within the cone in front of the student 5 according to the time modulation shown in Fig. 5.23 is presented here.

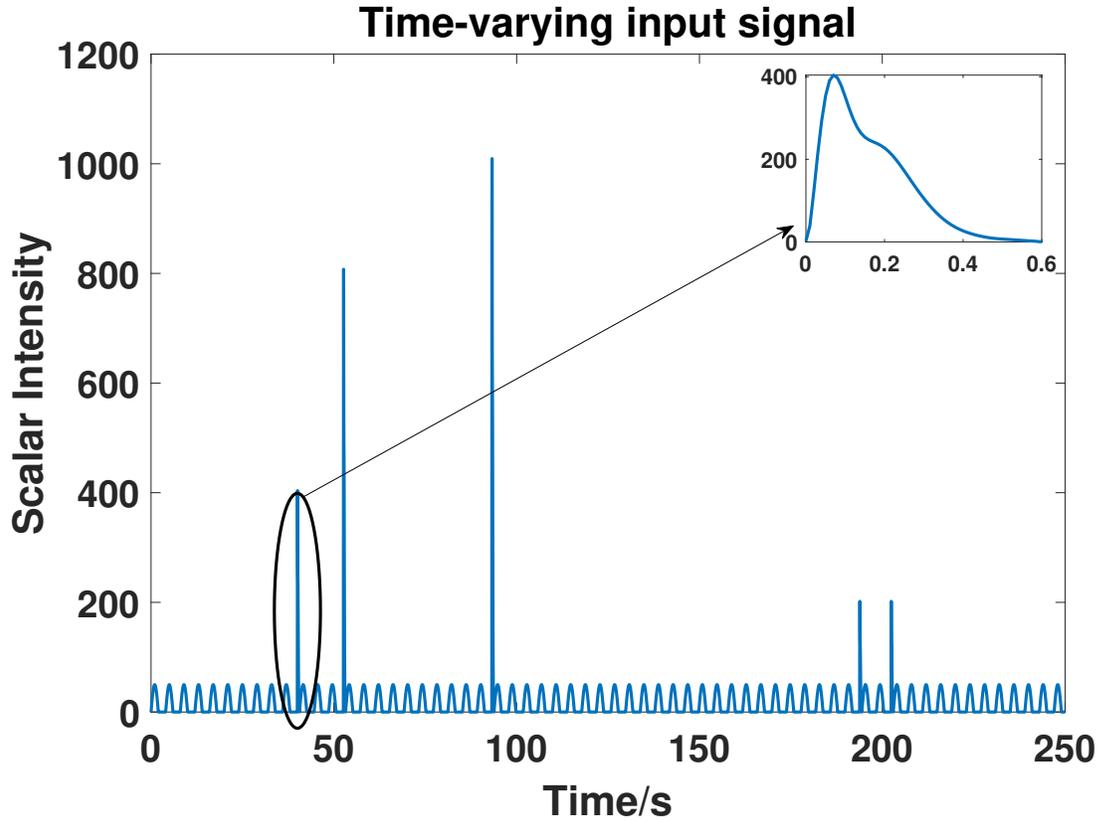


Figure 5.23: Source modulation signal  $u(t)$  used in the numerical experiment.

The infectivity field  $C$  is monitored in front of students 4, 5, 6 and 8, which are identified as ‘high risk’ for exposure to aerosol particles. As shown in Figure 5.24, the optimal frequency-shift ROM closely follows the full-order model solution. For students 4,6 and 8 located at a significant distance from the source, the infectious aerosol particles starts accumulate after approximately 100 seconds. The data clearly illustrate the potential of infection for these students. We note that small deviations from the FOM results can be observed during the transient process. The deviations are caused by different source of errors including, the numerical error of matrix manipulations, full-order model convergence, and imperfect linear independence of vectors in the Krylov subspace. We carefully operate the ROM generation in order to eliminate these factors and produce the optimal ROM. The results, as oen can see in Fig. 5.24, are reasonably accurate.

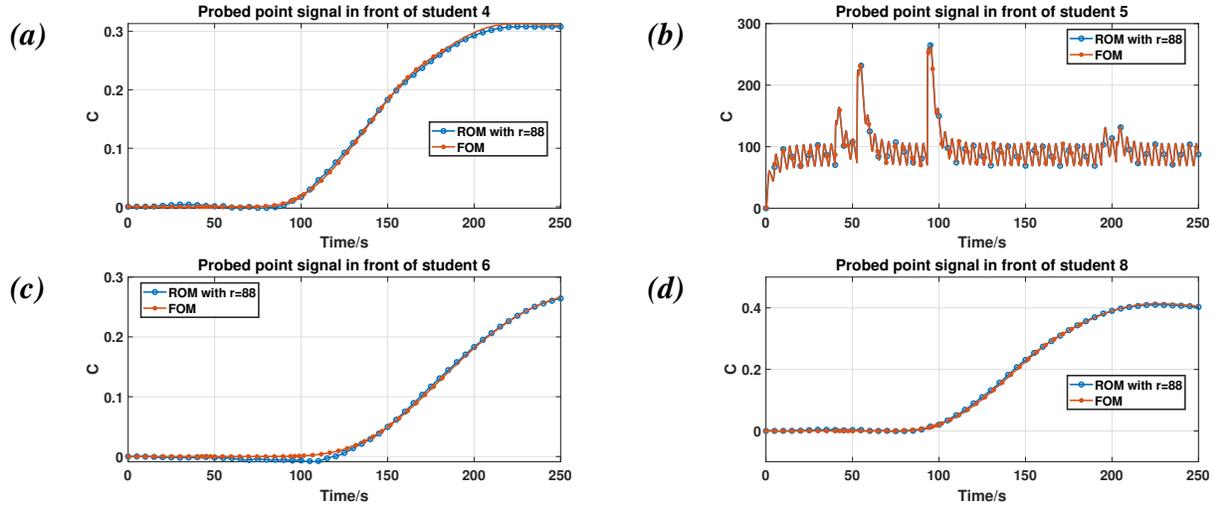


Figure 5.24: Transient scalar signals, under the input of 5.23, at the probed points in front of students whose frontal space are identified as 'high-risk' area.

The distribution of the infectivity field in the cross-section  $z = 4.5m$  at different time moments are presented in Fig. 5.25. We see that aerosol starts to diffuse around the space for the first 50 seconds. Its transport is dominated by the forced convection by turbulent flow at later time. The transport of the aerosol particle is accelerated by the ventilation which enable it to travel to a larger distances (see Fig. .5.25 e).

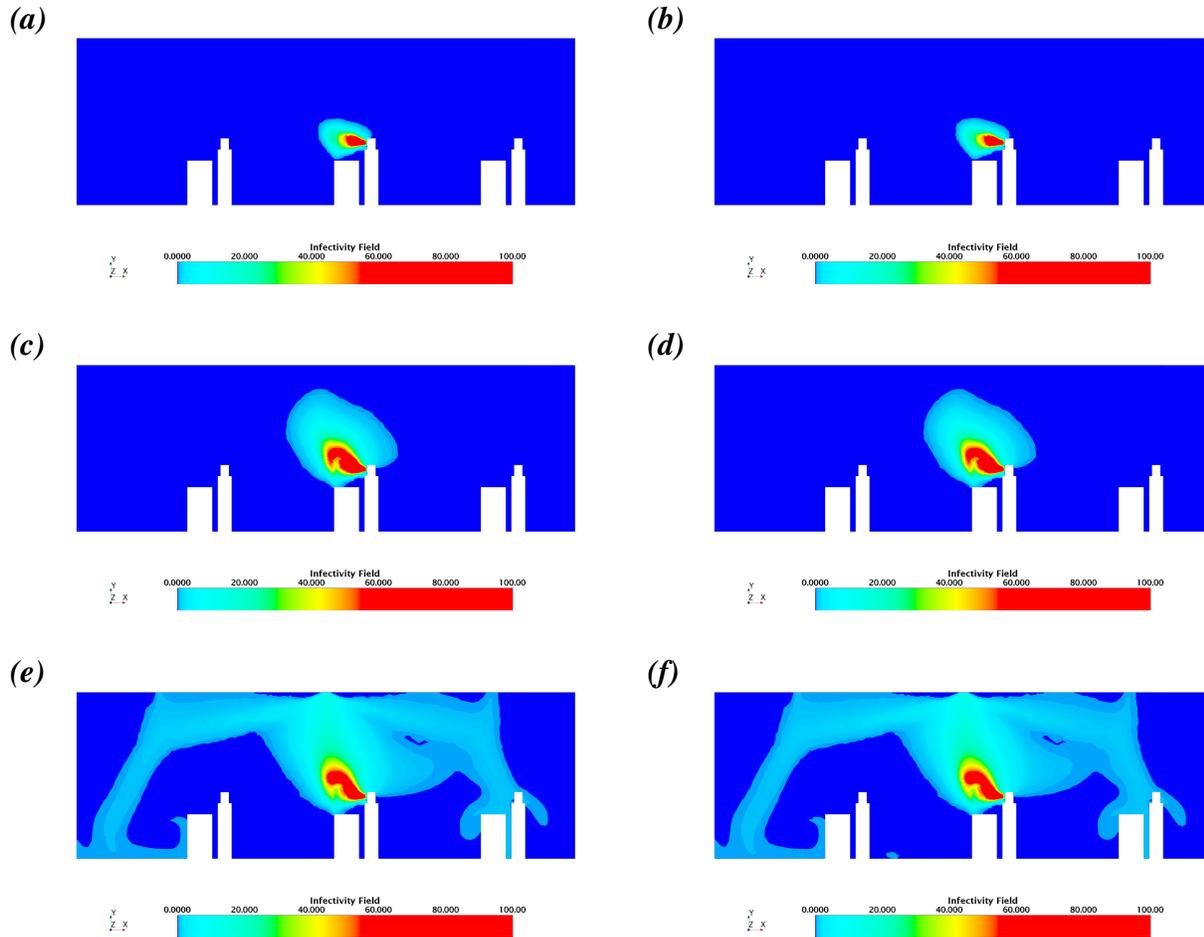


Figure 5.25: Cross section at  $t=10, 50, 220$ s of ROM at (a), (c), (e) and FOM at (b), (d), (f).

We use the line probes across the middle of classroom at  $y=1.3$  and  $2.5$  m to examine the quality of the ROM approximation. The results are plotted at different times. As shown in Figure 5.26, the ROM can capture the spatial distribution of the infectivity field when the transport of the aerosol is dominated by convection. However, at the early stage of transport, e.g.,  $t=10$ s and  $50$ s as shown in Figure 5.26 (a) and (b), some inaccuracies can be observed. This can be attributed to very low (nearly zero) value of  $C$  in most of the room at this time. The ROM produces inaccurate results in such situations, although the absolute value of incorrectly predicted  $C$  is typically low, so the error does not have significant practical consequences.

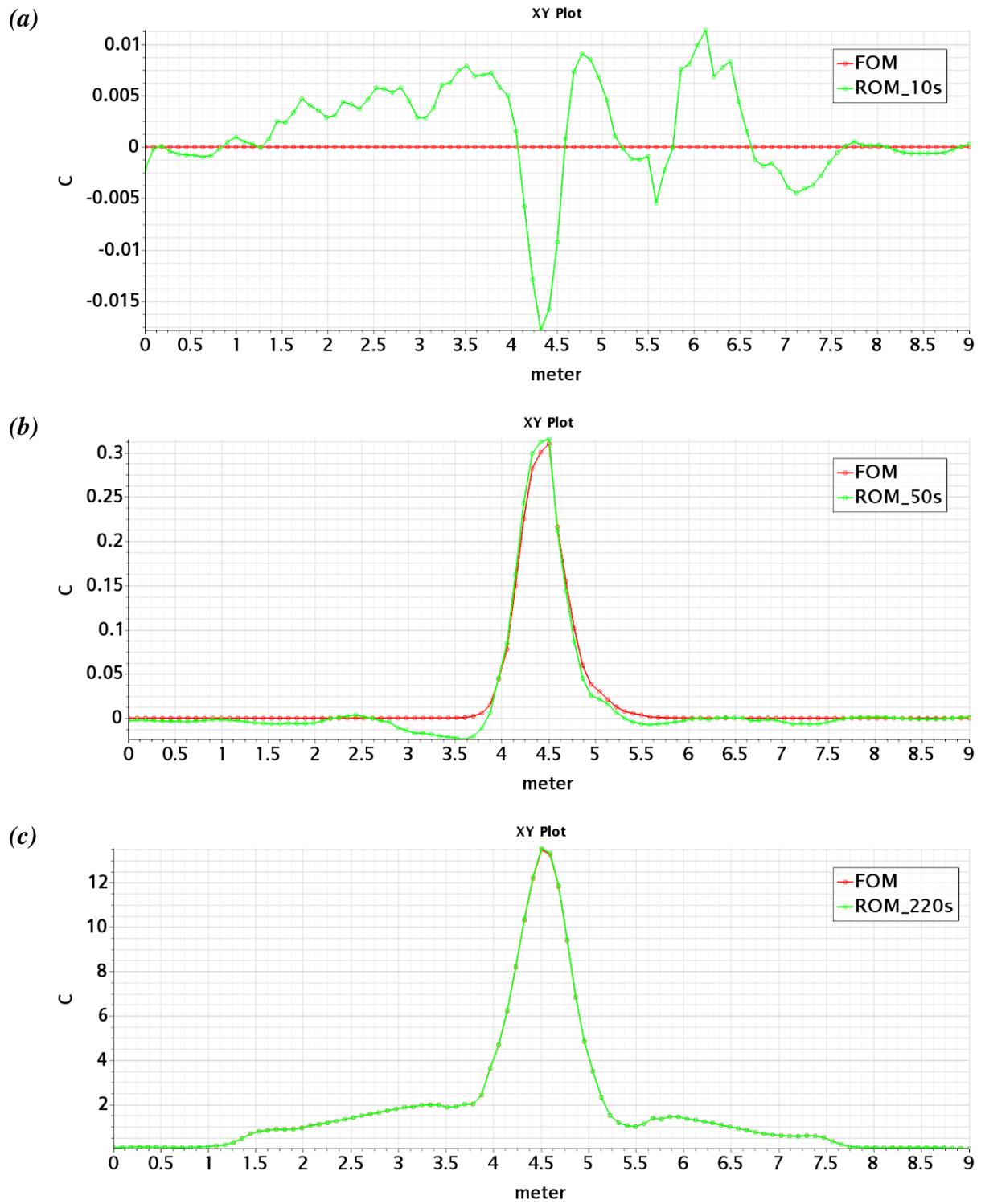


Figure 5.26: Probe line signal of (a)  $t=10s$ , (b)  $50s$  and (c)  $220s$  at  $y=2.5$  m (elevation)

The accuracy of the ROM prediction is much better along the probe line at  $y=1.3\text{m}$  (see Fig. 5.27). At all three time moments, the ROM data are practically indistinguishable from those of FOM.

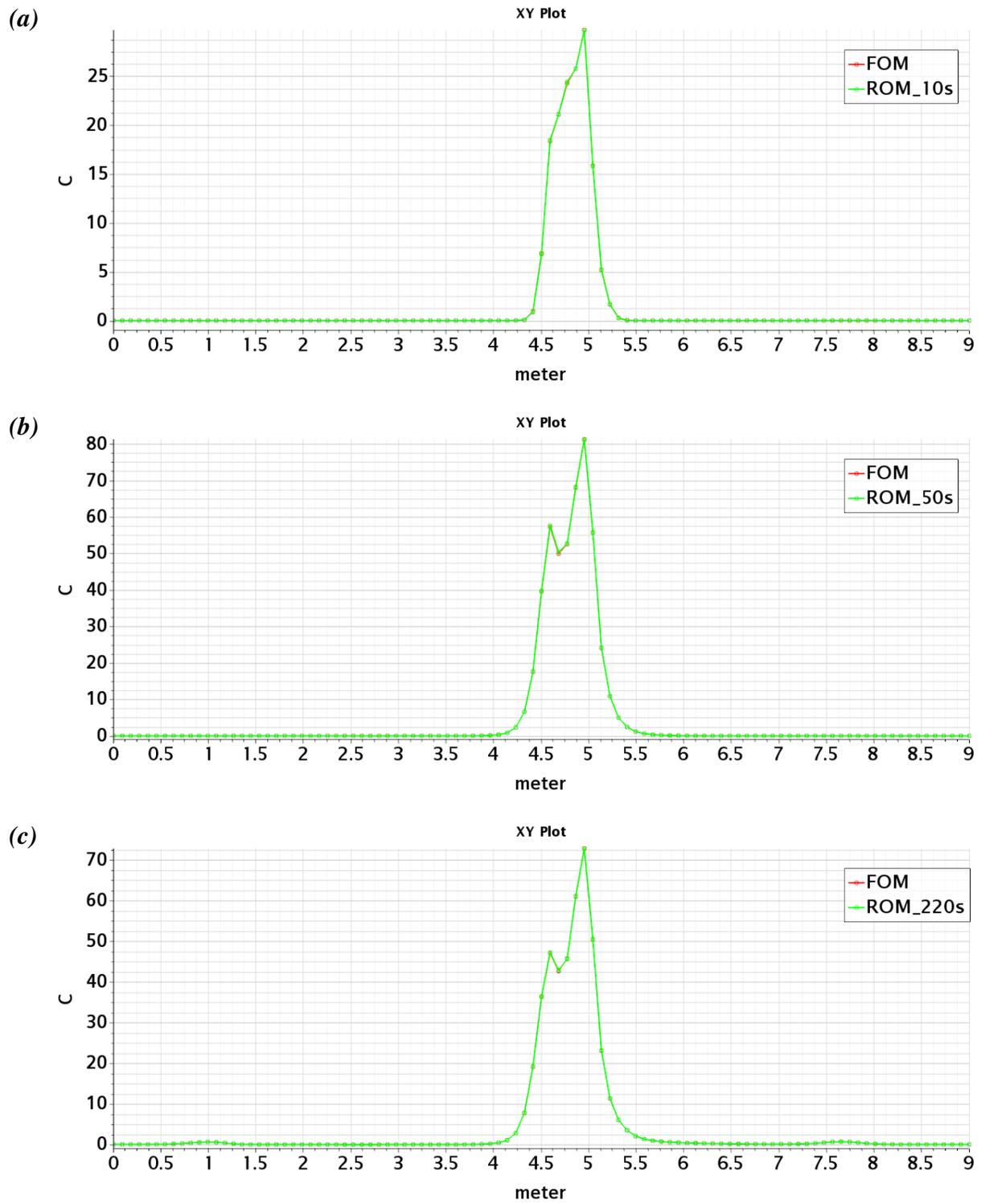


Figure 5.27: Probe line signal of  $t=10s$  (a),  $50s$  (b) and  $220s$ (c) at  $y=1.3$  m (elevation)

Finally, the accuracy of the ROM approximation is evaluated through norms of error vectors at different time moments as shown in Table 5.6. It is observed that the maximum deviation between the FOM and ROM fields does not exceed 4 at  $t=220$ s. This indicates good accuracy of ROM in predicting the response to coughing activities (high-frequency response). Note that the maximum value of  $C$  at 220s is 184. Therefore, the ROM poses a relative error of  $\sim 2.2\%$ . In this case, the ROM presents a reasonable approximation with respect to the dynamics of FOM.

Table 5.6: The norms at  $t=10, 50$  and  $220$ s

Time/s	Norms	
	$\ \epsilon\ _{\infty}$	$\ \epsilon\ _2$
10	2.5974	1.74E-05
50	3.0363	1.12E-05
220	4.0385	1.99E-05

### 5.3.3 Time-Varying Process Predicted by ROM

At this point, the accuracy of the ROM has been confirmed by validation test conduction for the first 250s with ROM with  $r=88$  in the previous section. We now apply the ROM with  $r=88$  to predict the aerosol transmission in the same classroom after  $t=250$ s in order to assess the risk of infection. This will illustrate the ability of ROM to rapidly predict and quantify the risk level in each area of the classroom.

We consider the spread of infectivity field in the classroom during a lecture lasting 45 minutes. We assume the student 5 is the source person while the other persons in the classroom are the potential recipients of the infection. We apply ROM to predict and analyze the aerosol transmission for the 45-min transient process in order to determine the area that is highly likely to be exposed to infectious particles.

We assume that the classroom is ventilated before the class, so value of  $C$  is uniformly 0 in the classroom. The time modulation  $u(t)$  of the source shown in Fig. 5.28 is utilized. We assume that

the human source has a steady breathing cycle during the trial, with brief intermittent coughing occurring from time to time. The whole process is a transient process with a duration of 2700s (45 minutes) as shown in Figure 5.28.

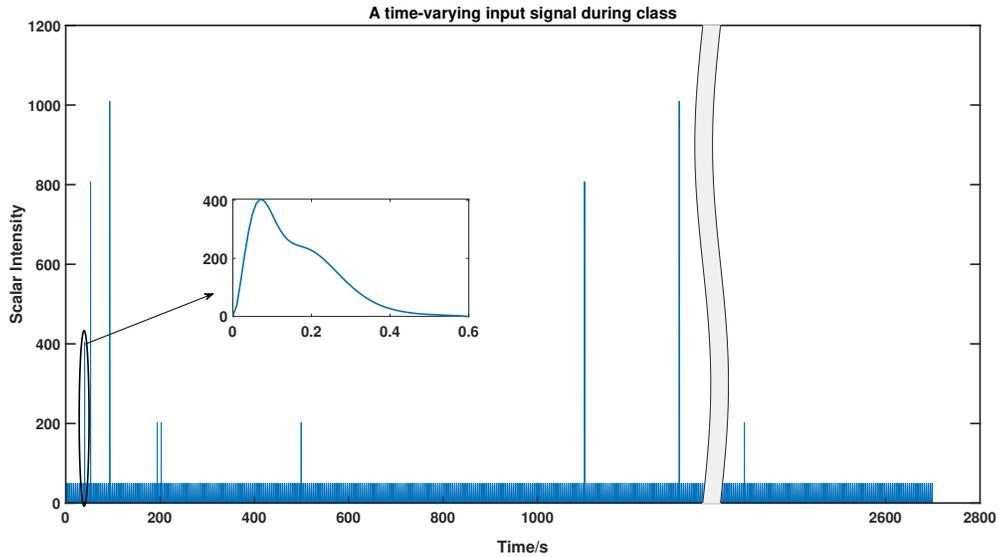


Figure 5.28: Time-varying source modulation signal  $u(t)$  in simulating a lecture.

Simulation of the process using the FOM is estimated to have taken 7200 core-hours to complete. Even a high-performance workstation would need at least half a month for parallel computing on 20 cores to complete the task. It would be a time consuming and inefficient process especially if a large number of parametric studies were needed. We apply the ROM with  $r=88$  to simulate the same process. The task is finished using a single core on a personal desktop within 90 minutes. This reduces the computation effort by approximately 4800 times. As a result, ROM can substantially reduce the computation time while maintain the accurate results.

### 5.3.4 Risk Assessment Using ROM Data

The values at the probed points are shown in Figure 5.29. The value for index patient, student 5, is represented on the left axis, while the data for the other are represented on the right axis. We see that the infectivity field is firstly transported into the frontal area of student 4,6 and 8. As

shown in Figure 5.29, the passive scalar signal in front of student 4, 6 and 8 grows around 100s. As time goes further beyond 250s, the values for other 5 students start to grow. It is found that the magnitude of C for students 4, 6 and 8 are greater than for the others. Therefore, we define the locations of students 4, 6 and 8 as ‘high-risk’ zones where infections are more likely. Interestingly, the signals at these three points fluctuate some time after the source signal fluctuates. However, when the peak magnitude of coughing is not severe, e.g., with peak value of 200 at  $\sim t=2300s$ , the fluctuation is not reflected in the signals in front of students 4, 6 and 8. Compare with that, the other five signal sources will not fluctuate with the fluctuation of the signal source, and the average value is relatively small. Therefore, we define the positions of five students other than students 4, 6, and 8 as low-risk locations.

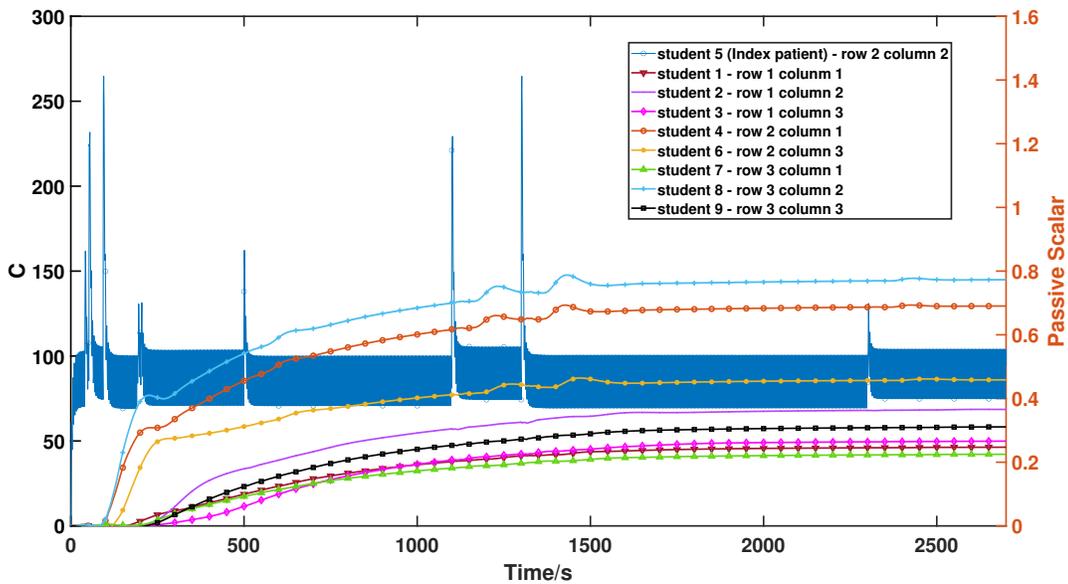


Figure 5.29: Concentrations of infectivity field C at probe points in front of students under the input signal of 5.28. Left axis data for student 5; Right axis-data for the other students.

Among the entire classroom, the passive scalar field mainly transported in the positions of frontal box of student 4 and 8 as indicated by the probed points’ signals. We further illustrate that in Fig. 5.30. The iso-surface of the passive scalar with iso-value of 0.8 shows that the area of aerosol cloud distributes along the outflow of ventilation systems. The airflow of ventilation

transports the aerosol from student 5 to the location of student 4, 6 and 8 respectively.

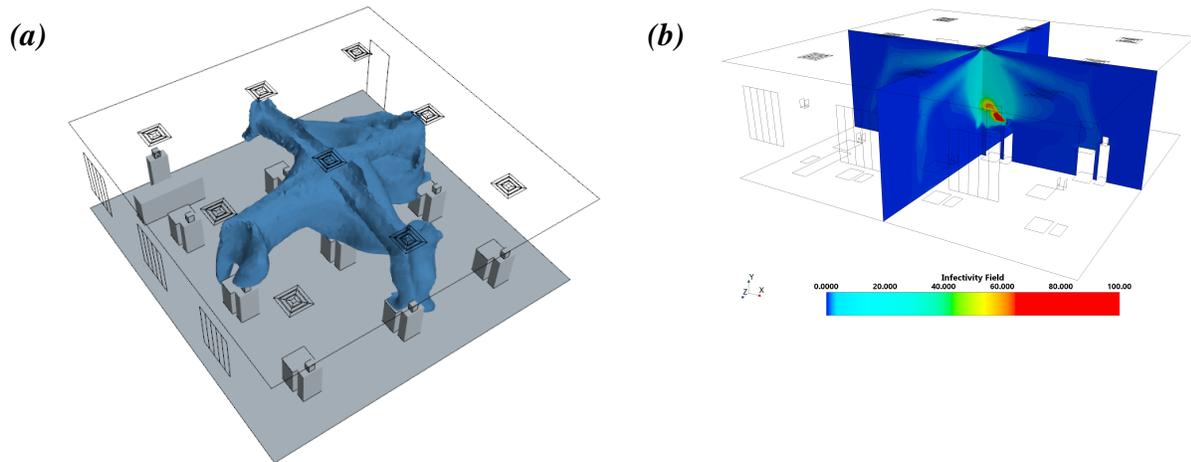


Figure 5.30: At  $t=2000s$ , (a) Iso-surface of passive scalar with iso-value of 0.8; (b) Cross-section of C.

We integrate the passive scalar concentration over volume of box in front of each student at  $t=2000s$ . The box is defined as a rectangular box in front of students as shown in Figure 5.31 in pink and thought of as an approximation of the zone of air inhaled by each student.

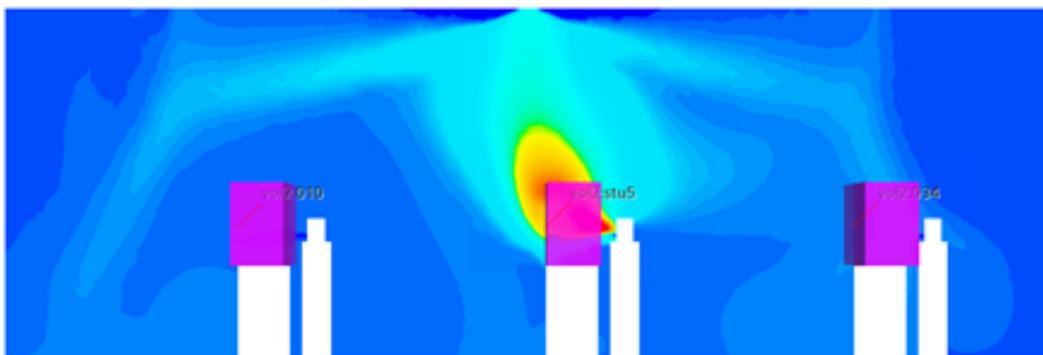


Figure 5.31: Integration area in front of students.

As indicated by the data of integration shown in Table 5.7 and chart in Fig. 5.32, amounts of infectious aerosol particles inhaled by students 4, 6, and 8 are relatively high.

Table 5.7: Volume integration of passive scalar concentration over boxes in front of students at  $t=2000s$

Student	1	2	3	4	5
Volume Integration	6.61E-02	9.38E-02	6.20E-02	1.25E-01	4.25E+00
Student	6	7	8	9	
Volume Integration	1.33E-01	5.18E-02	1.50E-01	6.99E-02	

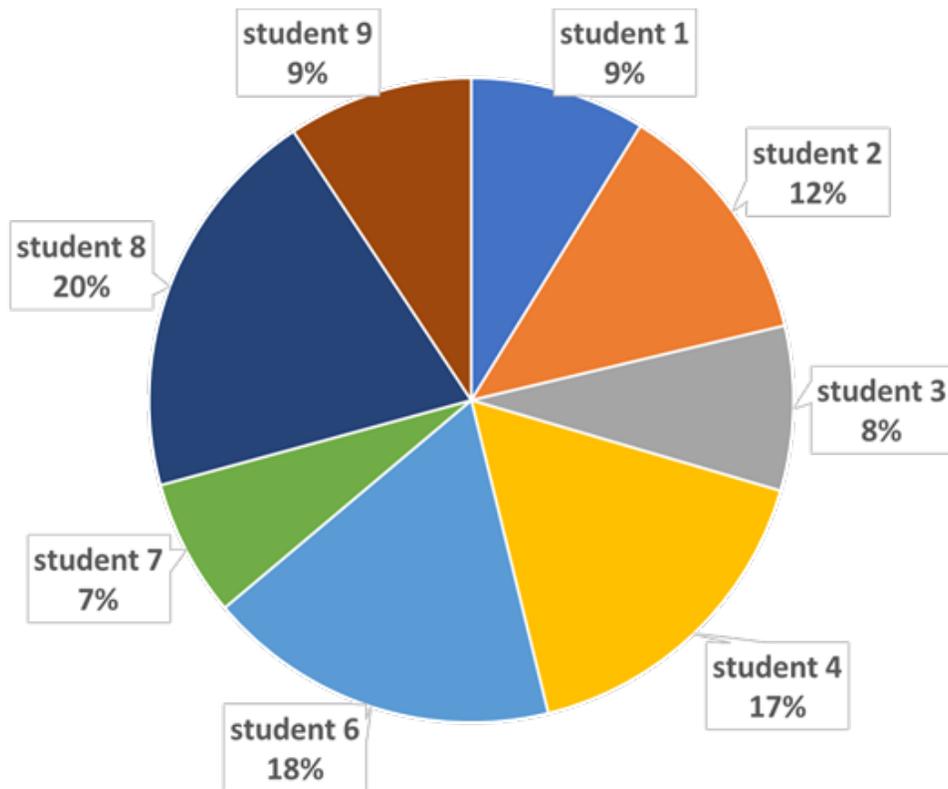


Figure 5.32: The data for student 5 (the source) are excluded.

## 5.4 Conclusion

In the current study, the reduced order modeling using Krylov subspace methods is discussed which allows us to simulate the transport of aerosol particles as passive scalar in a ventilation indoor environments. The Arnoldi algorithm-based SIMO algorithm and an Arnoldi based variant,

frequency-shift algorithm, are evaluated.

The aerosol infectivity field is simulated using the Eulerian approach as the concentration of the passive scalar, transport of which is determined by an advection-diffusion problem with constant background turbulent velocity fields. Through comparative study between the Eulerian and Lagrangian approaches, it has been found that the predictions of aerosol particles' transmission are very similar between the two approaches.

To the best of our knowledge, this study is the first, in which the Krylov-based reduced-order modeling approach is applied to indoor ventilation system. Different Arnoldi-based algorithms constructing the reduced order modeling are applied and evaluated. The SIMO algorithm that has record of success in the analysis of the pollutant transmission with external turbulent flow [29] and conjugate heat transfer in battery thermal management system [61] is evaluated first. We further develop a frequency-shift Arnoldi algorithm to improve the performance of ROM. We find that the frequency-shift Arnoldi algorithm substantially improves the accuracy of the approximation. The frequency-shift Arnoldi algorithm can capture the high and low frequency response of the system depending on the selection of the frequency points. Additionally, a discretized time derivative method is introduced to indirectly access the system matrix  $A$  which is normally inaccessible from the commercial CFD software.

According to the comparative analysis and numerical experiments, ROM can greatly save the computational cost required for the analysis of transient processes. Once the ROM is built, the user can perform extensive numerical exploration through the ROM, i.e., apply different transient exhaling scenarios (breathing, coughing, singing etc) to the model. A full-order model requires a supercomputer to utilize parallel computing for several days of computational tasks, while the ROM only takes a few seconds to get results on a single core to predict the spread of aerosols. It has been found that the simulation of aerosol transmission using ROM can accelerate the simulation speed up to 4800 times in comparison with the FOM simulation.

Admittedly, some unphysical fluctuations of scalar are found in the fields produced by ROMs. We identify the fluctuations as the limitation of the ROMs based on the existing algorithms. The

limitation comes from the FOM system's characteristics, i.e., indoor ventilation system with circulation flow, where the transport is dominated by convection.

Finally, the quantitative studies with parametric simulation can be achieved by ROM. The time-varying passive scalar input signals can be presumed and applied to optimized ROM in order to have large number of quantitative studies within a short period. The proposed approaches can potentially be applied to a broad range of scenarios, including a cabin of a commercial airplane, a movie theater, or a library. The efficient yet accurate predictions of aerosol transmission can be simulated by ROMs. The price one pays is only the computation effort in generating the ROM. A large number of rapid results from ROMs can effectively help policy makers or designer to predict/analyze the aerosol transmission of the aerosol particle. The high-risk areas in public places due to aerosol transmission can be identified. As a result, public places can be arranged and designed in advance, thereby reducing the risk of aerosol transmission in public places.

## CHAPTER 6

### Conclusion and Future Works

Large-scale linear systems are often used to predict and simulate the behavior of a system given various sets of parameters. In this dissertation, the work focus is on developing an efficient reduced-order model that can approximate the original model and allow one to accurately predict its dynamics at greatly reduced computation effort. Applications in different areas show the practical potential of reduced-order modeling. For example, the method can be applied to online predictive control of the BTMS as presented in Chapter 4 and to prediction of aerosol transmission in indoor environments under various exhaling behavior for a long period of time as presented in Chapter 5. Various techniques have been proposed in the study of reduced-order modeling, which was reviewed in Chapter 1. However, the system matrices are very large and, in general, unavailable from the commercial CFD software, which makes most of the reduced-order modeling methodologies inapplicable from the perspectives of unaffordable computation cost or require of access to the system matrices.

In the course of the dissertation, several algorithms for generation of reduced-order models of large-scale linear systems are presented in Chapter 3. This includes the single-input algorithm, multiple-input algorithm, frequency-shift algorithm, and parametric reduced order modeling algorithm. The algorithms are tested in application to modeling conjugate heat transfer in a battery pack in Chapter 4. Direct ROM-FOM comparison shows high accuracy at the computation costs reduced by many orders of magnitude.

As presented in Chapter 4, further investigation based on the Arnoldi-type algorithm has been

presented. The multi-input multi-output(MIMO) is developed in order to extend the applicability of the Arnoldi-type algorithm to multi-input scenarios. The parametric reduced-order modeling algorithm shows the potential of adapting and interpolating the pre-computed local ROMs to new parametric ROMs that allow the reduction of the large-scale linear system within a range of the parameter. Last but not least, change of variable and boundary conditions as input methods have been introduced and examined in a battery pack.

As presented in Chapter 5, a comparative study demonstrates the quantitatively similar result that can be achieved between the Eulerian and Lagrangian frameworks. The single-input Arnoldi algorithm is examined in the modeling of airborne transmission of respiratory infections. The frequency-shift Arnoldi algorithm is developed and shows proof of success in accurately simulating the airborne transmission of respiratory infection.

Our last comment concerns the range of applicability of the proposed method. Its accuracy and efficiency have been proven in this work only for the cases of conjugate heat transfer in a battery pack and the prediction of airborne transmission of respiratory infection in indoor environments. We do not, however, foresee any major obstacles precluding the use of the method for other situations in a much wide range of diffusive and convective transport phenomena in technology and nature.

Several topics are found during the course of the dissertation that are worth further investigation including:

- Investigation of the parametric reduced-order modeling for systems with time-dependent parameters (such as, e.g., mass flow rate of the inlet for battery thermal management systems);
- Investigation of the stability of the ROM generated by Arnoldi-type Krylov method;
- Development of ROM for indoor environments with different scenarios, including MIMO system, parametric system and time-varying systems.

## BIBLIOGRAPHY

- [1] M. Abuhegazy, K. Talaat, O. Anderoglu, and S. V. Poroseva. Numerical investigation of aerosol transport in a classroom with relevance to covid-19. *Physics of Fluids*, 32(10):103311, 2020.
- [2] T. M. Bandhauer, S. Garimella, and T. F. Fuller. A critical review of thermal issues in lithium-ion batteries. *J. Electrochem. Soc.*, 158(3):R1–R25, 2011.
- [3] S. Al Hallaj, H. Maleki, J. S. Hong, and J. R. Selman. Thermal modeling and design considerations of lithium-ion batteries. *J. Power Sources*, 83(1-2):1–8, 1999.
- [4] L. H. Saw, Y. Ye, A. A. O. Tay, W. T. Chong, S. H. Kuan, and M. C. Yew. Computational fluid dynamic and thermal analysis of lithium-ion battery pack with air cooling. *Appl. Energy*, 177:783–792, 2016.
- [5] S. Basu, K. S. Hariharan, S. M. Kolake, T. Song, D. K. Sohn, and T. Yeo. Coupled electrochemical thermal modelling of a novel li-ion battery pack thermal management system. *Appl. Energy*, 181:1–13, 2016.
- [6] E. Meyerowitz and A. Richterman. SARS-CoV-2 Transmission and Prevention in the Era of the Delta Variant. *Infect Dis Clin North Am.*, 2021.
- [7] R. Wang, J. Chen, and G.-W. Wei. Mechanisms of SARS-CoV-2 evolution revealing vaccine-resistant mutations in Europe and America. *The journal of physical chemistry letters*, 12(49):11850–11857, 2021.
- [8] J. Park and G. Kim. Risk of CoViD-19 infection in public transportation: the development of a model. *International journal of environmental research and public health*, 18(23):12790, 2021.
- [9] J. Pauser, C. Schwarz, J. Morgan, J. Jantsch, and M. Brem. SARS-CoV-2 transmission during an indoor professional sporting event. *Scientific Reports*, 11(1):1–5, 2021.
- [10] V. Vuorinen, M. Aarnio, M. Alava, V. Alopaeus, N. Atanasova, M. Auvinen, N. Balasubramanian, H. Bordbar, P. Erästö, R. Grande, et al. Modelling aerosol transport and virus exposure with numerical simulations in relation to SARS-CoV-2 transmission by inhalation indoors. *Saf. Sci.*, 130:104866, 2020.
- [11] W. W. Nazaroff. Indoor aerosol science aspects of SARS-CoV-2 transmission. *Indoor air*, 32(1):e12970, 2022.

- [12] M. P. Wan, G. N. Sze To, C. Y. H. Chao, L. Fang, and A. Melikov. Modeling the fate of expiratory aerosols and the associated infection risk in an aircraft cabin environment. *Aerosol Sci. Technol.*, 43(4):322–343, 2009.
- [13] G. Bagheri, O. Schlenczek, L. Turco, B. Thiede, K. Stieger, J.-M. Kosub, M. L. Pöhlker, C. Pöhlker, J. Moláček, S. Scheithauer, et al. Exhaled particles from nanometre to millimetre and their origin in the human respiratory tract. *medRxiv*, 2021.
- [14] Z. Wang, E. R. Galea, A. Grandison, J. Ewer, and F. Jia. A coupled computational fluid dynamics and wells-riley model to predict covid-19 infection probability for passengers on long-distance trains. *Safety science*, 147:105572, 2022.
- [15] D. S. Thatiparti, U. Ghia, and K. R. Mead. Computational fluid dynamics study on the influence of an alternate ventilation configuration on the possible flow path of infectious cough aerosols in a mock airborne infection isolation room. *Sci. Technol. Built Environ.*, 23(2):355–366, 2017.
- [16] C. Xu, W. Liu, X. Luo, X. Huang, and P. V. Nielsen. Prediction and control of aerosol transmission of SARS-CoV-2 in ventilated context: from source to receptor. *Sustainable cities and society*, 76:103416, 2022.
- [17] M. Muratori, M. Canova, Y. Guezennec, and G. Rizzoni. A reduced-order model for the thermal dynamics of Li-ion battery cells. In *Proc. 6th IFAC Symp. Adv. Automotive Control*, volume 43, pages 192–197, 2010.
- [18] F. He, A. A. Ams, Y. Roosien, W. Tao, B. Geist, and K. Singh. Reduced-order thermal modeling of liquid-cooled lithium-ion battery pack for evs and hev. In *2017 IEEE Transportation Electrification Conference and Expo (ITEC)*, pages 507–511, 2017.
- [19] F. He and L. Ma. Thermal management of batteries employing active temperature control and reciprocating cooling flow. *Int. J. Heat Mass Transf.*, 83:164–172, 2015.
- [20] X. Hu, S. Asgari, I. Yavuz, S. Stanton, C. C. Hsu, Z. Shi, B. Wang, and H. K. Chu. A transient reduced order model for battery thermal management based on singular value decomposition. In *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*, pages 3971–3976. IEEE, 2014.
- [21] X. Hu, S. Krishnaswamy, S. Asgari, and S. Stanton. An efficient transient thermal model for electronics thermal management based on singular value decomposition. In *2015 31st Thermal Measurement, Modeling Management Symposium (SEMI-THERM)*, pages 280–286. IEEE, 2015.
- [22] A. C. Antoulas and D. C. Sorensen. Approximation of large-scale dynamical systems: an overview. *Int. J. Appl. Math. Comput. Sci.*, 11(5):1093–1121, 2001.
- [23] A. C. Antoulas, D. C. Sorensen, and S. Gugercin. A survey of model reduction methods for large-scale systems. *Contemp. Math.*, 280:193–219, 2001.

- [24] U. Baur, P. Benner, and L. Feng. Model order reduction for linear and nonlinear systems: a system-theoretic perspective. *Arch. Comput. Methods Eng.*, 21(4):331–358, 2014.
- [25] P. Benner, V. Mehrmann, and D. C. Sorensen. *Dimension reduction of large-scale systems*, volume 45 of *Lect. Notes in Comp. Sci. Eng.* Springer, 2005.
- [26] R. W. Freund. Model reduction methods based on Krylov subspaces. *Acta Numerica*, 12:267–319, 2003.
- [27] B. Salimbahrami, B. Lohmann, T. Bechtold, and J. G. Korvink. Two-sided arnoldi algorithm and its application in order reduction of mems. In *Proc. 4th MATHMOD*, pages 1021–1028, 2003.
- [28] Y. Wang, H. Song, and K. Pant. A reduced-order model for whole-chip thermal analysis of microfluidic lab-on-a-chip systems. *Microfluid. Nanofluidics*, 16(1-2):369–380, 2014.
- [29] L. Vervecken, J. Camps, and J. Meyers. Stable reduced-order models for pollutant dispersion in the built environment. *Build. Environ.*, 92:360–367, 2015.
- [30] R. S. Puri and D. Morrey. A Krylov–Arnoldi reduced order modelling framework for efficient, fully coupled, structural–acoustic optimization. *Struct. Multidiscipl. Optim.*, 43(4):495–517, 2011.
- [31] K. Willcox, J. Peraire, and J. White. An Arnoldi approach for generation of reduced-order models for turbomachinery. *Comput. Fluids*, 31(3):369–389, 2002.
- [32] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Nat. Bur. Stand.*, 45(4):255–282, 1950.
- [33] Z. Bai, P. Feldmann, and R. W. Freund. Stable and passive reduced-order models based on partial Padé approximation via the Lanczos process. *Numerical Analysis Manuscript*, 97(3):10, 1997.
- [34] R. W. Freund. Krylov-subspace methods for reduced-order modeling in circuit simulation. *J. Comput. Appl. Math.*, 123(1-2):395–421, 2000.
- [35] ANSYS. Inc. *ANSYS FLUENT User’s Guide*. Canonsburg, PA, 17.2 edition, 2016.
- [36] Siemens PLM Software. *STAR-CCM+ User’s Guide*. Plano, Texas, 13.04.011-R8 edition, 2019.
- [37] J. Cullum and T. Zhang. Two-sided Arnoldi and nonsymmetric Lanczos algorithms. *SIAM J. Matrix Anal. Appl.*, 24(2):303–319, 2002.
- [38] B. Salimbahrami, B. Lohmann, T. Bechtold, and J. G. Korvink. A two-sided Arnoldi algorithm with stopping criterion and MIMO selection procedure. *Math. Comput. Model Dyn. Syst.*, 11(1):79–93, 2005.
- [39] C. C. Wang, K. A. Prather, J. Sznitman, J. L. Jimenez, S. S. Lakdawala, Z. Tufekci, and L. C. Marr. Airborne transmission of respiratory viruses. *Science*, 373(6558):eabd9149, 2021.

- [40] W. Garcia, S. Mendez, B. Fray, and A. Nicolas. Model-based assessment of the risks of viral transmission in non-confined crowds. *Safety Science*, 144:105453, 2021.
- [41] P. J. Bueno de Mesquita, W. W. Delp, W. R. Chan, W. P. Bahnfleth, and B. C. Singer. Control of airborne infectious disease in buildings: Evidence and research priorities. *Indoor air*, 32(1):e12965, 2022.
- [42] C.-H. Chien, M.-D. Cheng, P. Im, K. Nawaz, B. Fricke, and A. Armstrong. Characterization of the indoor near-field aerosol transmission in a model commercial office building. *International Communications in Heat and Mass Transfer*, 130:105745, 2022.
- [43] S. Asadi, N. Bouvier, A. S. Wexler, and W. D. Ristenpart. The coronavirus pandemic and aerosols: Does covid-19 transmit via expiratory particles? *Aerosol Sci. Technol.*, 45(6):635–638, 2020.
- [44] J. K. Gupta, C. H. Lin, and Q. Chen. Flow dynamics and characterization of a cough. *Indoor air*, 19(6):517–525, 2009.
- [45] Z. Y. Han, W. G. Weng, and Q. Y. Huang. Characterizations of particle size distribution of the droplets exhaled by sneeze. *J. R. Soc. Interface*, 10(88):20130560, 2013.
- [46] W. F. Wells. On air-borne infection: study ii. droplets and droplet nuclei. *American journal of Epidemiology*, 20(3):611–618, 1934.
- [47] L. Bourouiba, E. Dehandschoewercker, and J. W.M. Bush. Violent expiratory events: on coughing and sneezing. *J. Fluid Mech.*, 745:537–563, 2014.
- [48] G. Buonanno, L. Morawska, and L. Stabile. Quantitative assessment of the risk of airborne transmission of SARS-CoV-2 infection: prospective and retrospective applications. *Environ. Int.*, 145:106112, 2020.
- [49] N. Van Doremalen, T. Bushmaker, D. H. Morris, M. G. Holbrook, A. Gamble, B. N. Williamson, A. Tamin, J. L. Harcourt, N. J. Thornburg, S. I. Gerber, et al. Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1. *N. Engl. J. Med.*, 382(16):1564–1567, 2020.
- [50] N. L. Ribaric, C. Vincent, G. Jonitz, A. Hellinger, and G. Ribaric. Hidden hazards of SARS-CoV-2 transmission in hospitals: A systematic review. *Indoor air*, 32(1):e12968, 2022.
- [51] S. Shao, D. Zhou, R. He, J. Li, S. Zou, K. Mallery, S. Kumar, S. Yang, and J. Hong. Risk assessment of airborne transmission of covid-19 by asymptomatic individuals under different practical settings. *Journal of aerosol science*, 151:105661, 2021.
- [52] O. Zikanov. *Essential Computational Fluid Dynamics*. John Wiley & Sons, Hoboken, NJ, second edition, 2019.
- [53] M. Goodson, J. Feaster, A. Jones, G. McGowan, L. Agricola, W. Timms, and M. Uddin. Modeling Transport of SARS-CoV-2 Inside a Charlotte Area Transit System (CATS) Bus. *Fluids*, 7(2):80, 2022.

- [54] A. Bayram and A. Korobenko. Modelling the transport of expelled cough particles using an Eulerian approach and the variational multiscale method. *Atmospheric Environment*, 271:118857, 2022.
- [55] E. Lee, C. E. Feigley, and J. Khan. An investigation of air inlet velocity in simulating the dispersion of indoor contaminants via computational fluid dynamics. *Annals of occupational hygiene*, 46(8):701–712, 2002.
- [56] D. Bernardi, E. Pawlikowski, and J. Newman. A general energy balance for battery systems. *J. Electrochem. Soc.*, 132(1):5, 1985.
- [57] D. D. Gray and A. Giorgini. The validity of the Boussinesq approximation for liquids and gases. *Int. J. Heat Mass Transf.*, 19(5):545–551, 1976.
- [58] C. Y. H. Chao, M. P. Wan, L. Morawska, G. R. Johnson, Z.D. Ristovski, M. Hargreaves, K. Mengersen, S. Corbett, Y. Li, X. Xie, et al. Characterization of expiration air jets and droplet size distributions immediately at the mouth opening. *Journal of aerosol science*, 40(2):122–133, 2009.
- [59] Z. Bai. Motivations and realizations of Krylov subspace methods for large sparse linear systems. *J. Comput. Appl. Math.*, 283:71–78, 2015.
- [60] Peuscher, H. and Mohring, J. and Eid, R. and Lohmann, B. Parametric model order reduction by matrix interpolation. *Automatisierungstechnik*, 58:475–484, 08 2010.
- [61] L. Xiang, C. W. Lee, O. Zikanov, and C.-C. Hsu. Efficient reduced order model for heat transfer in a battery pack of an electric vehicle. *Appl. Therm. Eng.*, Submitted, 2020.
- [62] H. Lundgren, P. Svens, H. Ekström, C. Tengstedt, J. Lindström, M. Behm, and G. Lindbergh. Thermal management of large-format prismatic lithium-ion battery in PHEV application. *J. Electrochem. Soc.*, 163:A309, 2016.