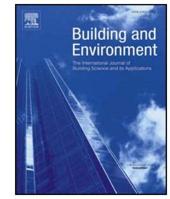
Building and Environment 126 (2017) 184–194

Contents lists available at ScienceDirect

Building and Environment

journal homepage: www.elsevier.com/locate/buildenv



A state-space method for real-time transient simulation of indoor airflow

Qiujian Wang[a], Yiqun Pan[a,*], Mingya Zhu[a], Zhizhong Huang[a], Wei Tian[b], Wangda Zuo[c], Xu Han[c], Peng Xu[a]

[a] Tongji University, Shanghai, China
[b] University of Miami, Coral Gables, FL, USA
[c] University of Colorado, Boulder, CO, USA

ARTICLE INFO

Keywords:
CFD
FFD
State-space fluid dynamics (SFD)
Transient air flow simulation
Real-time simulation

ABSTRACT

Inhomogeneous airflow distribution is common in air-conditioned rooms, especially the large open spaces. To evaluate the thermal comfort of such space, or the control performance of the Heating, Ventilation, and Air Conditioning (HVAC) systems in an efficient way, one will need a fast prediction method to simulate the airflow and temperature distribution. This paper proposes a discrete state-space method, called state-space fluid dynamics (SFD), for the fast indoor airflow simulation. To handle time-varying velocity and temperature field, SFD converts all the governing equations of fluid dynamics into the form of a state-space model. Four typical cases are selected to evaluate both the accuracy and speed of SFD, compared with fast fluid dynamics (FFD), which is another fast airflow simulation program. Results show that SFD is capable of achieving faster-than-real-time airflow simulation with an accuracy similar to FFD. The computing time of SFD is longer than FFD when the time step size is the same. However, SFD can generally produce better results than FFD when the time step size is larger, which allows SFD run faster than FFD.

1. Introduction

Advanced controls of Heating, Ventilation, and Air Conditioning (HVAC) systems are intensively being implemented to improve energy efficiency and thermal comfort for buildings. Stratified airflow distribution, such as displacement ventilations, in a space is usually introduced by those systems. To evaluate the control performance of the systems using simulation, predictions of temperature and airflow distribution are critical. As summarized by Chen [1], there are mainly three types of models for airflow prediction: multizone models, zonal models, and computational fluid dynamics (CFD) models. Multizone models are computationally fast. However, they are not suitable for the simulation of non-uniform airflow and temperature distribution, as they assume that the air is well-mixed [2]. Zonal models [3] divide the room into several subzones and use empirical formulas to simulate airflow in specific area. But users of zonal model should be aware in advance of where the specific areas are, which is not very flexible. CFD models are versatile and can provide the most comprehensive information. It is nevertheless too computationally demanding to be applied in control simulations [4].

To resolve the obstacle imposed on CFD models, many researchers have explored various alternatives in order to reduce the computing time for non-uniform airflow and temperature distribution. These include the coarse-grid techniques, and some efficient equation-solving techniques. The coarse-grid techniques are more straightforward solutions to speed up CFD simulations, as calculation loads drop substantially with the decrease of grid nodes. Wang and Zhai [5] systematically examined the accuracy and speed improvement of coarse-grid CFD under several cases. Coarse-grid CFD can reduce the computing time by more than 16 times (even 100 times for some cases) compared to fine-grid CFD and achieve acceptable accuracy at the same time (even more accurate for some cases). However, one should bear in mind that the pre-acquisition of the airflow prior to applying the coarse-grid techniques is pivotal.

A more fundamental way to speed up CFD is to develop efficient equation-solving techniques such as fast fluid dynamics (FFD) and state-space methods. FFD uses the time splitting method and solves the governing equations sequentially after dividing the complex equation into several simple ones according to the number of terms in the equation. FFD was first developed by Stam [6] for simulating fast fluid movement in computer games. Zuo and Chen first introduced FFD into the airflow simulation in building area, by systematically evaluating its accuracy and speedup over CFD [7]. With a certain loss of accuracy, FFD is found to be 50 times faster than CFD models when running on CPU, and another 30 times speedup can be achieved if one is running simulations on a graphics processing unit (GPU) [8]. Furthermore, Tian,

* Corresponding author. 4800 CaoAn Highway, Shanghai 201804, China.
E-mail address: yiqunpan@tongji.edu.cn (Y. Pan).

http://dx.doi.org/10.1016/j.buildenv.2017.09.032
Received 6 July 2017; Received in revised form 27 September 2017; Accepted 29 September 2017
Available online 30 September 2017
0360-1323/

Nomenclature

u, v	Velocity, m/s
ρ	Density, kg/m^3
e	Internal energy, J
T	Temperature, K
R	Gas constant for air, J/kg
C_p, C_v	Specific heat at constant pressure and volume, $J/(kg{\cdot}K)$
μ	Dynamic viscosity, $Pa{\cdot}s$
μ_t	Turbulent viscosity, $Pa{\cdot}s$
λ	Thermal conductivity, $W/(m{\cdot}K)$
l	Length, m
Pe	Peclet number
Pr	Prandtl number
CTR	Computing time ratio

Superscripts

n, $n-1$	Time step index

Subscripts

Eff	Effective
r, l, t, b	Location index of the cell boundaries
R, L, T, B	Location index of the cell

Sevilla, Li, Zuo and Wetter [9] combined FFD with an in situ adaptive tabulation (ISAT) algorithm that is essentially a reduced order model. In addition, Jin, Liu and Chen [10] applied coarse grid technique to FFD in simulation of buoyancy-driven flow in buildings to improve the FFD speed even further. They integrated an analytical model of plume to improve the simulation performance for the heat source whose physical size is smaller than the mesh size. Additionally, the simulation speed was also accelerated.

After all the descriptions above, FFD seems like a more versatile framework with which to do fast flow simulation. However, FFD usually requires a small time step size (sometimes as small as 0.01s) when using semi-lagrangian method for the convection term, which introduces a numerical viscosity related to the time step size. Additionally, the first-order time splitting method imposes restrictions on choosing the time step size. Thus, FFD can be costly in computation time when simulating the slowly changing flows because of the long simulation time required. To solve this problem, some researchers have tried to use higher order interpolation to mitigate the numerical diffusion caused by semi-lagrangian method [11] or directly replacing the semi-lagrangian with other implicit schemes [12]. But a more complicated scheme or method is likely to impact the speed and convergence.

Compared to FFD, state-space method works towards the same goal with a different approach. Conventionally, state-space method only solves the energy conservation equations. Precedent CFD calculations are often utilized to calculate the mass flow rate and thermal conductivity on the boundaries in order to define the mutual influence of adjacent cells. The matrices of state-space model are generated based on the precedent CFD results. More details of state-space method are given in section 2.1.

Peng and van Paassen [13] used the state-space method to simulate the dynamic response of temperature distribution for a room with forced convection air flow. They assumed the flow field was fixed and used CFD to pre-calculate the velocity field and turbulence viscosity. After that, the discretized energy conservation equations were transformed into the state-space model and solved. By not solving the momentum conservation equations, they saved a great amount of computing time. Yao, Yang, Huang and Wang [14] also used state-space method to calculate the dynamic response of not only the temperature but also humidity in a three-zoned room. In addition, Parker, Lorenzetti and Sohn [15] implemented it for the multi-zone contaminant transport.

To further improve the speed of state-space method, researchers offered several different options. Some of them [13,14] merged the adjacent small cells into relatively large zones to directly reduce the size of state-space model. Sempey, Inard, Ghiaus and Allery [16] used Proper Orthogonal Decomposition (POD) method to find an optimal basis using several snapshots extracted from precedent transient CFD simulation result and achieved the reduced order. Parker, Lorenzetti and Sohn [15] proposed to solve the equations analytically using the matrix exponential and achieved speed improvement as well.

However, for the conventional way of using state-space method in fluid dynamics, a major limitation is the assumption of fixed velocity field. When the objective room is an open space with multiple independently controlled supply air inlets, the dimensions of the problem increases substantially, leaving it impractical to train the model using the pre-calculated CFD results.

In this paper, we propose to apply the state-space technique to the Navier-Stokes equations together with all the other governing equations in CFD. For ease of illustration, we call the proposed method State-space Fluid Dynamics (SFD) in the rest of the paper. First, we describe the SFD model in the methodology section. Then, we evaluate the accuracy and speed of SFD by using multiple typical airflows in buildings, including a lid driven cavity flow, a forced convection flow, a natural convection flow, and a mixed convection flow. We compare the results of SFD with its counterpart FFD. Finally, we propose the future work for SFD.

2. Methodology

This section will first introduce the method of state-space model and the governing equations of fluid dynamics. Then we will focus on the discretization and linearization of the equations.

2.1. State-space model

As we know, the state space model is usually used to describe a linear system represented as:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t), \tag{1}$$

where t is time, $\boldsymbol{x}$ the state vector, $\boldsymbol{u}$ the input vector, $\boldsymbol{A}$ the system matrix, and $\boldsymbol{B}$ the input matrix. The system matrix $\boldsymbol{A}$ is used to describe the interactions between different variables in the state vector $\boldsymbol{x}$, while the input matrix $\boldsymbol{B}$ represents the influences from the input vector $\boldsymbol{u}$ to the state vector $\boldsymbol{x}$. Matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ can be time-invariant or time-variant as shown in Equation (1). When they are time-variant, we can also use discrete time-variant state-space model to describe the system as Equation (2):

$$\boldsymbol{x}(k+1) = \boldsymbol{A}(k)\boldsymbol{x}(k) + \boldsymbol{B}(k)\boldsymbol{u}(k), \tag{2}$$

where k is the time step index, $\boldsymbol{A}(k)$ and $\boldsymbol{B}(k)$ are constant within the time step k and vary before entering time step $k+1$. In the context of airflow simulation, the state vector $\boldsymbol{x}(k)$ represents the different variables (e.g. velocity, temperature, and density) and the input vector $\boldsymbol{u}(k)$ represents the boundary conditions (e.g. inlet velocity and temperature, and outlet pressure etc.). The convection and diffusion between cells are represented by the system matrix $\boldsymbol{A}(k)$. The influences from the boundary conditions are described in the input matrix $\boldsymbol{B}(k)$. Since the velocity and diffusion coefficients on the cell boundaries are always changing as the flow field develops, we use the form of discrete time-variant state-space model to fit in the descriptive equations.