Doctoral Dissertations and Master's Theses

Spring 2023

Numerical Modeling of Synthetic Vortical Disturbance Interactions Using OpenFOAM

Ahmed Bilal
ahmedb2@my.erau.edu

Follow this and additional works at: https://commons.erau.edu/edt

Part of the Aerodynamics and Fluid Mechanics Commons

NUMERICAL MODELING OF SYNTHETIC VORTICAL DISTURBANCE

INTERACTIONS USING OPENFOAM

By

Ahmed Bilal

A Thesis Submitted to the Faculty of Embry-Riddle Aeronautical University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Aerospace Engineering

March 2023

Embry-Riddle Aeronautical University

Daytona Beach, Florida

ABSTRACT

Nowadays, the number of Advance Air Mobility (AAM) Electic Vertical Take-off and Landing (eVTOL) concepts is rapidly increasing due to their capability of vertical take-off and landing at vertiports located on rooftops of tall urban buildings, which does not require using the ground space in a condensed urban environment. Such flight operations, however, will be greatly affected by the vertiport's highly unsteady, turbulent flow environment. The goal of this study is to model the turbulent wind interaction with a building to understand the unsteady flow characteristics around vertiports and match the induced unsteady disturbance field to the canonical disturbance forms, which then could be used in the development of the reduced order models (ROMs) for multi-rotor unsteady aerodynamic and acoustics responses. This study employs an open-source Navier-Stokes OpenFOAM solver to model the turbulent wind flow around a building. Based on the previous studies, a momentum source generating an upstream synthetic disturbance field (such as a time-harmonic gust or turbulence with required characteristics) is employed. Validation test studies are conducted to examine and compare with previous results for the benchmark case of 2D time-harmonic gust interaction with NACA0012 airfoil, and then the case of the turbulent wind interaction with a building. Furthermore, the non-uniform mean flow profile characteristic of the urban atmospheric boundary layer (ABL) is implemented in the OpenFOAM simulations to examine and compare the effect of the non-uniform mean flow on the turbulent wind evolution and interaction with the rooftop vertiport.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1   Introduction

In recent years, the number of proposed Advance Air Mobility (AAM) Electic Vertical Take-off and Landing (eVTOL) designs have been rapidly increasing (to reach nearly 600 in 2022, Figure 1.1) due to the interest in their capability to conduct vertical take-off and landing at vertiports located on rooftops of tall urban buildings, which does not require using the ground space in a condensed urban environment. Such eVTOL vehicles will be operating in highly unsteady urban flow environments characteristic of the rooftop vertiports. Thus, there is a need to model the eVTOL unsteady interactions and corresponding aerodynamic and acoustic responses using CFD simulations of turbulent urban flows. This thesis' objective is to explore the use of the synthetic disturbance models recently developed in OpenFOAM CFD software for future eVTOL performance prediction studies. Indeed, both the rotorcraft safety and noise will be affected due to the interaction with the turbulent wind flow around downstream structures such as a building, as shown in Figure 1.2, particularly at the rooftops where a vertiport may be located. Therefore, the objective is to model the turbulent wind interaction with a building to understand the unsteady flow characteristics around the vertiport, which will pave the way to future studies of multi-rotor responses in such an environment. By matching the obtained unsteady flow characteristics to the canonical disturbance forms, this study will also create an opportunity to develop reduced-order models (ROMs) of eVTOL responses amenable to optimized active response control strategies.
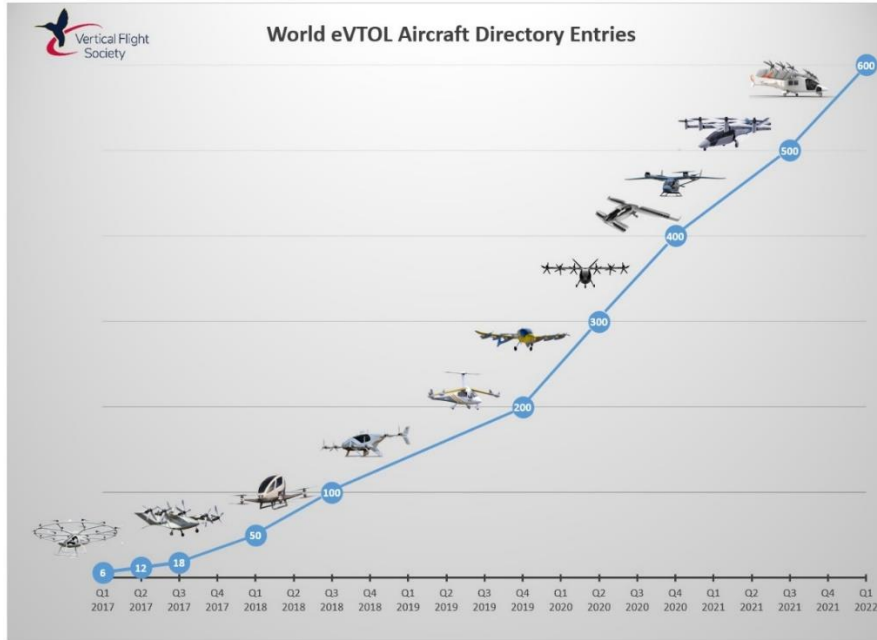
Figure 1.1 The number of eVTOLs concepts [1]
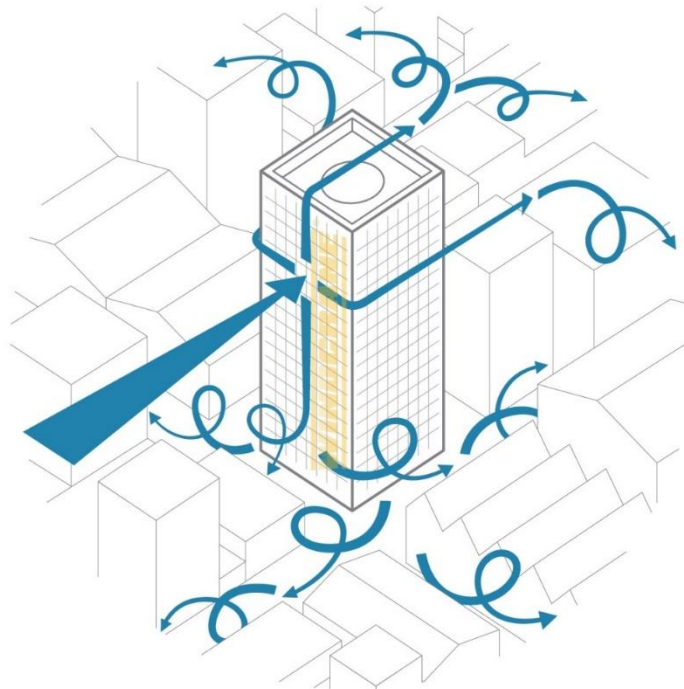
Figure 1.2 Turbulent flow interaction with building [2]

2   Study Objectives

There are four objectives in this study. First, our OpenFOAM model will be verified by implementing the benchmark case study of NACA0012 airfoil interacting with 2D time-harmonic gust [3]. Secondly, OpenFOAM simulations will be conducted for the turbulent wind interaction with a downstream building structure assuming a uniform mean flow. Thirdly, OpenFOAM simulations will be conducted for a non-uniform mean flow characteristic of the urban atmospheric boundary layer (ABL), with results for the induced rooftop flow compared against those obtained using the uniform mean flow assumption. Finally, the building rooftop vertiport unsteady flow characteristics will be discussed with preliminary matching to canonical disturbance forms, to be employed in the future analyses of multi-rotor interaction with unsteady vertiport flow environment and its control.

In the first study (Section 4), the OpenFOAM model will be verified by implementing the NACA0012 gust-airfoil interaction study and examining the spectrum of the lift coefficient. After the computational model is validated, Section 5 will analyze the turbulent wind interaction with downstream building structure using a synthetic turbulence generation (STG) based momentum source model [3], first using the assumption of the uniform mean flow, and then considering the effect of the non-uniform mean flow due to the urban ABL development. The objective will be to compare and validate the predictions for the unsteady forces induced on the building against the experimental results of Rainhold's [4], and numerical results of Huang et al [5] obtained using the discretizing and synthesizing random flow generation (DSRFG) procedure. Finally, Section 5 discusses the unsteady vertiport flow field and its matching to the canonical disturbance forms.

3   Numerical Methodology

Both 2D gust-airfoil validation and vertiport flow simulation were conducted using OpenFOAM computational fluid dynamics software.

3.1   OpenFOAM Modeling

OpenFOAM is a multipurpose open-source platform with various possible applications. Written in C++, it offers the option to develop custom codes and includes several native apps (such as solvers). Version 17.06 is used in this work. The selected application is the native incompressible flow solver pimpleFoam. The discretization is based on a finite-volume method, with second-order centered schemes used to approximate the convective and diffusive terms. The solution algorithm is based on the PIMPLE algorithm, a merged version of PISO and SIMPLE. The pimpleFoam algorithm uses an inner PISO loop to get an initial solution, which is then under-relaxed and corrected using an outer SIMPLE loop. Unsteady simulations are possible with this method at Courant numbers greater than 1. Theoretically, very large Courant numbers could be maintained if a large number of SIMPLE correction loops were applied along with large under-relaxation factors. However, this impacts the solution's time accuracy because the solver misses events that occur within the time step. A too-large time step generally smears the solution, regardless of the grid resolution. When using pimpleFoam one has to keep in mind that too large Courant numbers risk reducing the accuracy of the solution. Hence, the time step is selected so that the max of Courant number is less than 1. pimpleFoam starts with solving momentum equations, then pressure coefficient and velocity based on its initial condition, then equations related to turbulence. By using this model, a three-dimensional, incompressible convected perturbation velocity field can be generated by prescribing a momentum source in an arbitrary region of the computational domain. The governing equations are the well-known Navier-Stokes equations for three-dimensional, incompressible fluids as follows:

Continuity equation:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial u}{\partial z} = 0 \tag{1}$$

Momentum equations:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}$$
$$= -\frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{\partial}{\partial x}\left(v_{eff}\left(\frac{\partial u}{\partial x}\right)\right) + \frac{\partial}{\partial y}\left(v_{eff}\left(\frac{\partial u}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(v_{eff}\left(\frac{\partial u}{\partial z}\right)\right) \tag{2}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z}$$
$$= -\frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{\partial}{\partial x}\left(v_{eff}\left(\frac{\partial v}{\partial x}\right)\right) + \frac{\partial}{\partial y}\left(v_{eff}\left(\frac{\partial v}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(v_{eff}\left(\frac{\partial v}{\partial z}\right)\right) \tag{3}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}$$
$$= -\frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{\partial}{\partial x}\left(v_{eff}\left(\frac{\partial w}{\partial x}\right)\right) + \frac{\partial}{\partial y}\left(v_{eff}\left(\frac{\partial w}{\partial y}\right)\right) + \frac{\partial}{\partial z}\left(v_{eff}\left(\frac{\partial w}{\partial z}\right)\right) \tag{4}$$

3.2   Momentum Source for Synthetic Disturbance Field

The synthetic disturbance field was first initialized with a momentum source. According to Kraichnan's research [6], the superposition of the harmonic functions listed below can be used to create an isotropic fluctuating velocity field in x space:

$$u_i(x_j, t) = \sum_{n=1}^{N}\left(a_i^n\cos\left[k_j^n x_j - \omega_n t\right] + b_i^n\sin\left[k_j^n x_j - \omega_n t\right]\right) \tag{5}$$

Without sacrificing generality, it is assumed that the synthetic unstable disturbance field is convected by a uniform mean flow u_∞ aligned with x_1-direction resulting in a relationship between the modal perturbation frequency ω_n and the wavenumber k_1^n by $\omega_n = u_\infty k_1^n$. Additionally, it is assumed that the disturbance is incompressible (divergence-free), which implies the coupling between the modal amplitudes and wavenumbers are followed:

5

$$a_j^n k_j^n = b_j^n k_j^n = 0 \qquad (6)$$

It is thought that there are very few interactions between the perturbation modes and other waves. The solution for the momentum source terms imposed in the computational domain region with extent in the mean flow convection direction (x_1) can be found in the following form,

$$S_i(x_j, t) = \begin{cases} \displaystyle\sum_{n=1}^{\infty} \omega_n K_n g(x_1)(b_i^n \sin \sigma_n - a_i^n \cos \sigma_n) & \text{for } i = 1, \\ \displaystyle\sum_{n=1}^{\infty} U_\infty K_n g'(x_1)(b_i^n \cos \sigma_n + a_i^n \sin \sigma_n) & \text{for } i = 2,3,3 \end{cases} \qquad (7)$$

In Eq. (3), K_n is the constant.

$$K_n = \frac{\left(\frac{k_1^n}{2\pi}\Delta_s\right)^2 - 1}{\sin\left(\frac{k_1^n}{2}\Delta_s\right)} \qquad (8)$$

the limiter function $g(x_1)$ in this work is defined as,

$$g(x_1) = \begin{cases} \frac{1}{2}\left(1 + \cos\left[\frac{2\pi}{\Delta_s}(x_1 - x_{1s})\right]\right) & \text{for } |x_1 - x_{1s}| \le \frac{\Delta_s}{2} \\ 0 & \text{for } |x_1 - x_{1s}| > \frac{\Delta_s}{2} \end{cases} \qquad (9)$$

3.3 Turbulence STG Model

In the current study, a synthetic turbulence generation (STG) model is used, in which an upstream turbulent flow field is generated using a three-dimensional Fourier spectrum for perturbation velocity that fits a prescribed energy spectrum with a set of turbulence integral scales. The process to build a synthetic turbulent field $u'(r, t)$ matching target spectrum comprises multiple steps, as outlined by Adamian and Travin (2011) and Shur (2014b). First, the second-moment tensor $< u_i' u_j' >$ is checked to see if it is equivalent to R using the Cholesky decomposition of the Reynolds stress tensor $\widehat{R}$.

$$\hat{A} = \{a_{ij}\} = \begin{pmatrix} \sqrt{R_{11}} & 0 & 0 \\ R_{21}/a_{11} & \sqrt{R_{22} - a_{21}^2} & 0 \\ R_{31}/a_{11} & (R_{32} - a_{21}a_{31})/a_{22} & \sqrt{R_{33} - a_{31}^2 - a_{32}^2} \end{pmatrix} \tag{10}$$

$$\hat{R} = \hat{A}^T \hat{A} \tag{11}$$

Fluctuating velocity $u'(r,t)$ is expressed via the components of the tensor $\hat{A}$

$$u'(r,t) = a_{ij} v_i'(r,t) \tag{12}$$

Equation (10)'s requirements should be met by the auxiliary vector of fluctuating velocity

$v'(r,t)$, which is known as the superposition of Fourier modes (11).

$$< v_i' v_j' > = \delta_{ij} \text{ and } < v_i' > = 0 \tag{13}$$

$$v'(r,t) = \sqrt{6} \sum_{n=1}^{N} \sqrt{q^n} (\sigma^n \cos [k_j^n d^n r + \varphi^n + \omega^n t]) \tag{14}$$

where
- N is the number of modes;
- q^n is the amplitude of a single mode n (defined by the chosen energy spectrum $E(k^n)$ and normalized as shown in Equation (12), definition of $E(k)$ is givenby Equations (13) and (14);
- k^n is the amplitude of the wave number;
- $k^n = k^n d^n$ wavenumber vector;
- σ^n is a unit vector it must be normal to d^n;
- φ^n is the phase of the mode n;
- d^n a random unit vector, that denes direction of a wavenumber. The vector is distributed uniformly on a unit sphere that ensures isotropy of generated flow field.

$$E_g(k_j^n) = 16 \left(\frac{2}{\pi}\right)^{\frac{1}{2}} (k_j^n)^4 \exp\left(-2(k_j^n)^2\right) \tag{15}$$

$$q^n = \frac{E(k^n)\Delta k^n}{\sum_{n=1}^{N} E(k^n)\Delta k^n}, \sum_{n=1}^{N} q^n = 1 \tag{16}$$

$$E_{vk}(k^n) = \frac{55}{9} u_{rms}^2 \frac{(k^n/k_e)^4}{(1 + 2.4(k^n/k_e)^2)^{17/6}} \tag{17}$$

$$d_1^n = sin\theta^n cos\varphi^n$$

$$d_2^n = sin\theta^n sin\varphi^n$$

$$d_3^n = cos\theta^n$$

$$\sigma_1^n = \cos\varphi^n \cos\theta^n \cos\alpha^n - \sin\varphi^n sin\alpha^n \qquad (18)$$

$$\sigma_2^n = \sin\varphi^n \cos\theta^n \cos\alpha^n - \cos\varphi^n sin\alpha^n$$

$$\sigma_3^n = -sin\theta^n \cos\alpha^n$$

All the wavenumbers and mode amplitudes are generated only once because they should be constant for every time iteration. The geometry of the mode n is depicted in Figure 3.1. The probability distributions of random numbers required for turbulence generation are given in Table 3.1.
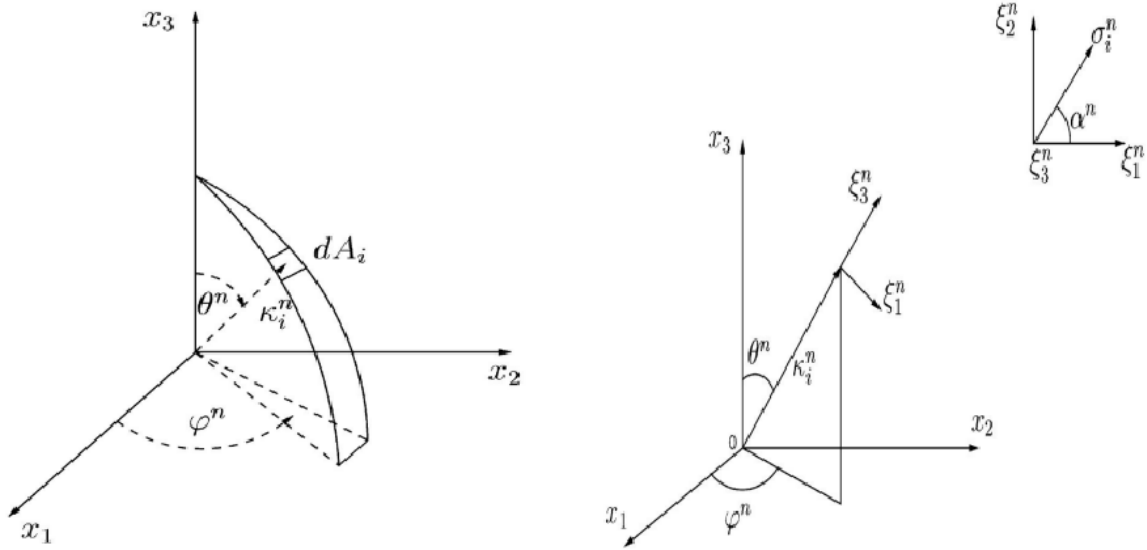
Figure 3.1 Directions of modes and wavenumbers in STG [7]; (a) Shell of a sphere on which wavenumbers are distributed (Davidson & Billson, 2004); (b) Geometry of the mode n (Davidson & Billson, 2004).

Table 3.1 Probability Distribution of Random Numbers φ^n, ψ^n, θ^n, α^n (Davidson & Billson, 2004)

$p(\varphi^n) = 1/(2\pi)$	$0 \leq \varphi^n \leq 2\pi$
$p(\psi^n,) = 1/(2\pi)$	$0 \leq \psi^n \leq 2\pi$

8

$p(\theta^n) = (1/2)sin\theta$	$0 \leq \theta^n \leq \pi$
$p(\alpha^n) = 1/(2\pi)$	$0 \leq \alpha^n \leq 2\pi$

It is important to note that in computational wind engineering applications, producing inflow turbulence that satisfies a target spectrum, such as the von Karman model, is crucial for assessing the effects of wind on buildings. It is clear from Figure 3.2, the von Karman model obeyed a realistic turbulence spectrum of wind speed fluctuation in the atmospheric boundary layer, and it was acknowledged that the turbulent eddies in the inertial subrange have a non-negligible impact on wind-induced fluctuating forces on buildings.

Figure 3.3 shows the velocities simulated by STG and DSRFG. The Figure demonstrates that higher frequency wind speed fluctuations are produced by the current approach same as DSRFG. This is consistent with the spectra displayed in Figure 3.4 (DSRFG) and Figure 3.5 (STG).
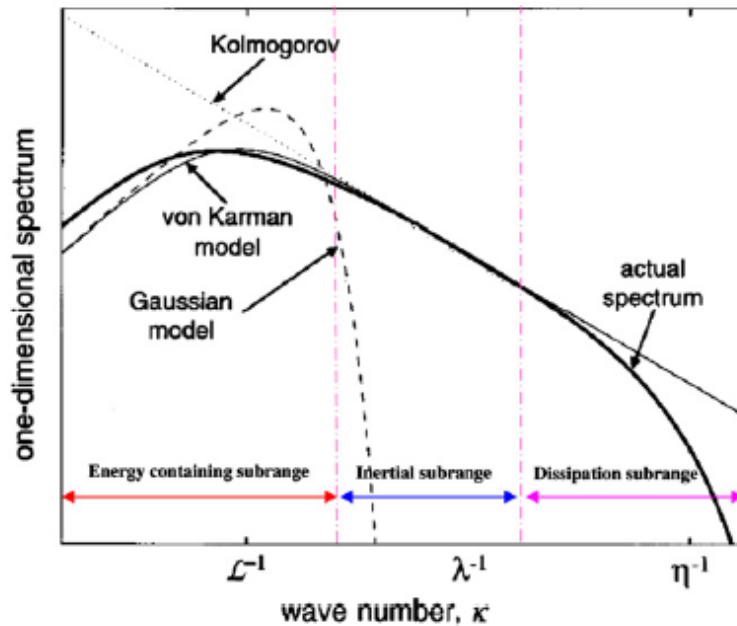
Figure 3.2 Turbulence subranges and display of spectra [5]
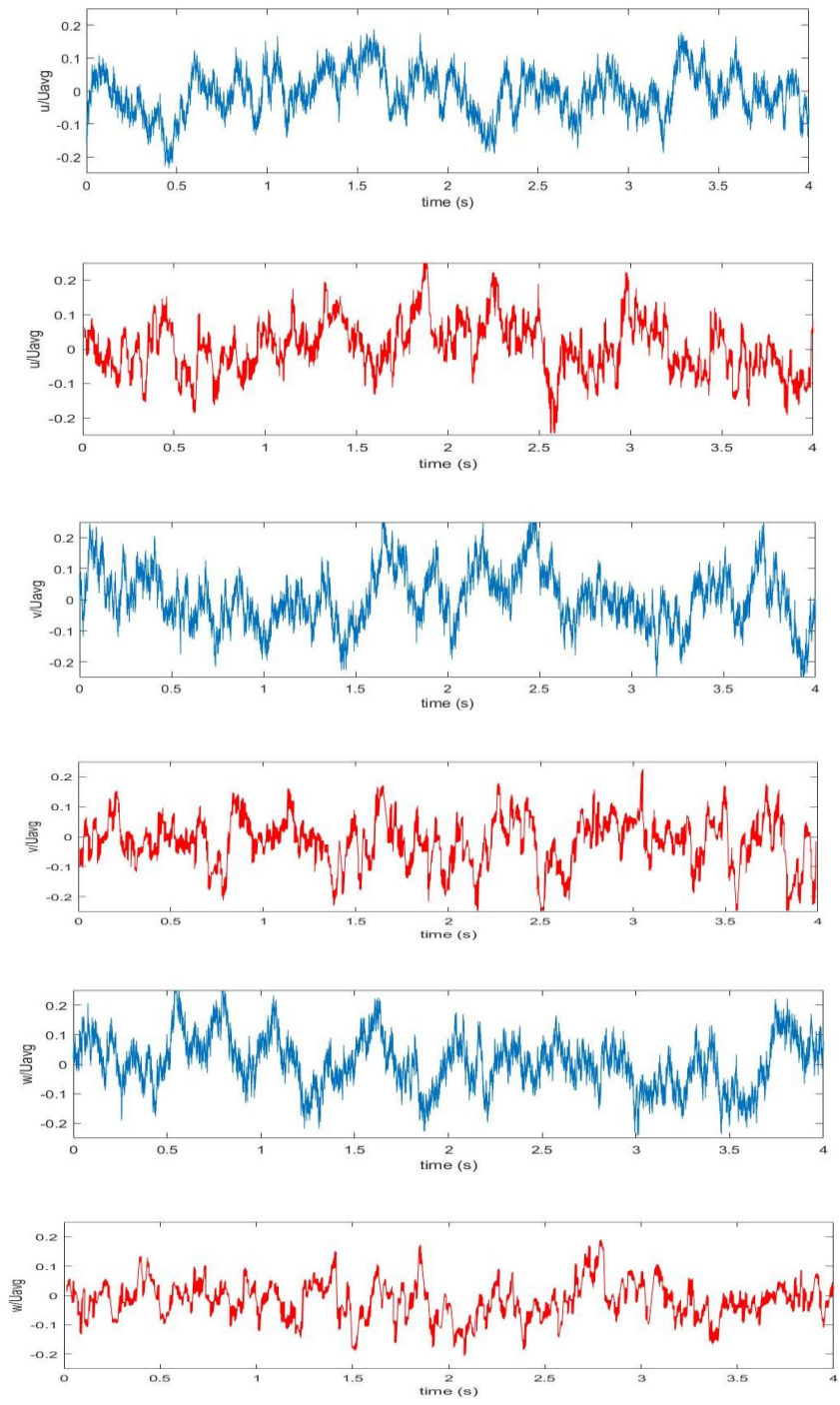
Figure 3.3 Wind velocity samples simulated by the STG (Blue) and DSRFG (Red) [5]
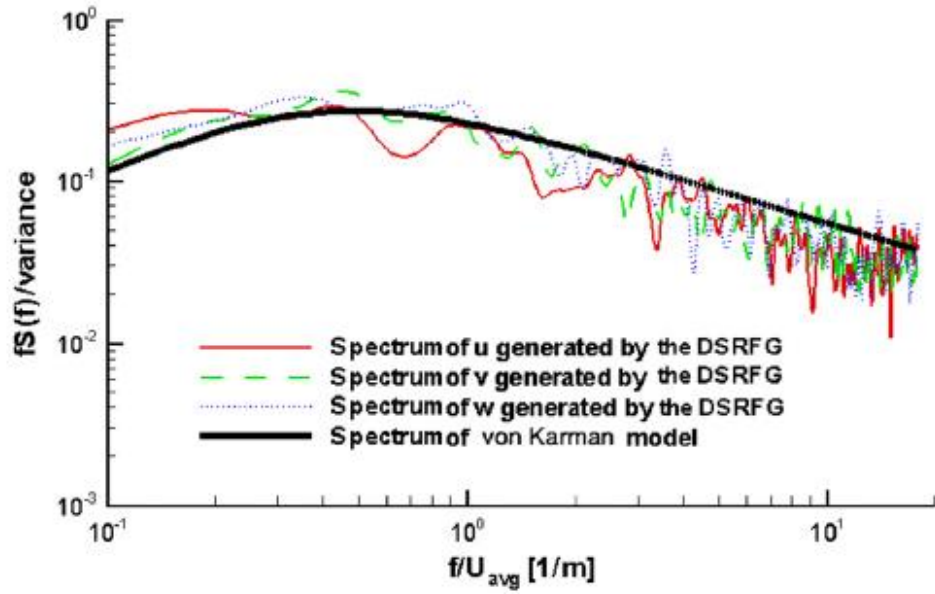
Figure 3.4 Comparison of the spectra by the DSRFG [5] method and von Karman model
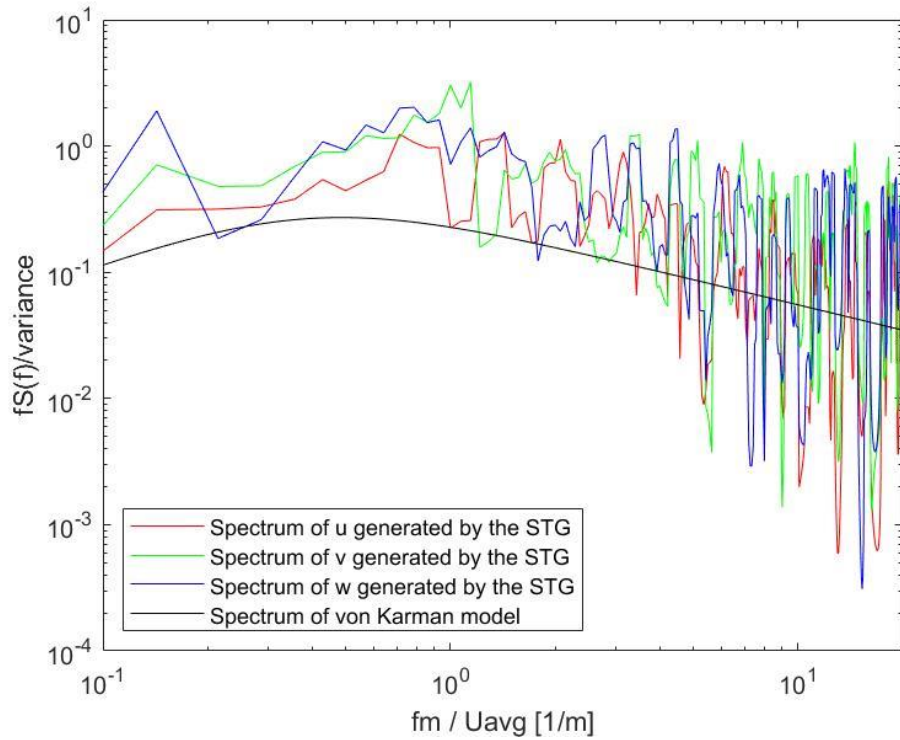
Figure 3.5 Comparison of the spectra by the STG method and von Karman model

4   Validation 2D Gust-Airfoil Case Study

First, the OpenFOAM model will be verified by implementing the NACA0012 gust-airfoil interaction study and comparing results with previous ANSYS Fluent results [3]. The grid of the simulation field is generated by using Pointwise mesh generation software.

4.1   Numerical Set-Up

The size of the computational domain is 50 by 50 non-dimensionalized by the airfoil chord, as shown in Figure 4.1. The red square indicates the region where the gust is generated over-imposed on the uniform upstream mean flow with velocity 16 m/s. The Reynolds number is 160,000, with the minimum grid cell size of 0.001 achieving a reasonably good resolution of the boundary layer. The grid information is shown in Table 4.1. The 2D gust was generated with a 45-degree wavefront angle, with the gust velocity amplitudes (non-dimensionalized by the mean velocity) of -0.1 and 0.1 in x and y-directions, respectively, as shown in Table 4.2 along with other parameters.
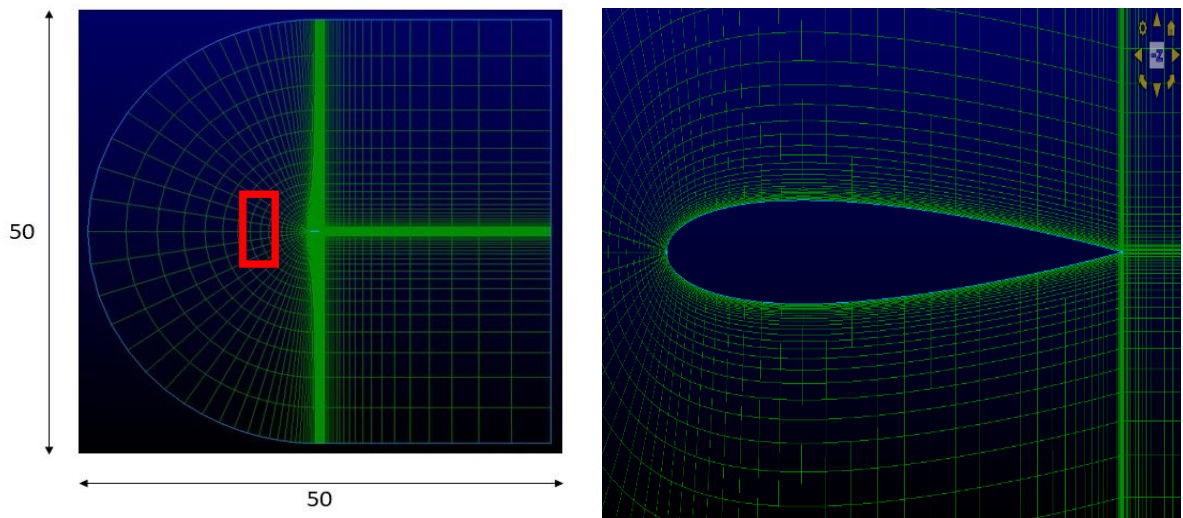
Figure 4.1 Computational domain (left) and airfoil near field (right) grid.

Table 4.1 Grid information

Number of Points	19600
Number of Faces	38612
Number of Cells	9604
Minimum Cell Size	0.001

Table 4.2 Flow condition

Uniform Velocity	16 m/s
Gust U and V Amplitude	-0.1, 0.1
Gust Angle	45^0
Reynolds Number	1.6×10^5
Boundary Layer Thickness	0.018 at 0.5 chord

4.2   Results

The result of the NACA0012 lift coefficient FFT spectral analysis is shown in Figure 4.2. The red line indicates the resolution cut-off based on the sampling frequency. The black dashed line represent Kolmogorov's $k^{-5/3}$ law for the spectral decay of the turbulent energy. The peak in the lift spectrum corresponds to the airfoil primary aerodynamic response at the impinging gust frequency.

Next, Figure 4.3 shows the surface mean pressure. One can note a discrepancy between the current OpenFOAM and the previous [3] ANSYS Fluent results near the trailing edge of the airfoil, attributed to the reduced mesh resolution in that region in the current simulations. However,

13

overall, the predicted surface pressure matches well with the previous result [3], which thus validates the current numerical approach.
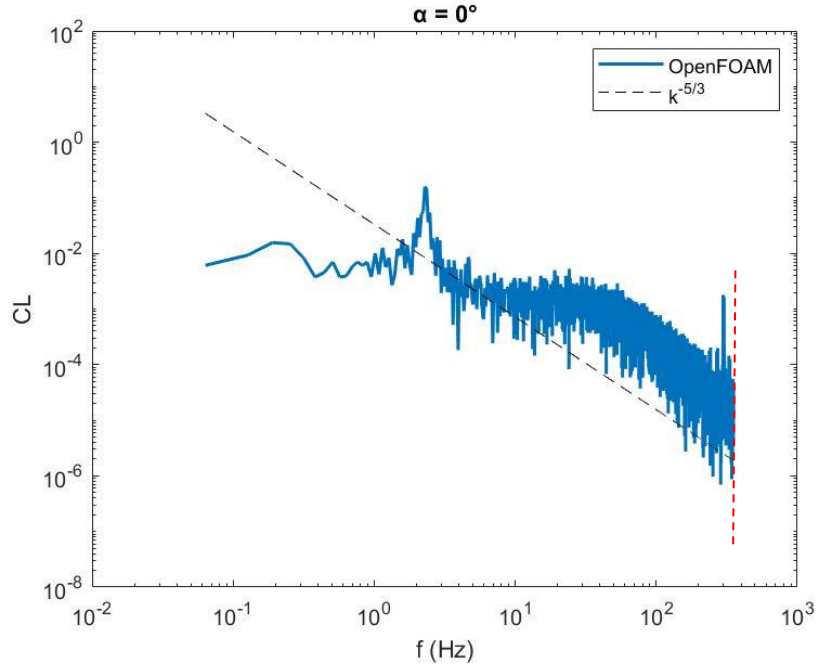
Figure 4.2 Spectrum of Lift coefficient with 2D Gust Interaction
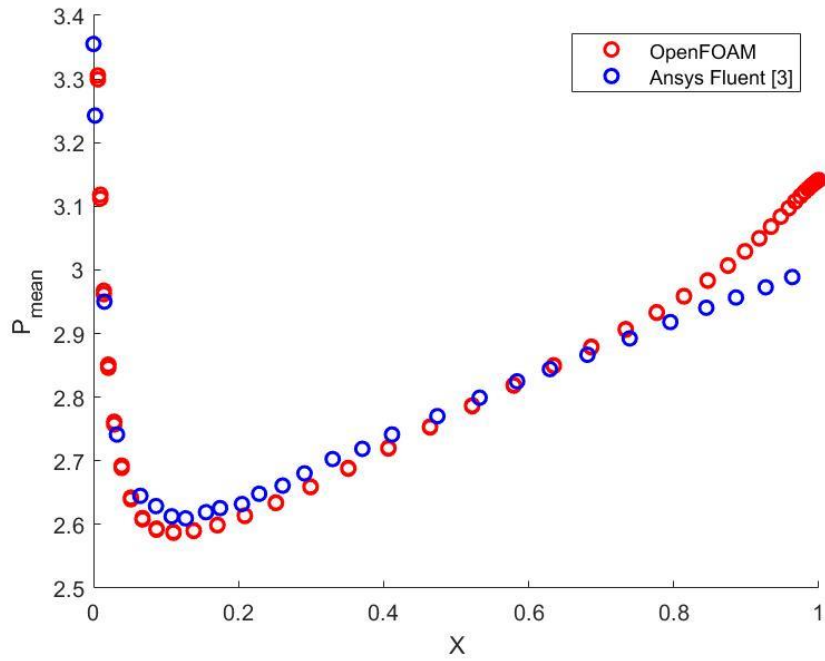
Figure 4.3 Mean surface pressure of NACA0012 with 2D Gust Interaction

5   Vertiport Flow Simulation

With the current OpenFOAM model validated, this section will analyze the turbulent wind interaction with downstream building structure using a synthetic turbulence generation (STG) based momentum source model [3], and using both uniform and non-uniform mean flow profiles. Because the building is a blunt body, the position of the turbulence source has to be carefully selected as this affects the generation of upstream turbulent flowfield.

5.1   Numerical Set-Up

To select the position of the source, the same 2D gust used above in the NACA0012 analysis was first generated upstream of the building. Figure 5.1 shows the computational domain parameters from the building flow simulations in [5], with the non-dimensional size of 39 in the x-direction and 17 in the y-direction, normalized by the building's width. The height of the domain in the spanwise z-direction is twice as big compared to the building's height. The left side of the domain is selected as the inlet and the right side is the outlet in the OpenFOAM model's boundary conditions. Figure 5.2 shows the simulation model generated with Pointwise. The turbulence parameters match those from the wind tunnel experiment [4] and DSRFG numerical model in [5].
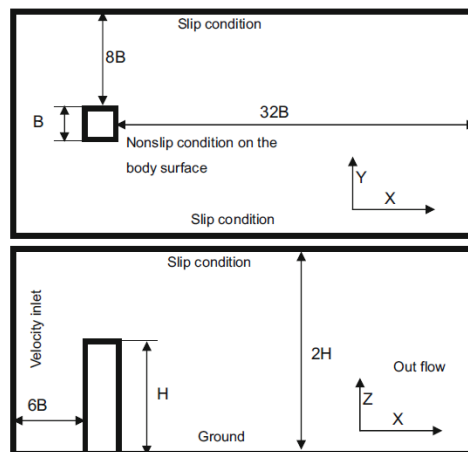
Figure 5.1 Computational domain [5].

15
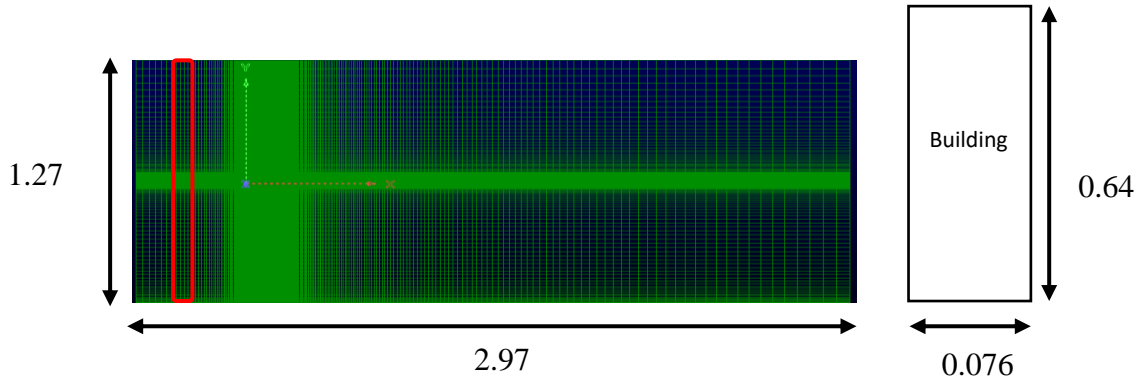
Figure 5.2 Computational grid (left) and building model (right)

5.2   Source Position

Figure 5.3 shows the contour plots of the obtained wind velocity and pressure, at 0.003s and 0.05s, respectively. The red square indicates the original position of the momentum source region in the 2D gust interaction test case study. However, due to the proximity of the gust source region to the building, the 2D gust did not have enough space to develop, and thus the size of the upstream domain and the synthetic disturbance source location were later modified.

The new (extended) upstream domain and the momentum source region location are shown in Figure 5.4. The additional grid size for the upstream region is half of the original computational grid size. Same as in Figure 5.2, the red square indicates the position of the gust source region. The velocity contours in Figure 5.5 correspond to the side and top views of the computational domain. The results clearly indicate the correct development of the gust, which confirmed the proper position of the source region for the synthetic turbulence generation.

The previous study [7] proposed two methods to prescribe the inflow turbulence superimposed with a non-uniform mean flow, as shown in Figure 5.6. It should be noted that the original synthetic disturbance approach for the momentum source region was developed with the assumption of the uniform mean flow. Nevertheless, with the assumption that the gradient of the mean flow in the slowly-changing atmospheric boundary-layer wind profile will not significantly

16

modify the synthetic disturbance development in the momentum source region, the first method prescribes the turbulence source "as is" in the non-uniform mean flow field and allows it to evolve downstream along with the freestream velocity profile. The second method generates the turbulence source in a piecewise manner, which involves imposing multiple source regions in the y-direction with piecewise-uniform mean velocity profiles. Both methods thus introduce a non-uniform mean velocity profile in the inflow region and convect the generated disturbance downstream. In this work, the first method is used to prescribe the synthetic turbulence in the flow field.

The generated turbulence distribution of intensity levels is significantly affected by the mean velocity and its non-uniform profile. This can be seen from the comparison of the vorticity contours obtained with uniform and non-uniform flows in Figures 5.7 and Figure 5.8.
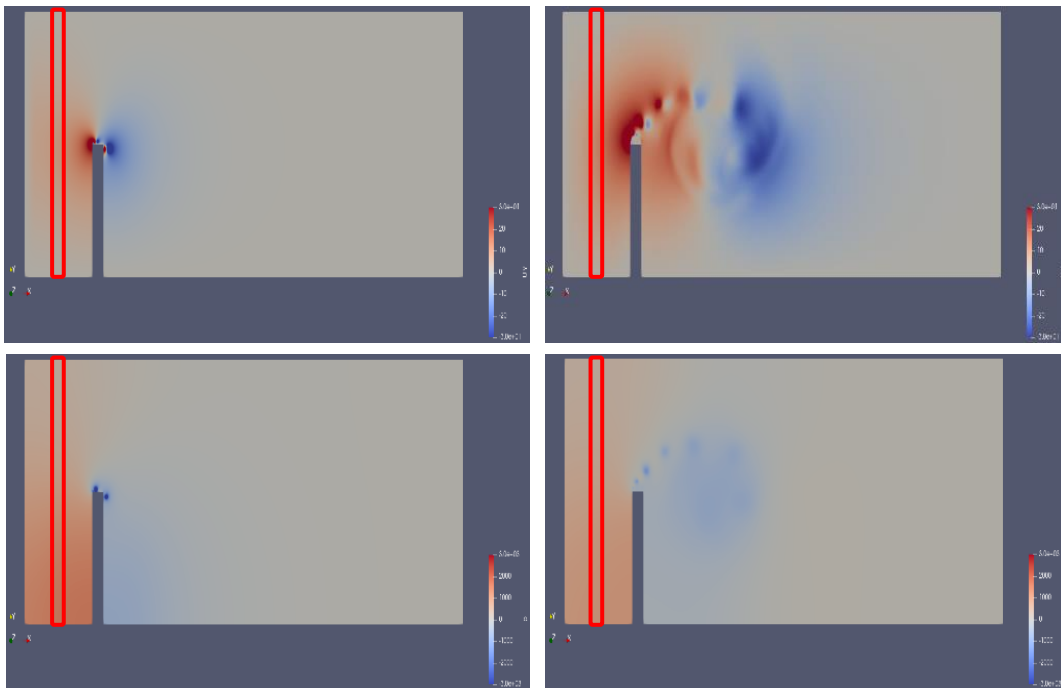
Figure 5.3 Contours of across-wind velocity (top) and pressure (bottom) at 0.003s (left) and 0.05s (right).
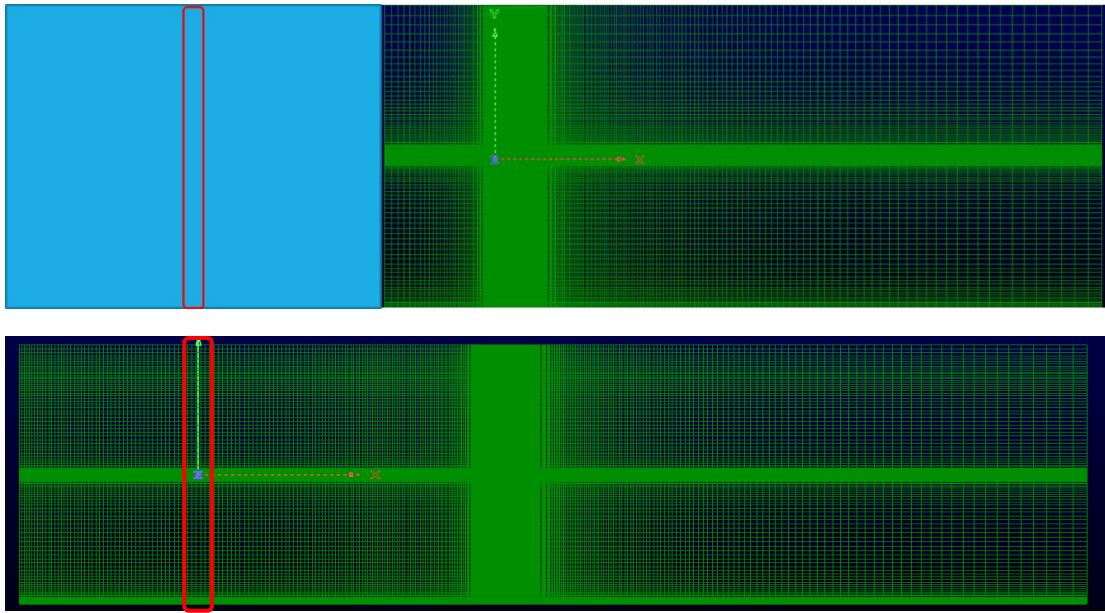
Figure 5.4 Revised computational grid.
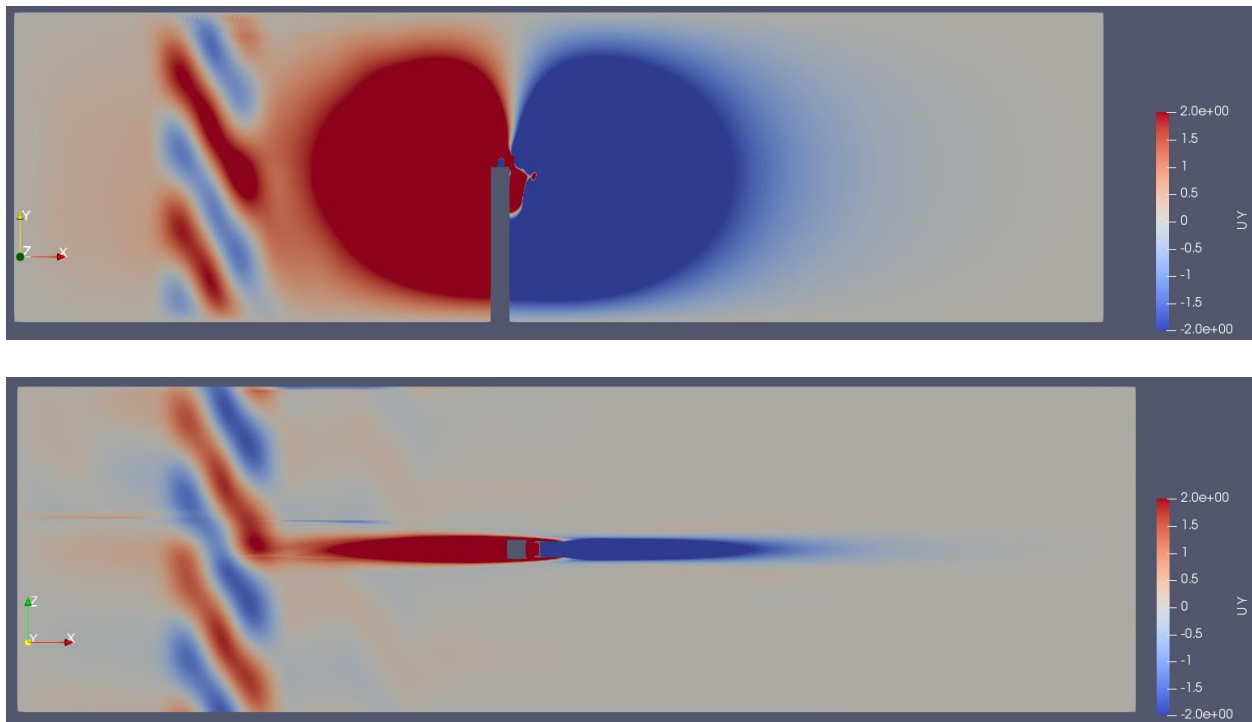
Figure 5.5 Couture of across-wind velocity side view (top) and top view (bottom).
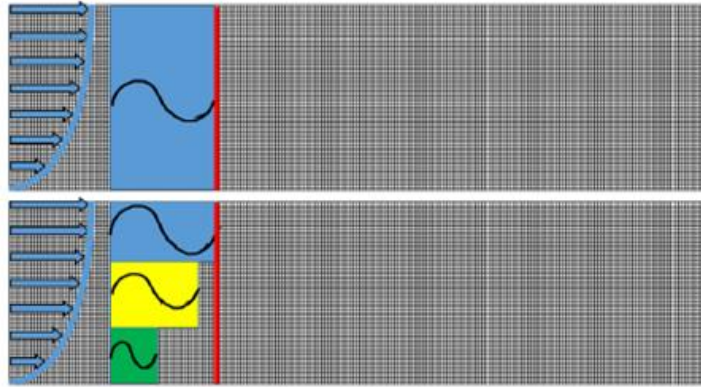
Figure 5.6 Two methods to prescribe turbulence using non-uniform mean flow profiles [7], with single turbulence source and multiple turbulence source regions.
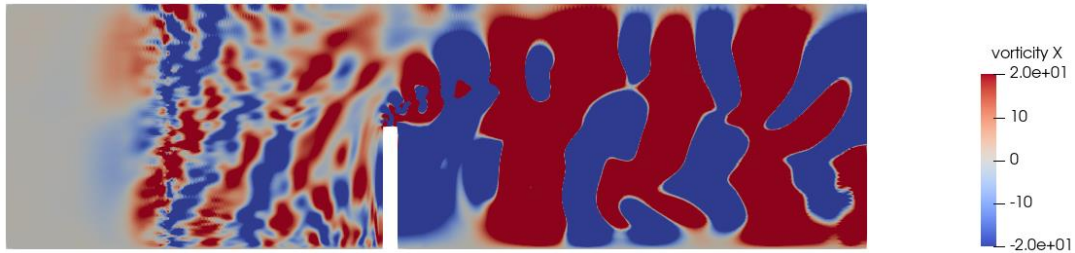
Figure 5.7 Building and turbulence interaction with uniform mean flow.
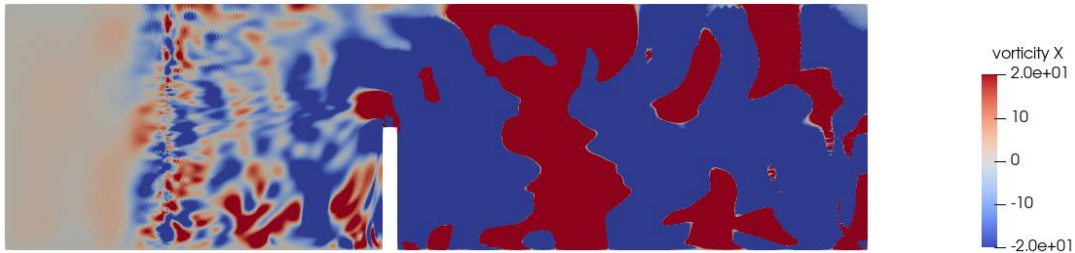
Figure 5.8 Building and turbulence interaction with non-uniform mean flow

5.3 Validation of STG Model

The inflow turbulence was generated using the STG method in OpenFOAM. For proper comparison with previous experimental [4] and numerical [5] studies, the generated synthetic turbulence parameters were chosen as specified in Table 5.1. The same parameters are used in the

study characterizing the vertiport unsteady environment with and without wind turbulence in Sections 5.4-5.6.

Figure 5.9 illustrates the time history of the obtained along-wind force coefficient fluctuation on the building surface at 80% height of the building, while 5.10 shows the across-wind coefficient fluctuation. The forces are calculated using velocity U_H at the building surface, building chord B and building level H_{LEVEL} as shown in Equation (19). In this case, H_{LEVEL} is 0.508.

$$C_F = F/(1/2 U_H^2 B H_{LEVEL})\qquad(19)$$

Figure 5.11 and Figures 5.12 show the corresponding normalized spectral densities of the wind-induced building forces compared against experimental results [4] and DSRFG results [5]. In these Figures, the modal frequencies nB/U_H are based on the FFT modal numbers n, the building chord B and the wind mean velocity U_H at the building surface, while $U_H C_F/B\sigma_F$ corresponds to the normalized spectral density of the force coefficients, and σ_F is rms value determined from the force coefficient,

$$\sigma_F = \sqrt{\overline{C_F^2}}\qquad(20)$$

The obtained across-wind force spectrum in Figure 5.12 slightly overpredicts both the experimental and DSRFG results at the higher frequencies. However, the overall comparison is quite satisfactory and provides a successful validation of the current model.

Table 5.1 Synthetic turbulence parameters and spectral characteristics

Average Velocity	14 m/s
Turbulence Intensity	10.25 %
Turbulence Length scale	45^0
Reynolds Number	1.1×10^4

Sampling Frequency	500 Hz
Sampling Rate	1000 Hz
Time Step Delta	0.001 s
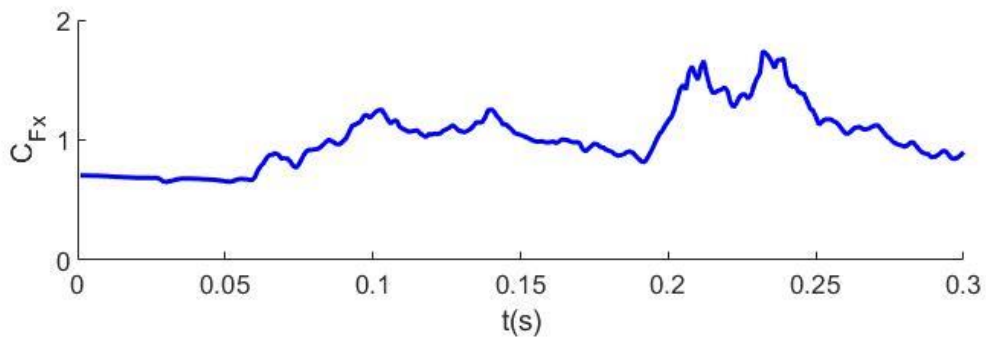
Figure 5.9 Along-wind force coefficient vs. time (s) by non-uniform mean flow with turbulence
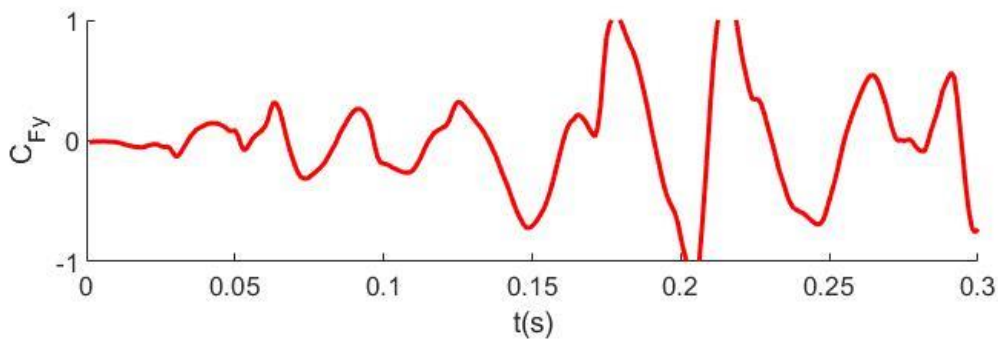
Figure 5.10 Across-wind force coefficient vs. time (s) by non-uniform mean flow with turbulence
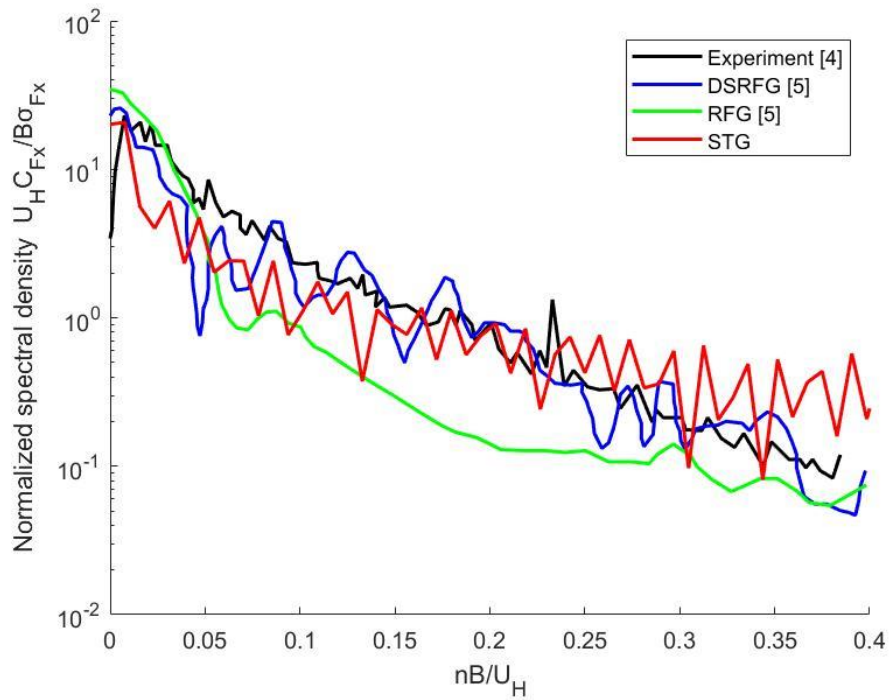
Figure 5.11 Normalized along-wind force coefficient spectral density by non-uniform mean flow with turbulence.
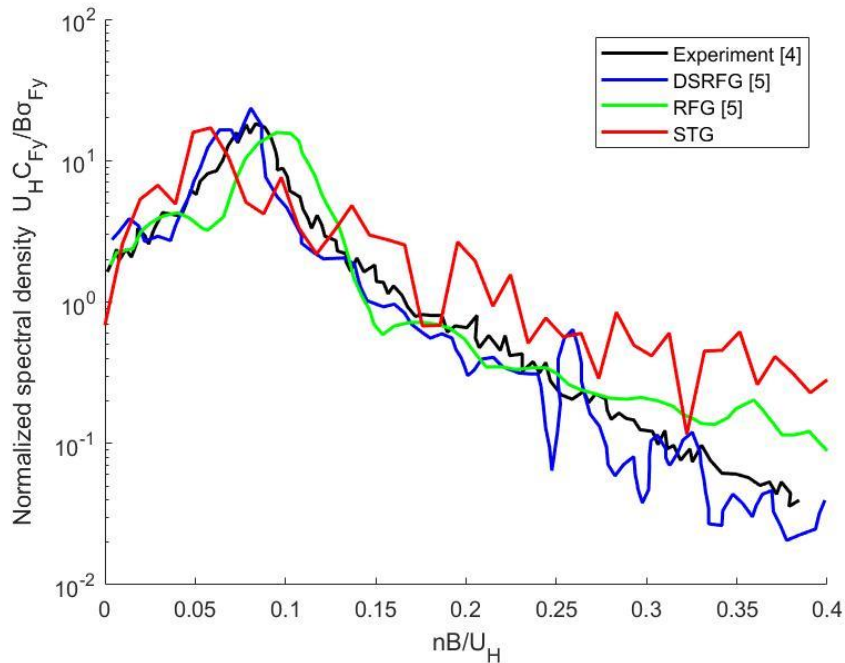
Figure 5.12 Normalized across-wind force coefficient spectral density by non-uniform mean flow with turbulence.

5.4   Results for Non-Turbulence Uniform vs. Non-Uniform Wind Velocity Profile

Figure 5.13 shows the velocity contour obtained with uniform mean flow, and Figure 5.14 is showing it obtained with non-uniform mean flow. The non-uniform mean flow velocity will less affect the lower part of the building as the velocity at the near-ground part of the building is slow compared to the uniform mean velocity. However, near the vertiports, the non-uniform mean velocity has a strong effect. In the study, the results will be compared against the 80% height of the building as it's used in the DSRFG simulation. Figure 5.15 shows the results for the building force time history obtained for uniform mean flow profile and non-uniform flow profile. The converged result of the along-wind coefficient was 1.364 with uniform mean flow and 1.178 with non-uniform mean flow. The percentage difference is 14.6%. Consequently, the results are significantly affected by the assumption of a uniform vs. non-uniform mean flow profile in the simulations.
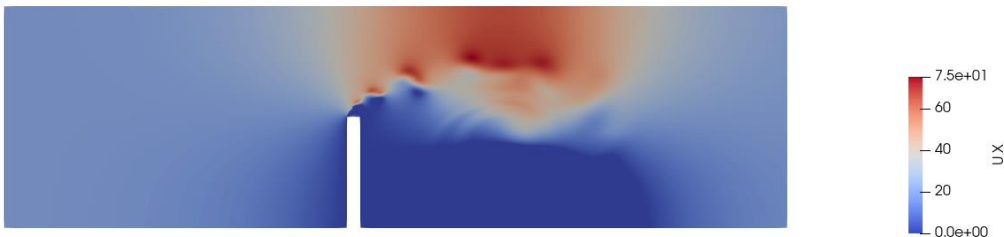
Figure 5.13 Uniform mean flow interaction with building without turbulence.
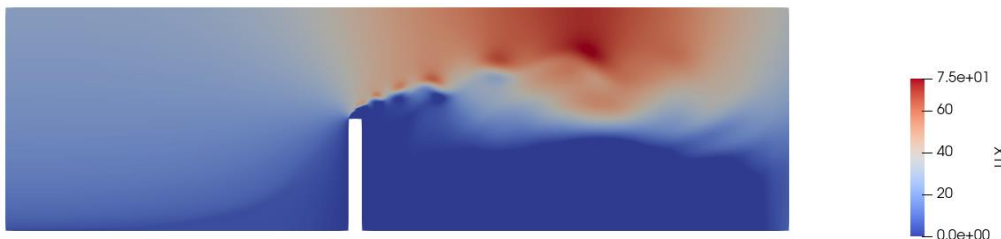
Figure 5.14 Non-Uniform mean flow interaction with building without turbulence.
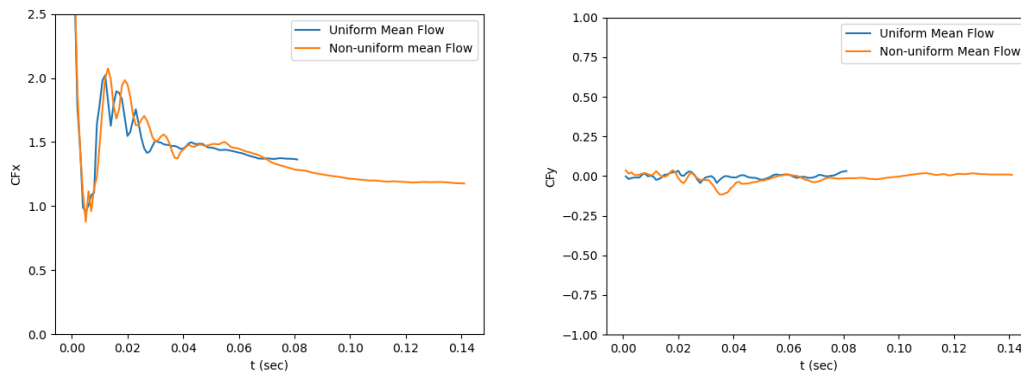
Figure 5.15 Along-wind coefficient vs. time (s) (left) and across-wind coefficient vs. time (s) (right) of uniform mean flow and non-uniform mean flow without turbulence

5.5 Results for Turbulence Uniform vs. Non-Uniform Wind Velocity Profile

Figure 5.16 shows the results for the building force time history obtained for uniform mean flow profile with imposed turbulence. Figure 5.17 shows the same for the non-uniform mean flow profile. The fluctuation of the across-wind force coefficient with the uniform mean flow is larger compared to the non-uniform mean flow result. The turbulence is generated with the average velocity of 14 m/s. The non-uniform mean profile velocity at 80 % height of the building is 18.3 m/s. This difference in the velocity profile causes notable variations in the induced forces, while the effect of the induced turbulence is rather modest.
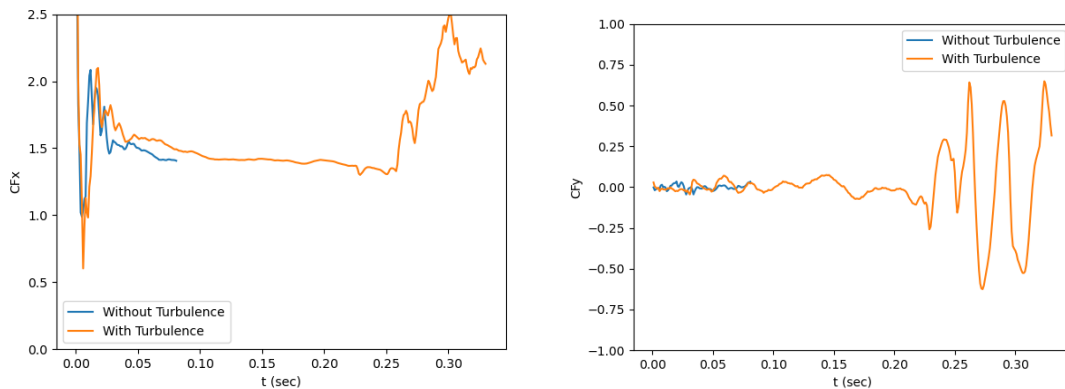
Figure 5.16 Along-wind force coefficient vs. time (s) (left) and across-wind force coefficient vs. time (s) (right) of uniform mean flow without turbulence and with turbulence.

24
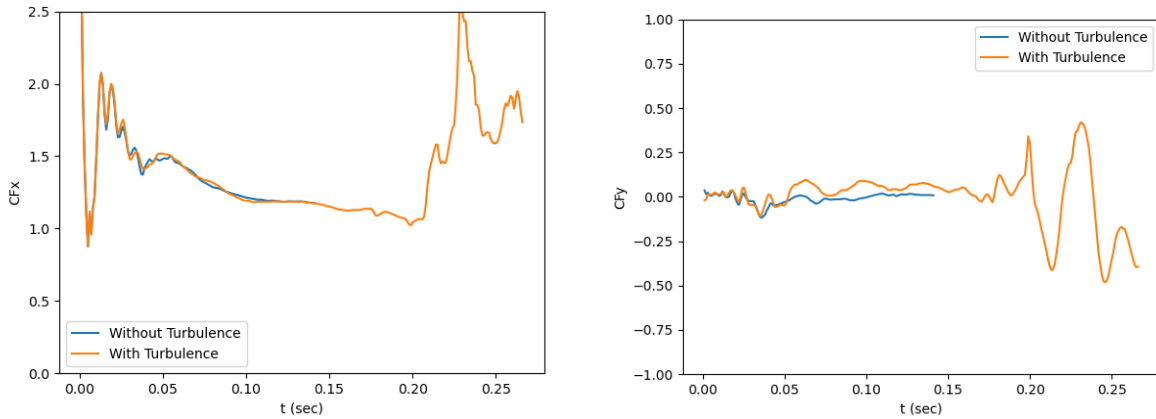
Figure 5.17 Along-wind force coefficient vs. time (s) (left) and across-wind force coefficient vs. time (s) (right) of non-uniform mean flow without turbulence and with turbulence.

5.6   Discussion of Vertiport Flow Characteristics

To examine the characteristics of the unsteady vertiport flow, four monitor points were considered. The first point (in Figure 5.18) is located between the turbulence source and the building, and provides with the reference inflow conditions. The red square in the plot is the position of the momentum source region for the STG-generated turbulence. Figure 5.19 shows the vorticity contours for the non-uniform mean flow with turbulence. The other three points are located at the same vertical elevation above the building rooftop (in the vertiport region) equal to the building span, as shown in Figure 5.20. The second point corresponds to the front ("leading edge") of the building rooftop region, the third one is at the mid-point of the building, and the fourth one is at the back ("trailing edge") of the building rooftop region.

Figure 5.21 shows the time fluctuations of the U-component of the flow velocity normalized by the average value of 14 m/s, while Figure 5.22 shows the corresponding FFT spectra. Clearly, compared to the first (reference) monitor point, the turbulent velocity fluctuations are dramatically increased in amplitude at the vertiport positions, with progressively larger fluctuations (and

corresponding spectral amplitudes) observed towards the trailing edge of the rooftop region. Figures 5.23 and 5.24 show similar characteristics for the V-component of the vertiport flow velocity.

Figure 5.25 illustrates the differences in the vorticity contours at the rooftop region at the time snapshots of 0.22s, 0.23s, 0.24s, and 0.25s, obtained for the uniform vs. non-uniform mean flow profiles. Figure 5.26 shows the predicted fluctuations of the along-wind force coefficient at the rooftop, obtained for turbulent non-uniform mean flow. The rooftop force coefficient averages at about 0.25, which is less than the values obtained at 80% of the building height, as discussed in Section 5.5.

Five spanwise and five chordwise slices of the vertiport region are considered to examine the time snapshots of the vertiport flow velocity, with the slices equally separated by a distance of 0.019 in both directions. First, Figure 5.28 illustrates a time snapshot of the velocity contours at the rooftop for the selected spanwise (left) and chordwise (right) slices indicating a complex rooftop flow evolution pattern characterized by vortical structures and cavity regions forming towards the trailing edge of the vertiport region.

Figure 5.29 and Figure 5.30 shows the midspan U (left plot) and V (right plot) velocity profiles in vertical (z) direction obtained for non-turbulent and turbulent winds, respectively, and compared for 5 chordwise locations from rooftop leading to trailing edge. Without imposed turbulence (Figure 5.29), both velocity components reach the maximum values at the top of the vertiport region equal to 40-50 m/s (compared to the upstream average wind velocity of 14 m/s). This occurs near the leading edge, but subsequently the maximum flow velocities reduce towards the trailing edge of the region, along with the appearance of the recirculation zones near the rooftop surface indicating the formation of the massive vortical structures and shielded (cavity) region

26

towards the trailing edge. The results appear similar in character but quite different quantitatively in the case of the turbulent wind (Figure 5.30) where the formation of even larger recirculation regions is observed.

For completeness, Figures 5.31 and 5.32 present another view of these results, with the midchord vertical distrubtion of U (left) and V (right) velocity profiles for wind without and with turbulence, respectively, for 5 spanwise locations from right to left rooftop edge. Once again, large recirculation zones appear in the core of the vertiport region indicating the formation of large vortical structures.

To characterize the unsteady flow characteristics at the vertiport in more detail, the unsteady flow velocities recorded at the monitor points are further interpreted by decomposing them into the "gust" and "turbulence" velocity components, with the frequency of 35 Hz selected to demarcate the "gust" (low-frequency) and the "turbulence" (high-frequency) content based on the corresponding two clusters of the velocity FFT spectra observed in Figures 5.22 and 5.24.

Figures 5.33 and 5.34 show thus identified "turbulence" and "gust" components of U-velocity, while Figures 5.35 and 5.36 show such results for V-velocity, obtained from the corresponding low- and high-pass filtering of the FFT velocity spectra. Note that the turbulent content in both velocity components consistently increases towards the rooftop trailing edge. On the other hand, the gust velocity peaks at the leading edge for V-component and the mid-roof point for U-component, with both decreasing at the trailing edge due the formed cavity zone following the massive flow separation at the leading edge of the rooftop region.

To further quantify the increase in the flow turbulence fluctuations, the turbulence intensity was calculated at each monitor point for both U and V velocity components, as

27

$$I = u_{rms}/\bar{u} = \sqrt{\overline{u'(t)^2}}/\bar{u} \tag{21}$$

Table 5.2 shows the results for U and Table 5.3 shows the results for V. As expected, the turbulence intensities increase towards the rooftop trailing edge. It should be re-emphasized that the predicted mean velocity fluctuations at the rooftop region can be essentially characterized as the superposition of time-harmonic gust components ("gust modes") of the wind-induced vertiport flow in the computed velocity FFT spectrum, to which the multi-rotor blade aerodynamic and acoustic responses can be predicted using well-established low-fidelity analysis tools including the blade momentum theory. The corresponding blade response to the extracted high-frequency turbulence content can also be computed using the previously developed low-fidelity high-frequency approaches developed for blade-turbulence interaction analyses. Finally, the variations of the instantaneous velocity profiles in Figures 5.29-5.32 indicate the sharp-edge gust effects that also contribute to the unsteady blade responses that can be predicted using low-fidelity (including analytical) approaches, in the spirit indicated for all types of canonical disturbance forms in Ref. [11].

Figure 5.18 Monitor point No. 1 (Mid-point of turbulence source and building) and position of turbulence source (red square) and building.
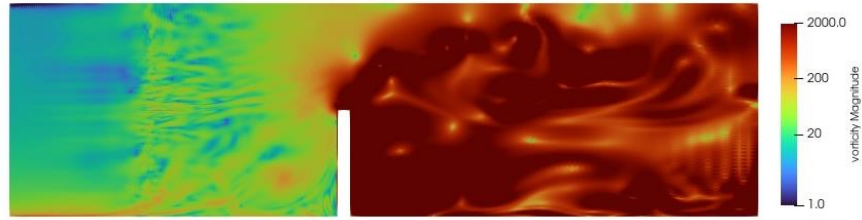
Figure 5.19 Vorticity contour of non-uniform mean flow, with upstream turbulence.
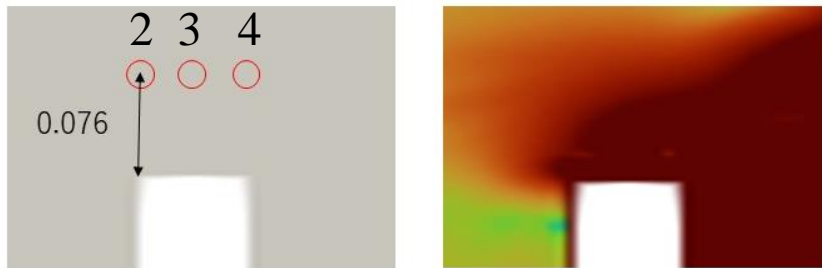
Figure 5.20 Monitor points No. 2 (leading edge of the roof), 3 (mid-point of the roof), and 4 (trailing edge of the roof) above building and vorticity contour of non-uniform mean flow, with upstream turbulence.
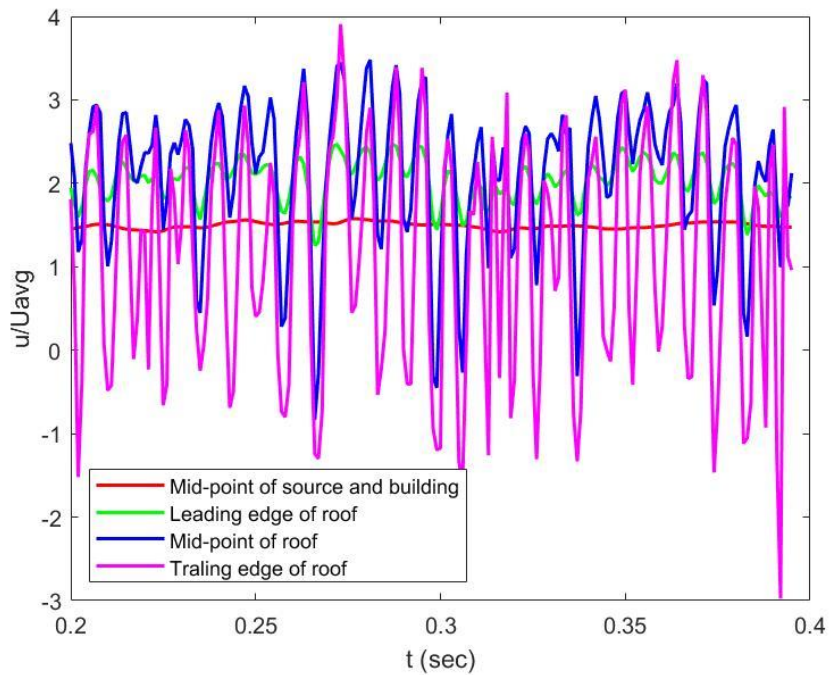
Figure 5.21 U velocity time history at Mid-point, Leading edge, Mid-roof and Trailing edge monitor points, with upstream turbulence.

29
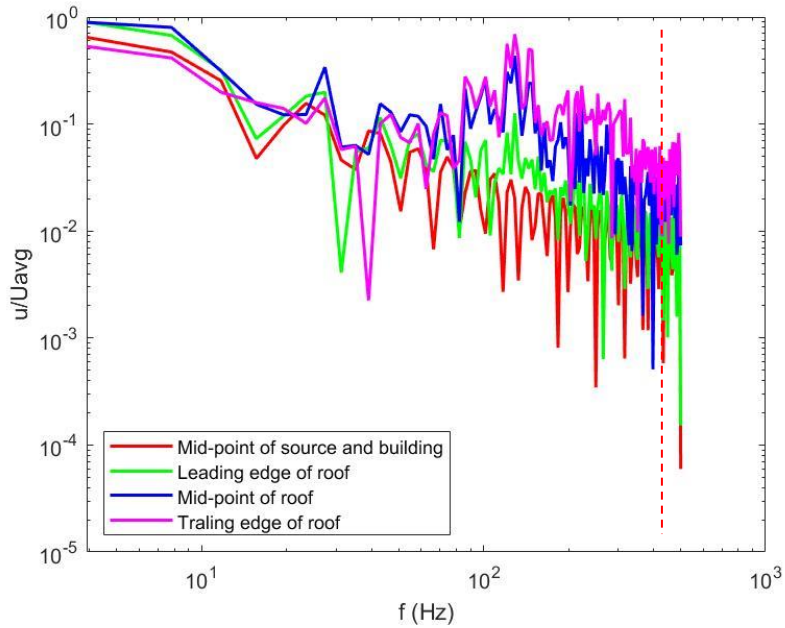
Figure 5.22 U-velocity FFT of Mid-point, Leading edge, Mid-roof and Trailing edge monitor points, with upstream turbulence.
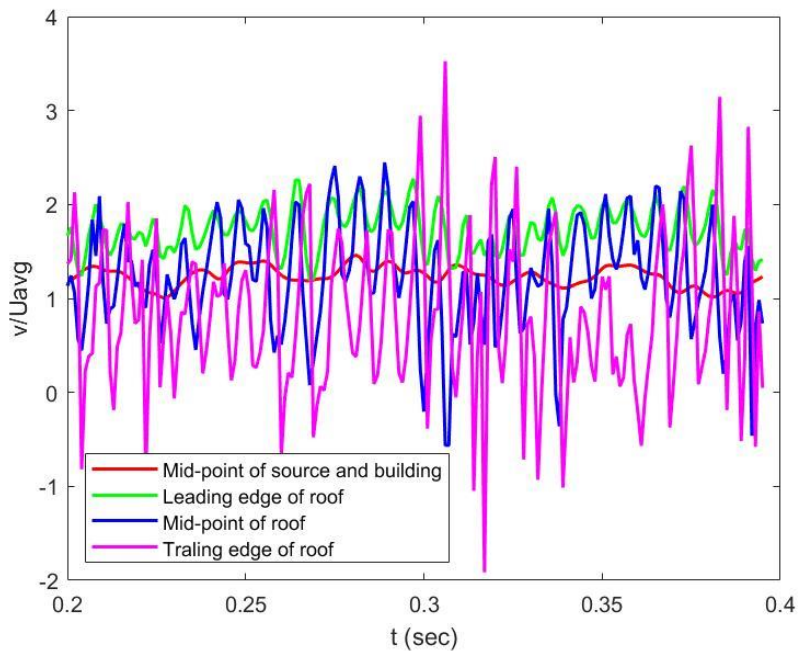
Figure 5.23 V-velocity time history at Mid-point, Leading edge, Mid-roof and Trailing edge monitor points, with upstream turbulence.
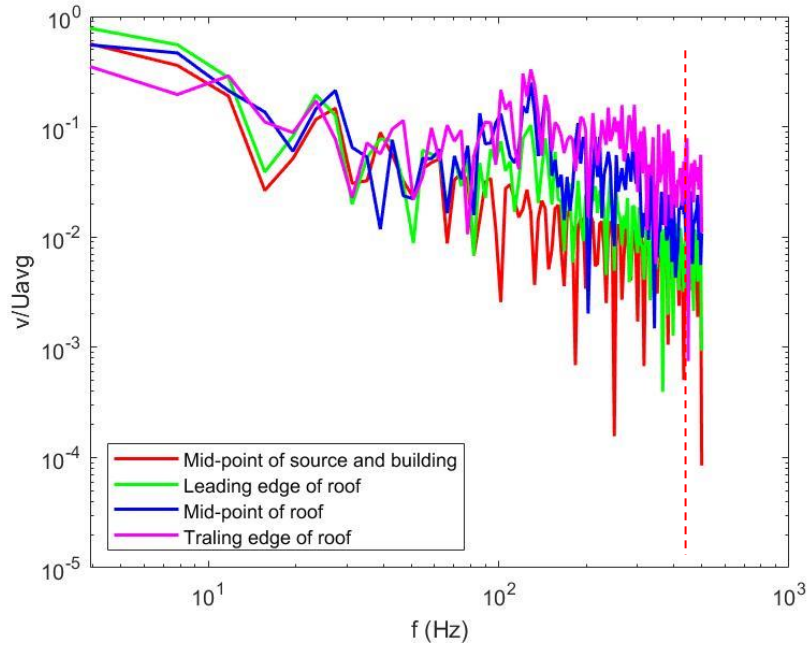
Figure 5.24 V-velocity FFT at Mid-point, Leading edge, Mid-roof and Trailing edge monitor points, with upstream turbulence.
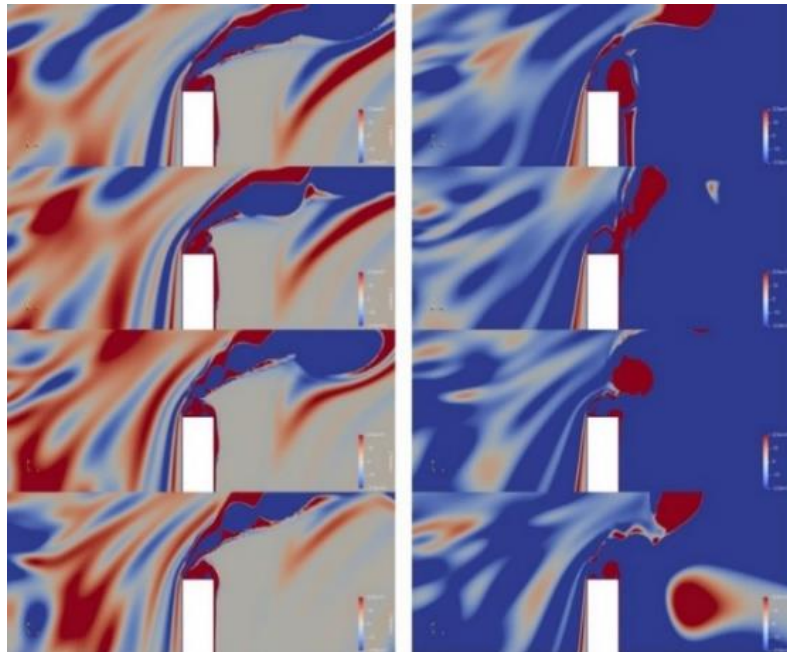
Figure 5.25 Vorticity around roof top with uniform (left) and non-uniform (right) mean inflow, with upstream turbulence.
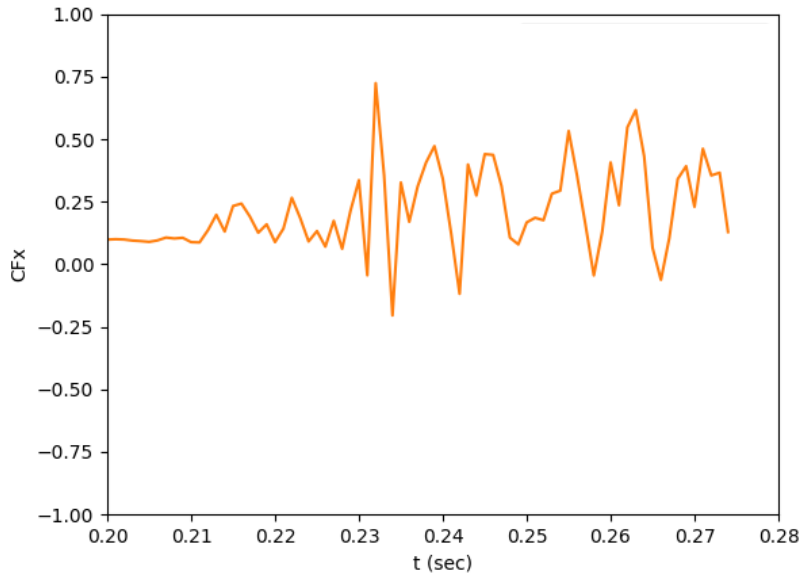
Figure 5.26 Along-wind force coefficient vs. time (s) near rooftop with non-uniform mean flow, with upstream turbulence.
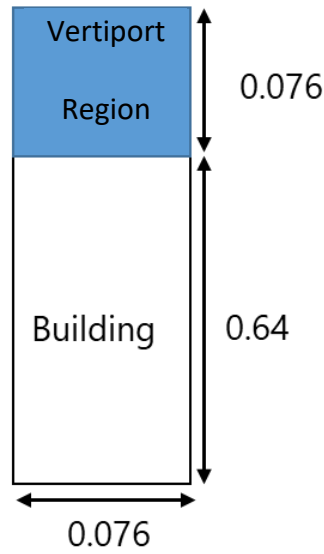
Figure 5.27 Rooftop region for analysis of vertiport unsteady velocity dynamics.
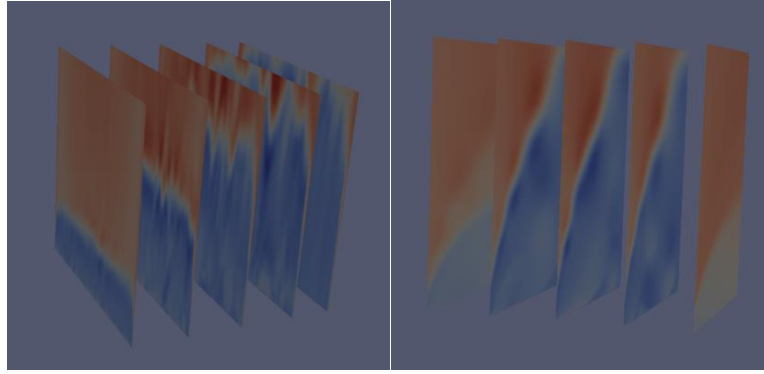
Figure 5.28 Velocity contours for rooftop spanwise (left) and chordwise (right) slices.
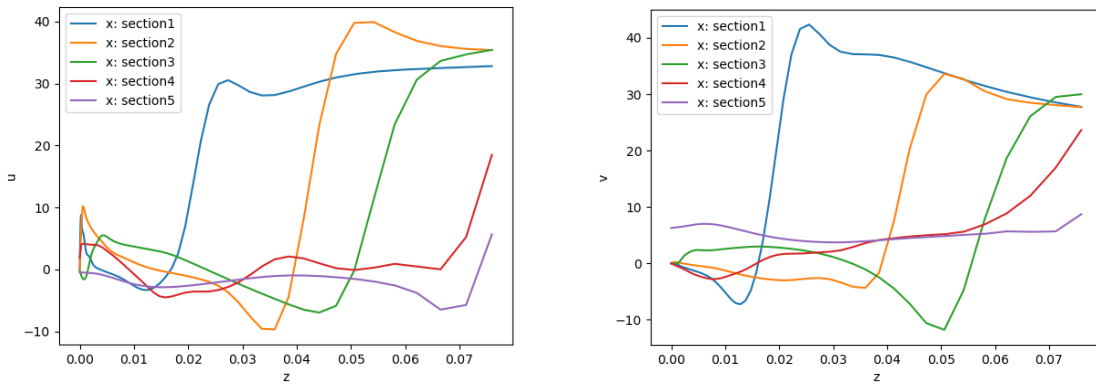
Figure 5.29 Midspan U (left) and V (right) velocity profiles for wind *without* turbulence, for 5 chordwise locations from rooftop leading to trailing edge.
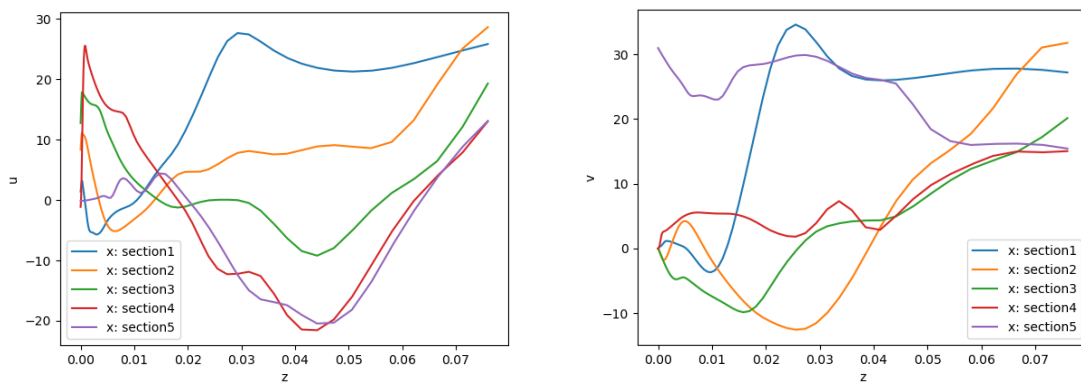
Figure 5.30 Midspan U (left) and V (right) velocity profiles for wind *with* turbulence, for 5 chordwise locations from rooftop leading to trailing edge.
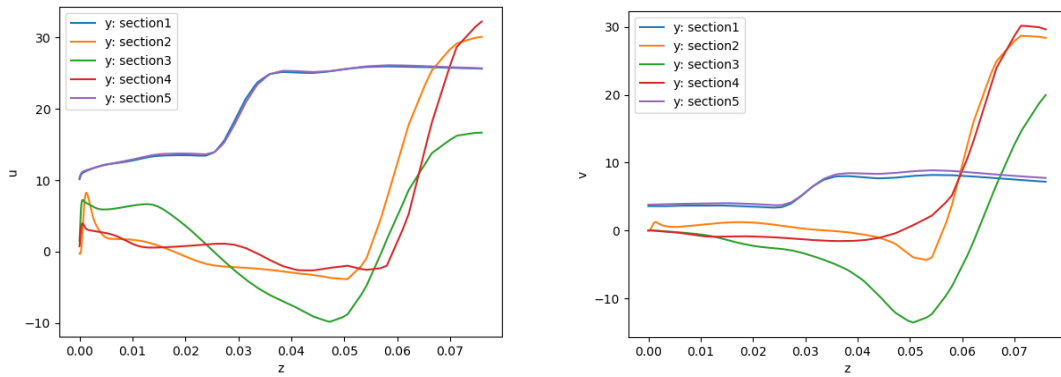
Figure 5.31 Midchord U (left) and V (right) velocity profiles for wind without turbulence, for 5 spanwise locations from right to left rooftop edge.
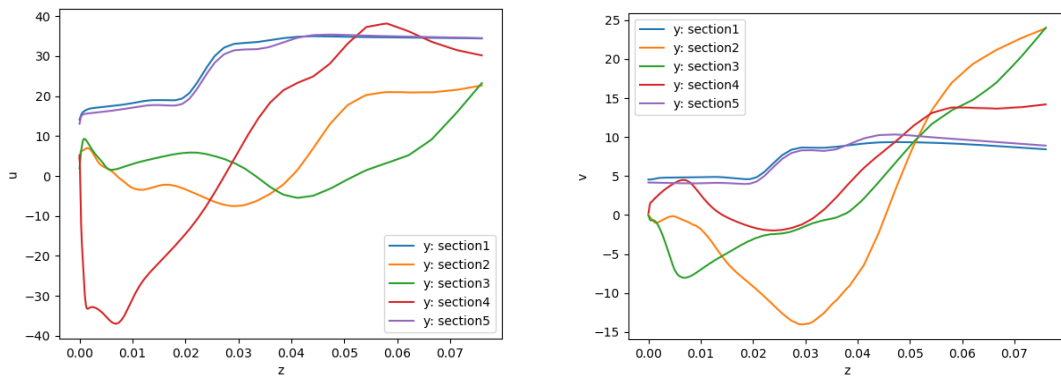
Figure 5.32 Midchord U (left) and V (right) velocity profiles for wind with turbulence, for 5 spanwise locations from right to left rooftop edge.
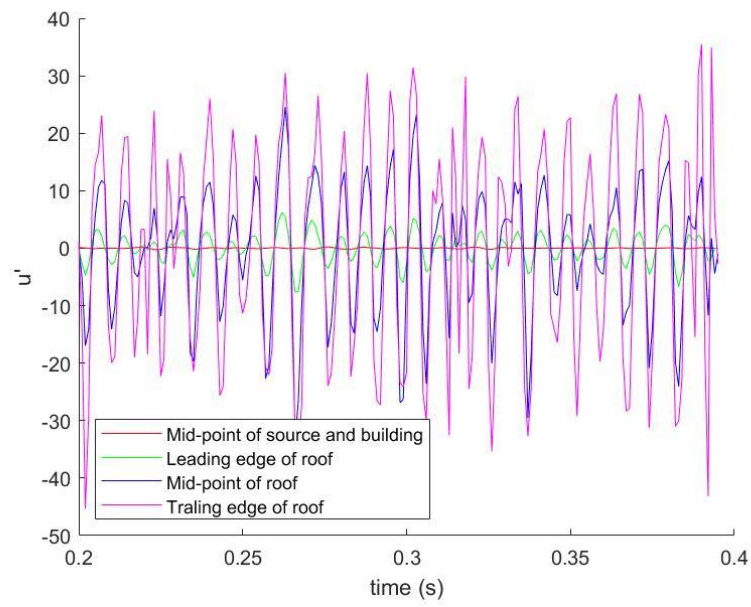
Figure 5.33 Turbulence component of U velocity.
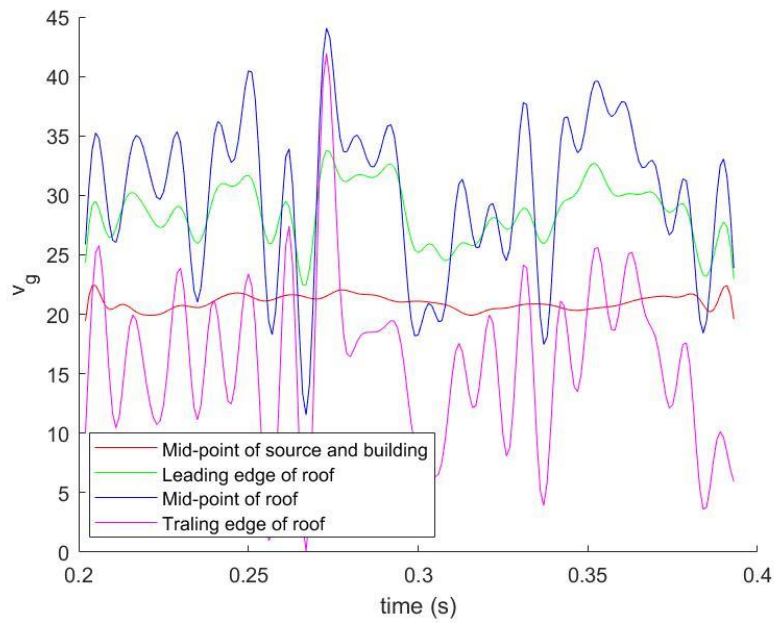
Figure 5.34 Gust component of U velocity.
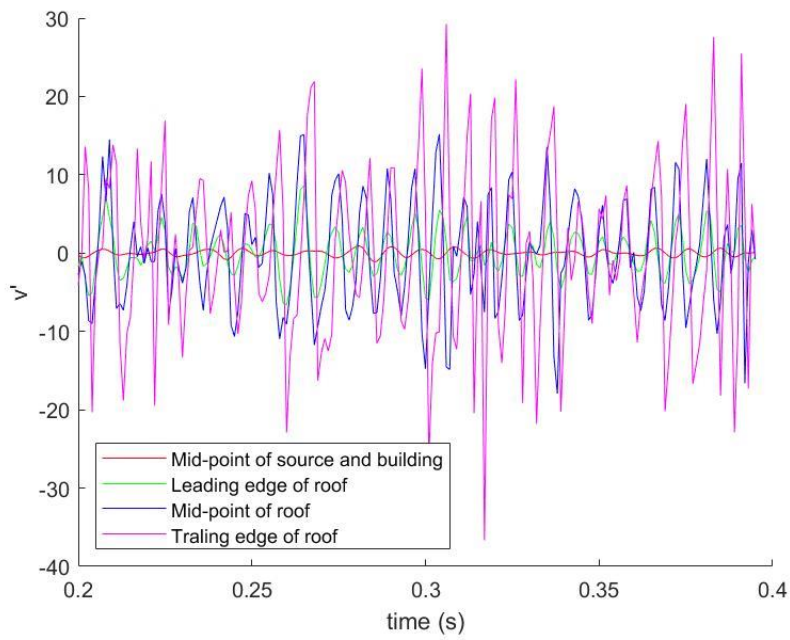
Figure 5.35 Turbulence component of V velocity.
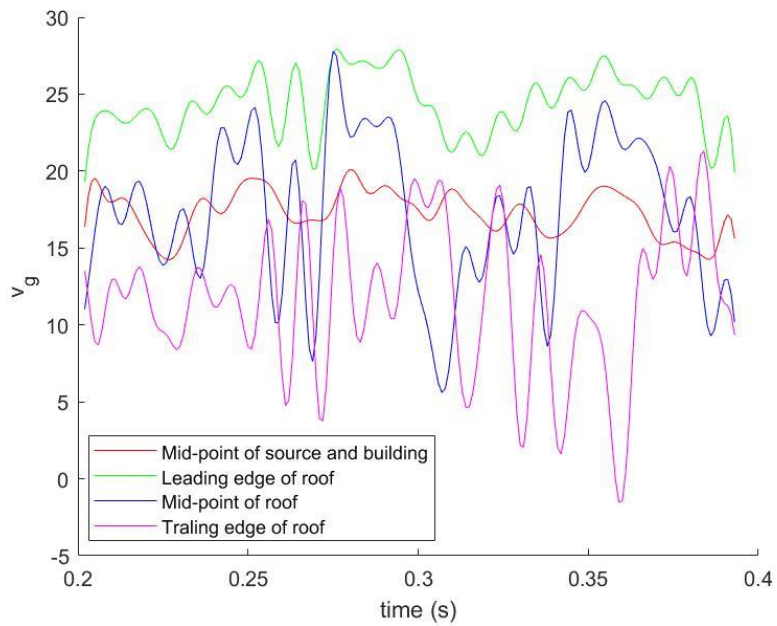
Figure 5.36 Gust component of V velocity.

Table 5.2 Turbulence character of u velocity

	Turbulence Intensity
Mid-point of source and building	0.025
Leading edge of roof	0.122
Mid-point of roof	0.398
Trailing edge of roof	1.231

Table 5.3 Turbulence character of v velocity

	Turbulence Intensity
Mid-point of source and building	0.084
Leading edge of roof	0.136
Mid-point of roof	0.470
Trailing edge of roof	1.004

6   Conclusions and Future Work Recommendations

This study employed an open-source Navier-Stokes OpenFOAM solver to model the turbulent wind flow around a building, with objective to characterize the unsteady flow environment at the building rooftop that could serve as a vertiport for future multi-rotor eVTOL vehicle operations.

The OpenFOAM model was first validated against previous ANSYS Fluent results using 2D unsteady gust-airfoil interaction benchmark study. The predicted surface pressure matched well with the previous results, which validated the current numerical approach.

For the benchmark validation case study of the flow around a building, the obtained across-wind force spectrum with STG model slightly overpredicts both the experimental and DSRFG results at the higher frequencies. However, the overall comparison is quite satisfactory and provides a successful validation of the current model.

The differences in results obtained for uniform mean flow profile with and without imposed turbulence, and non-uniform mean flow profile with and without imposed turbulence, were examined. In particular, the flow non-uniformity resulted in notable variation in the induced building surface forces, while the effect of the induced turbulence was rather modest.

Four monitor points were considered to examine the characteristics of the unsteady vertiport flow. The turbulent velocity fluctuations dramatically increased in amplitude at the vertiport positions, with progressively larger amplitudes observed towards the trailing edge of the rooftop region. Similarly, the gust amplitudes obtained as the variation of the mean of the vertiport flow velocity, also increased from leading edge.

For the instantaneous vertiport flow analysis, five chordwise and five spanwise slices of the vertiport region were considered to examine the mean flow velocity profiles in the vertical direction from the rooftop surface. In general, the flow velocity reach the maximum values at the

top of the vertiport region equal to 40-50 m/s (compared to the upstream average wind velocity of 14 m/s). This occurred near the leading edge of the region, but subsequently the maximum flow velocities reduced towards the trailing edge of the region, along with the appearance of the recirculation zones near the rooftop surface indicating the formation of the massive vortical structures and shielded (cavity) region towards the trailing edge.

For a more detailed characterization of the vertiport flows, advanced flow decomposition techniques may be employed in the future. For instance, the dynamic mode decomposition (DMD) [8] provides a means to decompose a time-resolved data into energetic modes, with each mode having a characteristic frequency and growth/decay rate. DMD is based on the eigenmode decomposition of a best-fit linear operator that approximates the dynamics present in the unsteady data. Another related method is the proper orthogonal decomposition (POD) [9] that gives an orthogonal set of basis vectors and is useful in constructing a reduced-order model of the flow field. A variants of POD, the spectral proper orthogonal decomposition (SPOD) [10] finds an optimal orthogonal basis for the data in the sense that a subset of such modes captures a larger fraction of the total energy (variance) in the data than any other orthogonal basis. The difference between SPOD and normal POD is that the modes may vary in both space and time and are orthogonal under a space–time inner product. As a result, they are optimal in expressing spatiotemporal coherence in the data. The outlined methods could be employed in the future, more detailed analysis of the vertiport flowfield for the purpose of developing effective ROM based safety and noise control strategies.

7   REFERENCES

[1] Sheehan, V., "Press release: VFS Electric Vtol Directory Hits 600 concepts," The Vertical Flight Society Available: https://vtol.org/news/press-release-vfs-electric-vtol-directory-hits-600-concepts

[2] WINDCRANE" Wind speed patterns in a city with buildings Available: https://www.windcrane.com/blog/construction/wind-speed-patterns-city-buildings

[3] Kazarina, M., Nguyen, L. D., Golubev, V. V., Borener, S., Hufty, D., Pasiliao, C. L., and Visbal, M., "Effects of 3D time-harmonic gust and turbulence on unsteady aerodynamic responses of loaded low-speed airfoils," 54th AIAA Aerospace Sciences Meeting, 2016

[4] Rainhold, Timothy A., "Measurement of simultaneous fluctuating loads at multiple levels on a model of a tall building in a simulated urban boundary layer," 1977

[5] Huang, S. H., Li, Q. S., and Wu, J. R., "A general inflow turbulence generator for large eddy simulation," Journal of Wind Engineering and Industrial Aerodynamics, vol. 98, 2010, pp. 600-617

[6] Kraochnan, H. R., "Diffusion by a random velocity field," *The Physics of Fluids*, 13(1), 22-31

[7] Marina Kazarina, "A volume – force synthetic disturbance approach for high – fidelity modeling of unsteady fluid structure interaction," 2019

[8] Kunihiko Taira, Steven L. Brunton, Scott T. M. Dawson, Clarence W. Rowley, Tim Colonius, Beverley J. McKeon, Oliver T. Schmidt, Stanislav Gordeyev, Vassilios Theofilis, Lawrence S. Ukeiley, "Modal Analysis of Fluid Flows: An Overview," February 7, 2017

[9] Anindya Chatterjee, "An Introduction to the Proper Orthogonal Decomposition," CURRENT SCIENCE, VOL.78, NO. 7, 10 APRIL 2000

[10] Oliver Schmidt, Tim Colonius, "Guide to Spectral Proper Orthogonal Decomposition," AIAA JOURNAL, 2020

[11 Lap Nguyen, Vladimir Golubev, Miguel Visbal, "Numerical Study of Transitional SD7003 Airfoil Interacting with Canonical Upstream Flow Disturbances*," AIAA JOURNAL, 2017

8   APPENDIX - A

Source code of turbulence

```
scalar GVEL=14;
scalar Pi = constant::mathematical::pi;
scalar chord = 1;
label  nHarm;
scalar XGS;
scalar YGS;
scalar BGS = 0/chord;
scalar dts = 0;
scalar dth = 1000;

scalar GALF;
scalar GBET;
scalar GGAM;
scalar GFUN = 0;
scalar GFUNX = 0;
scalar GTFUN=0;
scalar GCOMF=0;
scalar UTERM=0;
scalar VTERM=0;
scalar WTERM=0;
scalar x_loc;
scalar y_loc;
scalar z_loc;
scalar ZP;
scalar PHASE;

///*
scalar GKX[392] = {};
scalar GKY[392] = {};
scalar GKZ[392] = {};
scalar GAX[392] = {};
scalar GAY[392] = {};
scalar GAZ[392] = {};
scalar GBX[392] = {};
scalar GBY[392] = {};
scalar GBZ[392] = {};
//*/

scalar GOMEGA;
scalar RHO = 1;
scalar centroid[2];
scalar source_x = 0;
scalar source_y = 0;
scalar source_z = 0;
scalar flowTime=runTime.value();

      if(flowTime>=dts && flowTime<=dth)
```

```
{
        forAll(mesh.C(), celli)
        { source_x = 0;
         source_y = 0;
         source_z = 0;
          vector Ci = mesh.C()[celli];

        nHarm=0;
        while (nHarm<392)
        {
                GALF=GKX[nHarm];
                GBET=GKY[nHarm];
                GGAM=GKZ[nHarm];
                GOMEGA = GALF*GVEL;
                BGS = 2 * GALF;
                XGS = 0.1 * chord - Pi / BGS;

                YGS = 1.27;

                x_loc=Ci.x();
                y_loc=Ci.y();
                z_loc=Ci.z();

                ZP=0;

                PHASE=GOMEGA*(flowTime-dts)-GBET*y_loc-
GGAM*z_loc-GALF*XGS;

                if (fabs(x_loc-XGS)<Pi/BGS)
                {

                        GFUN=0.5*(1+Foam::cos(BGS*(x_loc-XGS)));
                        GFUNX=-0.5*BGS*Foam::sin(BGS*(x_loc-XGS));
                        GTFUN=0.5*(Foam::tanh(100*(y_loc+YGS))-
Foam::tanh(100*(y_loc-YGS)));
                        GCOMF=(Foam::pow(GALF,2)-
Foam::pow(BGS,2))/(Foam::pow(BGS,2)*Foam::sin(GOMEGA*Pi/GVEL/BGS));

                        UTERM=-
GOMEGA*GCOMF*GTFUN*GFUN*(GAX[nHarm]*Foam::cos(PHASE)-
GBX[nHarm]*Foam::sin(PHASE));
                        source_x=source_x+RHO*UTERM;

VTERM=GVEL*GCOMF*GTFUN*GFUNX*(GAY[nHarm]*Foam::sin(PHASE)+GBY[nHarm]*F
oam::cos(PHASE));
                                source_y=source_y+RHO*VTERM;

WTERM=GVEL*GCOMF*GTFUN*GFUNX*(GAZ[nHarm]*Foam::sin(PHASE)+GBZ[nHarm]*F
oam::cos(PHASE));
                                source_z=source_z+RHO*WTERM;
                        source_z = 0;
```

```
                }
                nHarm++;
        }
    dimensionedVector SSS1("SSS1", dimAcceleration,
vector(source_x,source_y,source_z));
    //dimensionedVector SSS1("SSS1", dimAcceleration,
vector(source_z,source_x,source_y));
            dimensionedVector SSS2("SSS2",
dimAcceleration, vector(0,0,0));
    SSS[celli]=SSS1.value();
      }
  }
        else
    {
        source_x=0;
            source_y=0;
        source_z=0;
    }
```

9 APPENDIX - B

STG code

```
spectr = 5;
InputParameters
Spectrum = spectr; % DELETE IF WANT TO TURN OFF USER INTERFACE

A = (turbI*Uavg);
urms = turbI*Uavg;
TKE = 1/2 * (Uavg * turbI)^2;   % Page 118
L_e = 300;
EDR_ = TKE^(3/2)/L_e;  % modelrate 0.045
ft^2/sec^3=0.045*0.3048^2=0.0042

% Experimental results
cmich = .1/10;
Lmich = .0035;
 % Turbulent Length Scale in m
% Lt = 100*7.5/1000;%0.1;%Lmich*chord/cmich
% Lt = 300*0.0762

% Lt = 100*10/1000;%0.1;%Lmich*chord/cmich

% Lt = 100*76.2/1000; %Lmich*chord/cmich   %%% real
Lt = 300   %%%% real2

if LP == 'Y'
    ni = 200;
end
if LP == 'N'
    % number of grid point in x direction
    ni = 2;
    yc=0;
end
% space point where timehistory of fluctuating velocity is taken for
% plotting spectrum E(f)
x_coord = 1;
% number of grid point in y direction
nj = 2;
% number of grid point in z direction
nk = 2;
%%
% Ref 4.
% generate x-grid for lineplots
xp(1) = -3;
dx = (3-(-3))/ni;

for k = 2:ni
   xp(k) = xp(k-1) + dx;
end
```

```
% generate y-grid
yp(1)=0;
yp(2)=0.0004;
% generate y-grid
zp(1)=0;
zp(2)=0.0004;
% maximum wavenumber that can be resolved with the grid used
k_cut = 2*pi / l_cut;
% wavenumber that corresponds to the peak of E(k) spectrum
k_E_max = 2 * pi / Lt;
% checkAdamian = 0;% 1 if take spectrum equation from Adamian, 1 if
take spectrum from Analytical reconstruction of
% % isotropic turbulence spectra based on the Gaussian transform A.
Wohlbrandt ,N. Hub ,S. Gu rina ,R. Ewertb
% if checkAdamian == 0
%     k_E_max = gamma(5/6)/gamma(1/3)*sqrt(pi);
% end
% time scale
tau = Lt / Uavg;% Lt / Uavg;%dtND * chord/Uavg;
dt = dtND;
% empiric constant Ref. 1
beta = 0.5;
% lowest WN Ref.1
k_min = k_E_max * beta;
% highest WN Ref.1
k_max = 1.5 * k_cut;
% determin number of modes N Ref.1
N = 1;
k_wn(1) = k_min;
while k_wn(N) < k_max
  N = N+1;
  % wavenumbers form geometric series Ref.1
  k_wn(N) = k_min * (1+alpha)^(N-1);
end
k_wn1 = k_wn';
% DAVIDSON
%% RANDOM NUMBERS GENERATION code is taken from Ref.4
% create a seed from time (must be negative)
if (new_realization == 1)  && (Spectrum ==5)
    v=clock;
    iseed=-(v(4)+v(5)+v(6));
    % initialization
    wnr=zeros(N,1);
    fi=zeros(N,1);
    teta=zeros(N,1);
    psi=zeros(N,1);
    alfa=zeros(N,1);

    iy=0;
    iv=zeros(32,1);

    [fi,psi,alfa,teta,iy,iv,iseed] = angles(N,iseed,iy,iv);
```

```matlab
    save(['forFDL3DI/','psi.mat'],'psi')
    save(['forFDL3DI/','teta.mat'],'teta')
    save(['forFDL3DI/','alfa.mat'],'alfa')
    save(['forFDL3DI/','fi.mat'],'fi')

else

    load(['forFDL3DI/','psi.mat'])
    load(['forFDL3DI/','teta.mat'])
    load(['forFDL3DI/','alfa.mat'])
    load(['forFDL3DI/','fi.mat'])

end

for m=1:N
     % unit direction of a wavenumber in Ref.1 denoted as d
    kxio(m)=sin(teta(m))*cos(fi(m));
    kyio(m)=sin(teta(m))*sin(fi(m));
    kzio(m)=cos(teta(m));
    % unit direction which gives the direction of a mode
    sxio(m)=cos(fi(m))*cos(teta(m))*cos(alfa(m))-
sin(fi(m))*sin(alfa(m));

syio(m)=sin(fi(m))*cos(teta(m))*cos(alfa(m))+cos(fi(m))*sin(alfa(m));
    szio(m)=-sin(teta(m))*cos(alfa(m));
end


for m = 1:N
  if m >= 2
      dk(m) = k_wn(m) - k_wn(m - 1);
  end
end
% wavenumber normalized by k_E_max that corresponds to the E(k) peak
k_ND = k_wn / k_E_max;
ut = Uavg*turbI;
switch Spectrum
    case 4 % RFG Gaussian Ref
        E = (urms^2)*(1/3)*16*sqrt(2/pi)*(k_ND.^4).*exp(-2*k_ND.^2);
    case 5 % Adamian Gaussian???karman????
        E1 = 55 / 9 *ut^2*(k_ND).^4./((1+2.4*(k_ND).^2).^(17/6));%
Adamian change k_E_max parameter!!!!
%         k_E_max = gamma(5/6)/gamma(1/3)*sqrt(pi);
        kstar = k_wn*Lt;k_ND = kstar/k_E_max;
%         k_ND = k_wn / k_E_max;
        E = 55*pi / 9 *ut^2*Lt*(k_ND).^4./((1+(k_ND).^2).^(17/6));%
Adamian change k_E_max parameter!!!!
    case 6 % Liepman
%         k_E_max = sqrt(pi)*gamma(5/6)/gamma(1/3);
        k_NDstar = k_wn*Lt;
```

```
        E = 8 / pi *ut^2*Lt*(k_NDstar).^4./((1+(k_NDstar).^2).^(3));
    otherwise
        error('Choose spectrum')
end
% normalized amplitude Ref. 5
qn = E .* dk / sum(E .* dk);

Ax = 2 * sqrt(3/2) * qn.^0.5 .* sxio .* cos(psi);
Ay = 2 * sqrt(3/2) * qn.^0.5 .* syio .* cos(psi);
Az = 2 * sqrt(3/2) * qn.^0.5 .* szio .* cos(psi);
Bx = - 2 * sqrt(3/2) * qn.^0.5 .* sxio .* sin(psi);
By = - 2 * sqrt(3/2) * qn.^0.5 .* syio .* sin(psi);
Bz = - 2 * sqrt(3/2) * qn.^0.5 .* szio .* sin(psi);
% kx = kxio.*k_wn;
% ky = kyio.*k_wn;
% kz = kzio.*k_wn;

switch Spectrum
    case 4
        kx = kxio.*k_ND;
        ky = kyio.*k_ND;
        kz = kzio.*k_ND;
    case 5
        kx = kxio.*k_ND;
        ky = kyio.*k_ND;
        kz = kzio.*k_ND;
    case 6
        kx = kxio.*k_ND;
        ky = kyio.*k_ND;
        kz = kzio.*k_ND;
    otherwise
        error('Choose spectrum')
end


% convection
s = kx*Uavg*tau;
% coefficient for rescaling of constants
RescCoef = 1;
%
RescalingForFDL3DI(chord,Lt,turbI,kx,ky,kz,Ax,Ay,Az,Bx,By,Bz,tau,Uavg,
RescCoef,Spectrum)
RescalingForAnsys(chord,Lt,turbI,kx,ky,kz,Ax,Ay,Az,Bx,By,Bz,tau,Uavg,R
escCoef,Spectrum,A)
% Used if a previously generated file used for FDL3DI is desired to be
read
scale1 = chord/Lt;
scale2 = turbI;
switch Spectrum
        case 4
        turbFile = 'STG_Gaussian.dat';
    case 5
```

```
            turbFile = 'STG_VonKarmanAnsys.dat';
%           turbFile = 'STG_VonKarmanAnsys2.dat';
      case 6
            turbFile = 'STG_LiepmanAnsys.dat';
      otherwise
            error('Choose spectrum')
end
%
% turbData = load(['forAnsys/',turbFile]);
% kx = turbData(:,1)%/scale1^RescCoef;
% ky = turbData(:,2)%/scale1^RescCoef;
% kz = turbData(:,3)%/scale1^RescCoef;
% Ax = turbData(:,4)%/scale2;
% Bx = turbData(:,5)%/scale2;
% Ay = turbData(:,6)%/scale2;
% By = turbData(:,7)%/scale2;
% Az = turbData(:,8)%/scale2;
% Bz = turbData(:,9)%/scale2;
% omega = k(:,1);
% clear turbData
%
t(1) = -dt;
u_mode = zeros(N,nstep);
for it = 1:nstep % time loop
    t(it+1) = t(it) + dt;
    if rem(it,100) == 0
        disp(['time step no: ',num2str(it)])
    end
    %% DAVIDSON
    for i = 1:ni - 1
        for k = 1:nk - 1
            for j = 1:nj - 1
    %     use  when generate spectra cell center coordinates when
generate spectra use
            xc(i) = 0.5 * ( xp(i) + xp(i + 1));
            zc=0;
    %       initiate turbulent velocities to zero.
            utrp=0;
            vtrp=0;
            wtrp=0;
                for m=1:N % loop over modes
                    arg_S = kx(m) * xc(i)+ky(m) * yc+kz(m) * zc + s(m)
* t(it+1)/ tau;
                    cos_fun = cos(arg_S);
                    sin_fun = sin(arg_S);
                    utrp = utrp + Ax(m) * cos_fun + Bx(m) * sin_fun;
                    vtrp = vtrp + Ay(m) * cos_fun + By(m) * sin_fun;
                    wtrp = wtrp + Az(m) * cos_fun + Bz(m) * sin_fun;
                    u_mode(m,it) = Ax(m) * cos_fun + Bx(m) * sin_fun;

                end  % end m=1:N
    % store the fluctuations u, v, w in 2D arrays
```

```
        u(j,i) = utrp;
        v(j,i) = vtrp;
        w(j,i) = wtrp;
    % store the fluctuations for each time step it (we have only one
cell in y-dirrection; j=1)
    % these will be used for plotting and evaluation below
        ut(i,it) = A * u(j,i);
        vt(i,it) = A * v(j,i);
        wt(i,it) = A * w(j,i);
          end % end j=1:nj-1
       end % end k=1:nk-1
    end% end i = 1:ni-1
end
```