

Multi-Scale CFD Modelling Towards Resolving Machined Roughness



Marios Kapsis

Linacre College
University of Oxford

*A thesis submitted for the degree of Doctor of Philosophy in Engineering
Science*

2019

Acknowledgements

As I reach the end of my DPhil, it seems appropriate to acknowledge those who have contributed the most and provided assistance during these years.

Firstly, I am grateful to my supervisor, Professor Li He, for his patience, guidance, and continuous efforts to help me become a better engineer and to grow as a person.

Of course, this project would not be feasible without the support of Siemens Industrial Turbomachinery Ltd and particularly of Dr Yan Sheng Li, Mr Roger Wells, and Dr Omar Valero.

It was the Engineering and Physical Sciences Research Council with its Centre of Doctoral Training in Gas Turbine Aerodynamics that provided the relevant background, and for this I am thankful to all those who made it happen.

Also, I would like to thank Dr Amir Ibrahim, Dr Alexander Bucknell, Dr Alexander Murray, and Dr Ian Bush for their valuable input and support.

Finally, I am grateful to my dear parents and friends for their continuous support.

Abstract

Multi-scale CFD Modelling Towards Resolving Machined Roughness

Marios Kapsis

Linacre College

Doctor of Philosophy in Engineering Science

2019

Recent advances in manufacturing technologies, such as additive manufacturing, have introduced the potential of selecting the roughness characteristics as a design parameter. Hence, the understanding and prediction of aerothermal effects of manufacturable (machined) micro-structures would be of paramount importance. So far, roughness has been largely considered as a stochastic attribute and, therefore, had been empirically modelled. Thus, the question arises of whether the shape and pattern of regular roughness and the fine flow structures around them require detailed resolution.

Initially, a systematic computational study is carried out on the aerothermal behaviour and impact of some discrete micro-structures. Micro-structure configurations are considered for a Reynolds number range typical for those roughness elements ($Re < 5000$). Several RANS and LES analyses are performed to compare the behaviour in a well-established turbulent flow regime to a transitional one. The results highlight the significance of resolving such fine scales and their interactive wakes, as they are dependent on both the shape and the pattern of the micro-structures.

Considering the need for resolving both the geometric and flow fine scales efficiently, a new multi-scale approach, the novel Block Spectral Method, is adopted. This method aims to provide efficient resolution of the local spatial and temporal flow variations due to the large scale micro-structures. This resolution is provided without resorting to the modelling of every single micro-structure in detail, as a conventional large scale CFD simulation would demand, but still demonstrating similar time-accurate and time-averaged flow properties. The emphasis is then put into developing a parallelised solver of the method to enable tackling large problems. The present work also includes a first of the kind verification of the Block Spectral Method application to surfaces fitted with 3D micro-structures that promote unsteadiness in the flow.

Table of contents

Nomenclature	x
1 Introduction	1
1.1 Background and Motivation	1
1.1.1 Additive Manufacturing: a New Era	1
1.1.2 Additive Manufacturing and Surface Roughness	3
1.2 Outline of Thesis	9
2 Literature Review	11
2.1 Conventional Roughness Parameters	11
2.2 Aerothermal Effect of Roughness	15
2.3 Studies on Roughness Parameters	18
2.4 Conventional CFD Modelling of Roughness Effects	24
2.5 Multi-Scale Methods	27
2.6 Remarks	29
3 Study of Shape and Pattern Effects of Manufacturable Micro-Structures	32
3.1 Preamble	33
3.2 Computational Configuration	35

3.3	Mesh/Turbulence Modelling Sensitivity	40
3.4	Role of Non-Dimensional Spacing $\mathbf{R_a}/\lambda$	44
3.5	Effect of Shape and Spacing	48
3.5.1	Shape Effect for Matched Relative Spacing $\mathbf{R_a}/\lambda$	48
3.5.2	Directionality of Spacing Effect	49
3.6	ILES Analysis for Transitional Flow Regime	52
3.7	Remarks	58
4	Present Computational Methodology	61
4.1	Baseline Solver	62
4.1.1	General Information	62
4.1.2	Data Storage	64
4.2	Embedded Dual-Mesh Multi-Scale Approach	66
4.2.1	Mesh Generation	66
4.2.2	Dual-Mesh Coupling	69
4.3	Block Spectral Method	73
4.3.1	Time-Averaged Flow Decomposition	75
4.3.2	Correction Terms	77
4.3.3	Propagation of the Total Source Terms	82
4.3.4	Computational Gain	87
4.4	Parallelisation	89
4.5	Remarks	89
5	Validation of the Baseline Solver	91
5.1	Computational Configuration	91

5.2	Multi-Block Test Cases	92
5.3	Dual-Mesh Test Case	94
5.4	Micro-Structures in Channel Flow	97
5.4.1	Experimental Configuration	98
5.4.2	Computational Configuration	100
5.4.3	Results	106
5.5	Remarks	122
6	Validation and Demonstration of Block Spectral Method	124
6.1	Unsteady 3D Flow Subject to Regular Roughness	124
6.1.1	Block Spectral Mapping Applied in the Freestream (X) Direction	125
6.1.2	Block Spectral Mapping Applied in the Lateral (Y) Direction	136
6.1.3	Block Spectral Mapping Applied in Both (X-Y) Directions	139
6.2	Heat Transfer Cases	145
6.2.1	Trapezoidal Ribs	145
6.2.2	Micro-Structures in Channel Flow	150
6.3	Block Spectral Method Scalability	152
6.4	Remarks	153
7	Conclusions and Future Work	156
7.1	Conclusions	156
7.2	Future Work	159
	References	161
	Appendix A Validation Cases for Chapter 3	173
A.1	Computational Configuration	173

A.2	Metrics	174
A.3	Mesh Sensitivity for Validation Cases	176
A.4	2D Single Cylinder at Low Reynolds Number	177
A.5	2D Staggered Cylinders at Low Reynolds Number	177
A.6	3D Staggered Cylinders at Low Reynolds Number	180
Appendix B Advection Upstream Splitting Method		182
Appendix C Parallelisation of the Embedded Dual-Mesh Multi-Scale Solver		187
C.1	Introductory Remarks	187
C.1.1	Hardware Architecture	187
C.1.2	Message Passage Interface	189
C.1.3	Blocking and Non-Blocking Communication	189
C.2	Halo Updating	191
C.2.1	Multi-Block Configuration	194
C.2.2	Dual-Mesh Configuration	196
C.2.3	Parallel Solver Algorithm	197
C.3	Example of Verification of the Parallel Version	198
C.4	Parallel Efficiency	200
C.4.1	Multi-Block Configuration	201
C.4.2	Dual-Mesh Configuration	202
C.5	Remarks	204
Appendix D Parallelisation of the Block Spectral Multi-scale Solver		205
D.1	Approach	206
D.1.1	Partitioned Base Domain	206

D.1.2	Non-Partitioned Base Domain	212
D.2	Parallel Efficiency	213
D.2.1	Case of Partioned Base Domain	213
D.2.2	Case of Non-Partitioned Base Domain	215
D.3	Remarks	216
 Appendix E MPI Collective Commands		218
 Appendix F Configuration of the Distribution of Source Regions		220
F.1	Periodic Mapping	220
F.2	Non-Periodic Mapping	221

Nomenclature

Abbreviations

(I)LES	(Implicit) Large Eddy Simulation
AM	Additive Manufacturing
AUSM	Advection Upstream Splitting Method
BS(M)	Block Spectral (Method)
CD	Central Differences
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy number
CPU	Central Processing Unit
DEM	Discrete Element Method
DM	Dual-Mesh
DNS	Direct Numerical Simulation
DRE	Discrete Roughness Element
DTS	Dual Time Stepping
FDS	Flux-Difference Splitting
FVS	Flux-Vector Splitting
GCI	Grid Convergence Index
HPC	High Performance Computing
HVM	High Value Manufacturing
IMB	Immersed Mesh Block
MEMS	Micro-Electro-Mechanical Systems
MPI	Message Passage Interface
MUSCL	Monotonic Upstream-Centered Scheme for Conservation Laws

PIV	Particle Image Velocimetry
$Rk - \epsilon$	Realizable $k - \epsilon$
RAM	Random Access Memory
RANS	Reynolds-Averaged Navier Stokes
SA	Spalart-Allmaras
SEM	Scanning Electron Microscope
SGS	Sub-Grid Scale
SIMPLEC	Semi-Implicit Method for Pressure Linked Equations-Consistent
SRST	Spatial Resolution Source Term
SST	Shear-Stress-Transport
TST	Total Source Term
UST	Unsteady Source Term
Ma	Mach number
Nu	Nusselt number
Pr	Prandtl number
Re	Reynolds number
St	Stanton number
Tu	Turbulence intensity

Symbols

ℓ	Characteristic length [m]
A	Area [m^2]
A, B, C, D	Fourier series coefficients
B	DRE base dimension [m]
C	Coarse mesh
c	Sound speed [m/s]
c^*	Critical c evaluated when local $Ma = 1$ [m/s]
C_C	Coarse mesh corrected by BS TST
C_d	Drag coefficient

c_e	Computational gain
C_F	Coarse mesh updated by fine mesh
C_f	Friction coefficient
C_{map}	BS computing time coefficient
D	Diameter [m]
D_h	Hydraulic diameter [m]
E	Specific internal energy [J/m^3]
e	DRE eccentricity
e_a	Relative error
F	Fine mesh
F_v	Flux vector
G	Basis function coefficients
H	Domain height [m]
h	Height of DRE [m]
H_0	Specific total enthalpy [J/m^3]
I	Unity matrix
J	Jacobian matrix
K	DRE height [m]
k	Time-step
k_s	Equivalent sand grain height [m]
k_t	Turbulent kinetic energy [m^2/s^2]
K_{cond}	Thermal conductivity [$W/(mK)$]
L	Domain length [m]
M_c	Coarse mesh cell-count of each C_F (C_C) region
M_h, N_h	Number of harmonics
M_S, N_S	Number of source regions
M_{BS}	Number of DREs of partially embedded configuration
M_{full}	Number of DREs of fully embedded configuration
N	Number of DREs

N_F^A	Number of fine mesh faces corresponding to a single coarse mesh face
N_F^V	Number of fine mesh cells corresponding to a single coarse mesh cell
N_m	Number of samples
N_{C_C}	Number of C_C regions
N_{cores}	Number of cores
N_{extra}	Extra C_F regions
P	Static pressure [Pa]
q	Heat flux [W/m^2]
R	Residual vector
r	Radius [m]
R_a	Roughness centerline height [m]
R_μ	Roughness mean height [m]
r_f	Radius at bisection plane [m]
R_{p-v}	Roughness average peak-to-valley height [m]
R_q	Roughness root-mean-square height [m]
R_{sk}	Roughness skewness
T	Temperature [K]
t	Time [s]
u, U	Axial/x/parallel to wall velocity [m/s]
u, v, w	Velocity components [m/s]
u^+	Dimensionless profile velocity
u_τ	Friction velocity [m/s]
U_{mag}	Velocity magnitude [m/s]
V	Volume [m^3]
W	Domain width [m]
y	Wall distance [m]
y^+	Dimensionless profile wall distance

Greek

α, β	Phase angles
δ	Boundary layer thickness [m]
ε	Turbulence dissipation [m^2/s^3]
η	Blasius non-dimensional wall distance
η_p	Weak parallel efficiency
γ	Perfect gas specific heats ratio
κ	Von Karman constant
Λ	Eigenvalues vector
λ	Wavelength (spacing) [m]
ν	Kinematic viscosity [m^2/s]
ν_t	Turbulent kinematic viscosity [m^2/s]
ω	Specific turbulence dissipation [s^{-1}]
Φ	Primary (Conservative) flow property
Φ'	Unsteady component of Φ
Ψ	Basis function
ρ	Density [kg/m^3]
τ	Pseudo-time [s]
θ	Angle [$^\circ$]
θ_t	Boundary layer momentum thickness [m]
$\tilde{\nu}$	SA turbulence field variable [m^2/s]

Subscripts

0	Total flow property
∞	Far-field
b	Bulk
c	Chord
cp	Convective part
d	Dimple

F	DRE frontal area
g	Global
I, J	Phase angles for each phase coordinate
i, j, k	Mesh point indices
in/out	Inlet/outlet
l	Local
p	Pimple
pp	Pressure part
R	Base area of DRE
S	Smooth area surrounding DRE
sp	Spanwise
st	Streamwise
W	DRE wetted area

Chapter 1

Introduction

Advanced manufacturing techniques introduce new design capabilities, as well as new challenges. In this first chapter the background and motivation of the present work, emerging from 3D-printing, are introduced and the new design capabilities and issues are discussed. In the end, the thesis outline is presented.

1.1 Background and Motivation

1.1.1 Additive Manufacturing: a New Era

Undoubtedly, over the recent decades manufacturing engineering has witnessed substantial progress. New techniques, described as High Value Manufacturing (HVM), have aided engineers in creating objects of high detail at small scales. One of these innovative methods is Additive Manufacturing (AM or commonly known as 3D-printing). AM is the manufacturing process of creating three dimensional objects by joining materials usually layer by layer, greatly reducing design restrictions on component complexity. In addition, the deposited material can be chosen depending on the location, creating multi-material objects. Moreover, AM techniques can create objects from micro-meters scaling up to meters with high accuracy,

maintaining robustness. The low number of multiple parts furthermore allows components to be lighter. The need for complex, light-weight components with a wide geometric scale range in modern engineering fields, e.g. turbomachinery or MEMS, in combination with relatively low component production rates set AM as a valuable manufacturing option [1–3].

This new manufacturing technique has already been utilised in the past for commercial products or research purposes. For example, under the supervision of Professor Wu at Monash University, the reproduction of a whole engine of a Dassault Falcon 20 occurred using 3D-printing [4].

CFM LEAP jet engines accommodate 3D-printed fuel injectors designed by General Electric (GE), shown in Figure 1.1a, which are both lighter and more durable than those produced by conventional means [5]. The specific design is implausible to manufacture with conventional methods, as the interior complexity is difficult to cast. Also, the conventional prototype consisted of multiple parts that needed welding and brazing, increasing the manufacturing difficulty [6]. Additionally, GE is utilising AM in producing the low pressure compressor blades of the GE9X jet engine [7].

Recently another achievement in additive manufacturing was registered: Siemens Industrial Turbomachinery Ltd 3D-printed full-scale gas turbine blades, shown in Figure 1.1b, which then were tested under realistic engine conditions [8]. The blades were exposed to temperatures beyond $1500K$ and $13000RPM$, enduring the full-load testing. As quoted in [8]: ‘The turbine blades were installed in a $13MW$ SGT-400-type industrial gas turbine. As shown in Figure 1.2, the printed turbine blades were produced from a high-temperature-resistant, powdered polycrystalline nickel-based superalloy and are able to withstand the high pressures, extreme temperatures, and centrifugal forces that arise during turbine operation. A laser beam is directed at fine layers of metal powder, which are heated and melted. When the laser is

removed, the metal cools. The process is repeated layer by layer until the blade model from the 3D printer is finished. Thanks to additive manufacturing, the team was able to reduce the period of time from the design of a new gas turbine blade to its production from two years to two months'.

Another field that has been significantly benefited by AM is Micro-Electro-Mechanical Systems (MEMS). MEMS can be inherently structurally complex with a wide range of materials [9]. Complex micro-channels are in need of AM to mitigate the rising heat generation of high-performance circuits.

It is also noted that the accuracy of AM is constantly progressing towards finer scales. The Freeform process of Microfabrica is claimed to print engineering grade metals consisting of nickel, palladium, rhodium, and copper with the extreme precision of $20\mu m \pm 2\mu m$ [10]. Vyatskikh et al [11] developed a two-photon lithography-based process capable of manufacturing complex metal geometries at 100nm resolution. Such finely produced structures can be seen in Figures 1.3, 1.4, 1.5, 1.6, 1.7 and 1.8.

1.1.2 Additive Manufacturing and Surface Roughness

It is therefore safe to claim that this particular technology is more than a trend. On the contrary, indications show it will form into a powerful tool in a wide range of fields, pushing technological advances one step further [1, 18]. Hybrid manufacturing, with AM as one of its processes, could allow the creation of large components whilst also focusing on detailed surface patterns. Considering the aforementioned accuracy and complexity even at the smallest scales, AM can be used to deliberately form roughness patterns by design.

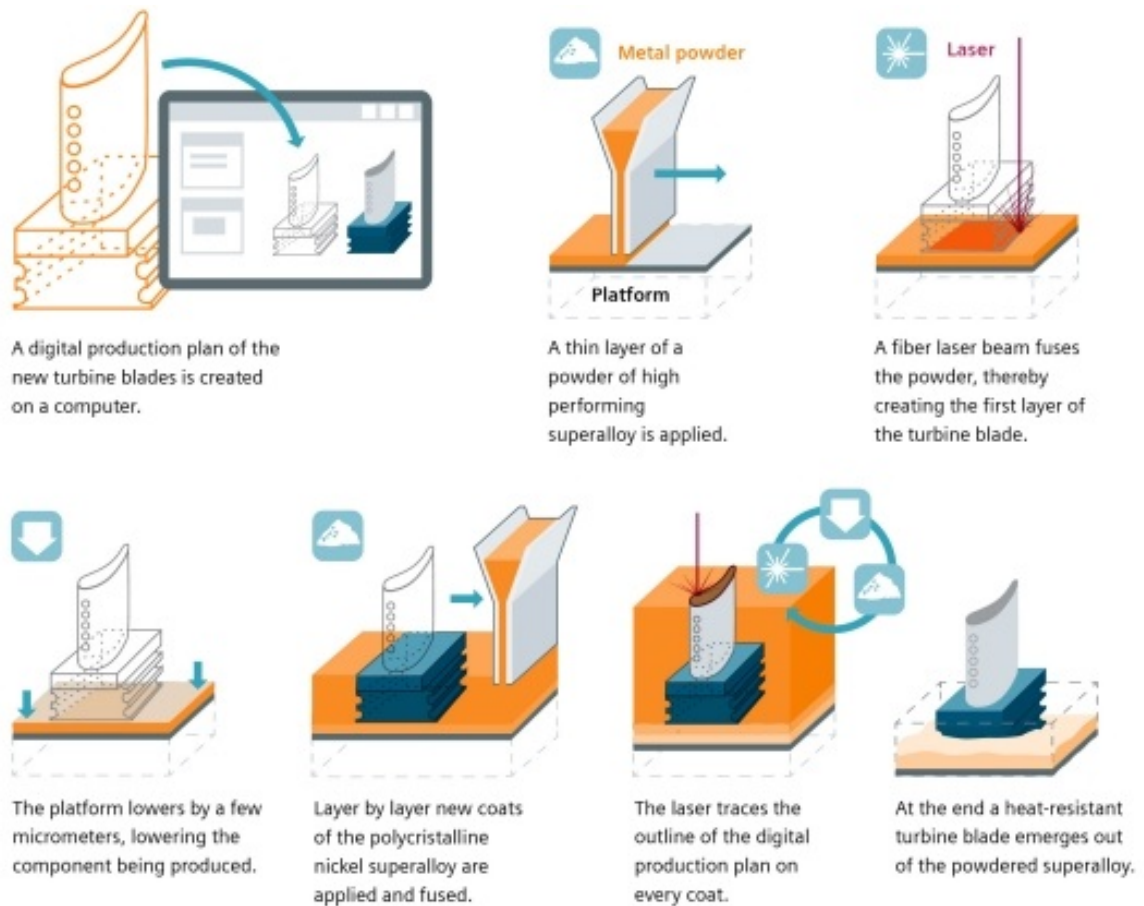
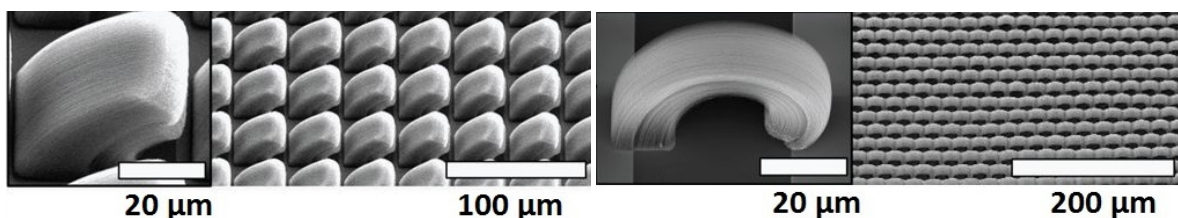
Such roughness can be described as 3D-printed regular roughness, machined roughness, deterministic roughness, manufacturable roughness or discrete roughness. Discrete Roughness Elements (DREs) are also mentioned as micro-structures, micro-features or micro-pillars.



(a) GE fuel nozzle [6]



(b) Siemens 3D-printed tested blades [8]

Figure 1.1 Additively manufactured components**Figure 1.2 3D-printing overview [8]****Figure 1.3 SEM image of Carbon Nano-Tubes micro-pillar growth using TiN as a rate-controlling catalyst [12]**

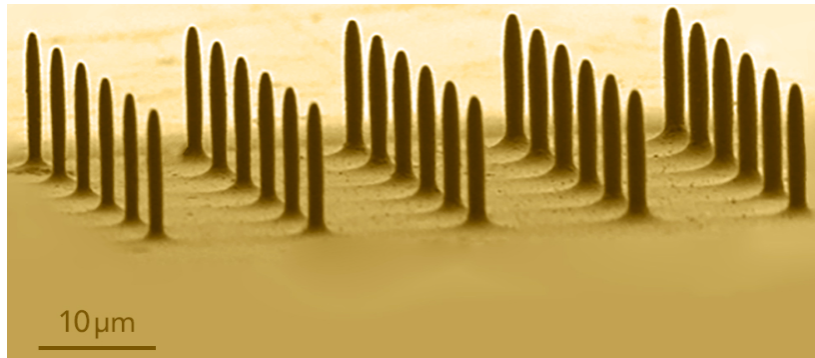


Figure 1.4 Metal AM pillars [13]

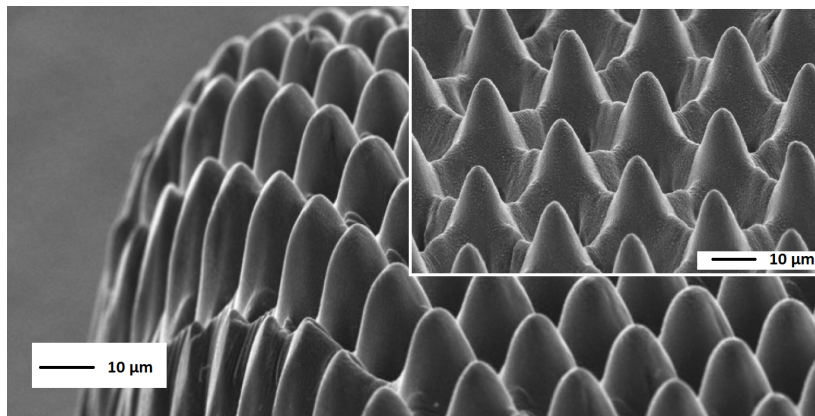


Figure 1.5 SEM image of machined pillars in stainless steel using laser drilling/milling [14]

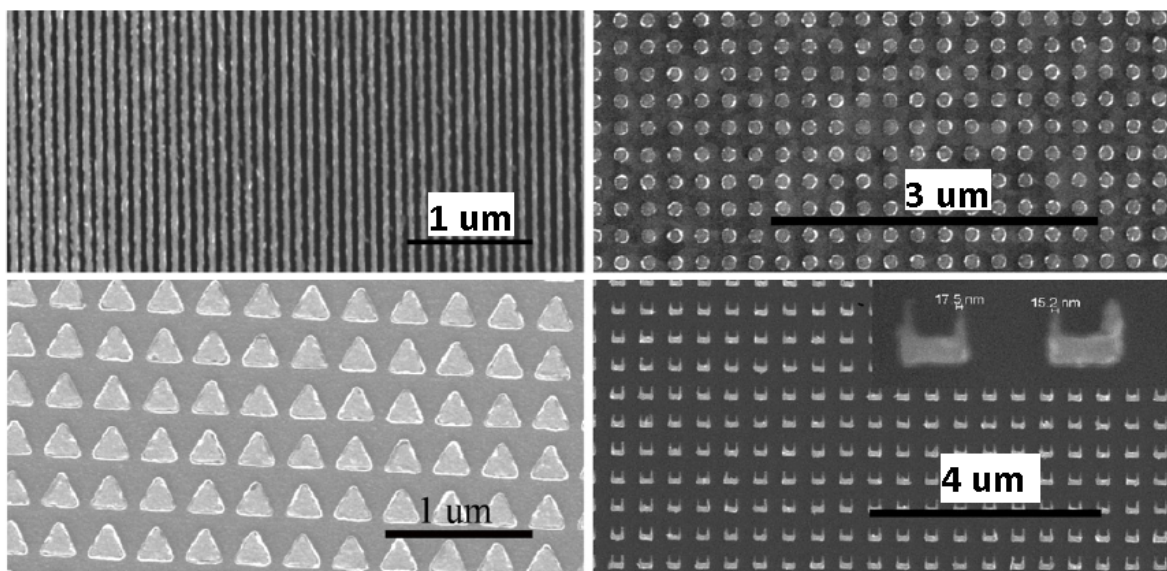


Figure 1.6 SEM images of gold nanostructures fabricated with e-beam lithography [15]

In reality, such discrete roughness elements would demonstrate some random roughness on their surface as well. Of course, that would be orders of magnitude finer than the already significantly fine DREs. Also, it has been shown in the above figures that AM could produce

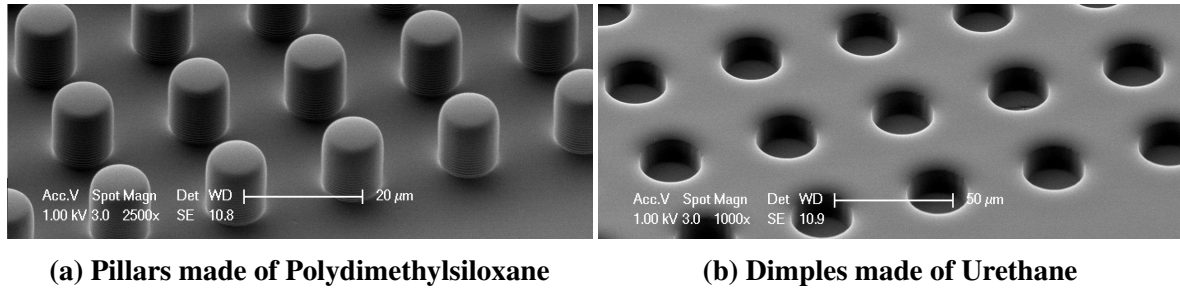


Figure 1.7 SEM images of silicon based micro-structures [16]

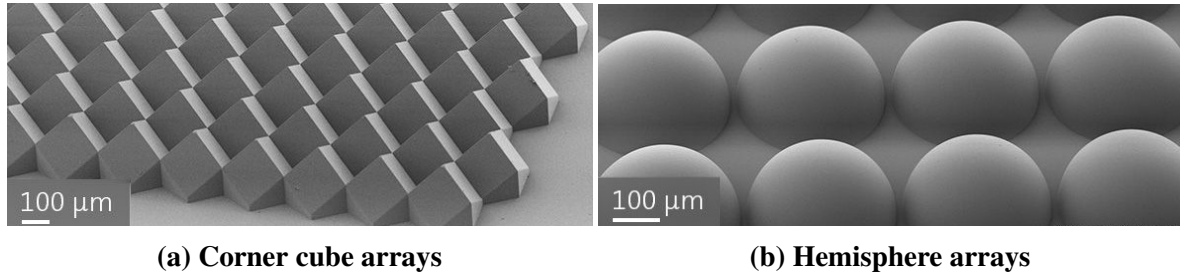


Figure 1.8 SEM images of micro-structures made of Polydimethylsiloxane [17]

highly accurate and well-defined micro-structures. Thus, in this work the random roughness on the surface of the discrete elements is regarded as out of scope.

Hitherto, roughness has been regarded either as a result of engine working cycles due to erosion, corrosion, and other mechanisms that compromise component structural integrity, or as an unavoidable remnant of the manufacturing process. Surface finish can affect both aerodynamics and heat transfer, either individually or in combination with other flow mechanisms, and in some occasions roughness is documented to beneficially alter aerothermal properties, such as promoting higher heat transfer for cooling devices or aiding to reducing drag. These effects can be affected by the flow surroundings, e.g. by adding non-zero freestream pressure distribution or cooling holes, or by changing roughness characteristics. For example, roughness may increase losses due to increased boundary layer momentum loss introduced by the roughness elements protruding through the boundary layer; however, when roughness causes early transition, which mitigates laminar separation, losses are decreased.

Stochastic roughness may enhance the desired component properties, but the lack of control over the local roughness elements may introduce collective negative effects. For

example, highly clustered roughness elements may locally increase the heat transfer, thermally stressing the component, as well as elevate the downstream turbulence to undesired levels. Also, promoting boundary layer transition using stochastic roughness does not allow for well-defined regions of rough areas; extending these rough patches to ensure transition may introduce both avoidable additional losses and increased heat transfer which compromises the component integrity. On the other hand, large scale discrete roughness elements can be shaped, distributed, and scaled to suit the local flow properties, such as the increasing boundary layer thickness, introducing roughness aerothermal effects in a targeted manner. Regular roughness provides the potential of ensuring higher flow control, enhancing desired component characteristics while minimising the negative ones. AM provides the opportunity for component designers to choose the roughness scales, shapes, and patterns, thus shifting roughness from an uncontrollable stochastic feature into an additional intentional design parameter. Examples of plausible manufacturable roughness elements were viewed in Figures 1.3-1.8. Still, manufacturing engineers cannot arbitrarily implement roughness patterns and shapes without the advice of component designers.

The component designer has to face the following question from the manufacturing engineer: "What shape/pattern specifications should the surface roughness have?". Considering typical problems, the component designer would be able to confidently answer by utilising numerical predictions. However, the immense scale disparity between main component and roughness dimensions impose fundamental challenges to conventional CFD implementations. It is often the RANS approach that is adopted for such problems, but RANS turbulence models are known to be inaccurate when predicting separation downstream of a bluff body. Roughness elements, especially those of distinctive shape, are themselves bluff bodies with interacting

wakes. Such configurations would compromise RANS capabilities for an acceptable level of accuracy.

Another issue is the low value of local Re based on the roughness length scale. Let us assume that machined roughness consists of elements whose main dimensions are 1% of those of the component, meaning the roughness characteristic length is smaller by two orders. Also, the local, relevant to the roughness, far-field velocity could be lower as well, due to the existing boundary layer. Thus, if the Re of a blade is of the typically investigated order 10^5 , the Re of the roughness elements would be at the highest 10^3 . A flow in the $Re = 10^2 - 10^4$ regime is prone to a transitional behaviour, and such Re ranges have not been widely investigated.

Engineers, with the assistance of AM, will be able to select the shape of micro-structures and their spatial arrangements. However, as it will be shown in the following chapter, there has not been wide research regarding the aerothermal behaviour of bluff objects in the relevant flow regime. In addition, conventional CFD predictions would encounter significant issues when simulating a large number of roughness elements due to the mesh size and modelling limitations. Therefore, the emerging questions are:

- Does the shape and pattern of manufacturable roughness elements matter?
- Is it essential to resolve such micro-structural configurations in future numerical analysis of component designs?
- If the above questions are positively answered, what can be done to include such fine features in future numerical predictions for aerothermal design applications?

The above analysis argument should be relevant to both internal and external aerodynamics, and in general this work is confined to neither of the two categories. Also, this work is

not confined to a specific application field. AM is already being utilised with remarkable accuracy, allowing the deterministic creation of fine structures, but this HVM method could be hard to apply due to limited numerical prediction capabilities. Any impact of the manufacturable roughness shapes and patterns would imply the need of resolving the flow structures of the relevant size. This would raise the question of how CFD can accurately evolve to accommodate the need of predicting and analysing the aerothermal effects of regular roughness.

1.2 Outline of Thesis

Considering the aforementioned points, the rest of the thesis consists of the following chapters:

- Chapter 2 gives a literature review of previous research exploring the aerothermal effects of roughness and its key parameters. Also, the conventional CFD approaches towards predicting the aerothermal impact of roughness is discussed.
- Chapter 3 presents a numerical investigation, utilising commercial software, exploring the shape and pattern aerothermal effect of machined roughness. The analysis of this chapter answers the first and second research questions and acts as a justification for investigating the third one.
- Chapter 4 describes the theoretical background of the adopted computational approach for tackling test cases with large arrays of DREs. It is partitioned into the dual-mesh approach and the Block Spectral Method (BSM).

- Chapter 5 presents validation test cases for the baseline and dual-mesh solver. Also, there are comparisons between numerical predictions against experimental results of flow over a surface with AM micro-features.
- In chapter 6 the BSM is validated when applied to self-excited flows induced by micro-structures. Also, the scalability of the BS solver when the number of DREs increases is presented.
- Chapter 7 concludes the current work and some future work is proposed.

Also, three appendices are included:

- Appendix A reports the performed validation utilising the commercial solver chosen for Chapter 3.
- Appendix B describes the AUSM numerical scheme implemented for the in-house solver simulations.
- Appendix C describes the parallelisation efforts of the in-house dual-mesh solver which have been performed.
- Appendix D describes the implementation of the BSM in the in-house parallel solver.
- Appendix E briefly presents the employed collective MPI commands.
- Appendix F presents two examples regarding the configuration of the BSM with the Fourier series as a basis function.

Chapter 2

Literature Review

In this chapter the literature regarding the importance and characterisation of conventional roughness as well as conventional numerical approaches for including its effects are presented. Firstly, a few parameters widely used when describing roughness are discussed. Then, the aerothermal effects of roughness are described as documented by a large number of researchers utilising both experimental and numerical results. Some more recent efforts are also reviewed where researchers have also been trying to identify new parameters that have significant impact on component aerothermal behaviour. Following this, conventional methods utilised to model roughness effects in CFD are discussed.

At the end of the literature review, the two research questions as presented in the previous chapter should be well established, suggesting that the shape and pattern of regular roughness need to be taken into account to generate a solution of acceptable accuracy.

2.1 Conventional Roughness Parameters

Conventional roughness is typically stochastic, hence various statistical parameters are utilised for its description. For expressing these parameters, consider the case of a rough

surface with a profile of local height y , over an area A . The first parameter is the roughness mean height, which effectively sets the reference plane for most of the following parameters:

$$R_\mu = \frac{\int y dA}{\int dA} \quad (2.1)$$

A widely common parameter is the roughness centerline mean height:

$$R_a = \frac{\int |y - R_\mu| dA}{\int dA} \quad (2.2)$$

Another utilised stochastic metric is the roughness root mean square deviation:

$$R_q = \sqrt{\frac{\int (y - R_\mu)^2 dA}{\int dA}} \quad (2.3)$$

For describing the prevalence of valleys or peaks the roughness skewness (non-dimensional) is employed:

$$R_{sk} = \frac{\int (y - R_\mu)^3 dA}{R_q^3 \int dA} \quad (2.4)$$

where negative or positive values signify the prevalence of valleys or peaks, respectively.

The depth of the stochastic roughness can be expressed through the roughness peak to valley distance:

$$R_{p-v} = \max(y) - \min(y) \quad (2.5)$$

Finally, a widely used parameter is the equivalent sand-grain height k_s , which is usually a function of the above parameters and was introduced by Schlichting [19]. The purpose of this parameter is to link various shapes of roughness to the well-documented surface roughened with glued sand, i.e. surfaces of the same value should demonstrate the same magnitudes of

C_f . This link is established through a wide range of correlations, as neatly summarised by Bons in [20], and some of them are shown in Table 2.1. Even by this small sample, it is clear that R_a is the parameter often used for roughness characterisation.

Table 2.1 Sample of correlations for calculating k_s as mentioned in [20]

Surface	Correlation	Reference
grooves parallel to the flow	$k_s = \frac{y_{groove}}{5}$	[21]
ordered array of pedestals	$k_s = 16R_a$	[22]
various surfaces	$k_s = 5.2R_a$	[23]
turbine vane surface roughness	$k_s = 4R_a$	[24]

In Figures 2.1 and 2.2 the calculated parameters for variations of stochastic and regular 2D roughness are shown, respectively. All horizontal lines are plotted with respect to the reference line that is determined by R_μ . For the regular roughness, the roughness elements are identically shaped into trapezia, positioned on a horizontal line (smooth surface). The surface roughness can be characterised as dominated by either valleys or peaks, depending on the density (spacing) of the DREs. This is also reflected in the sign of the roughness skewness.

However, there is a potential implication that needs to be addressed. Consider Figure 2.2b where the calculated metrics would have the same value, regardless the positions of the DREs. For example, the three peaks could be closely packed together instead of equidistantly positioned and the statistical values would not change. Thus, the question arises if such alteration would affect the aerothermal properties of the regular surface finish. This would suggest inaccuracy when characterising surfaces based on their statistical roughness parameters, since the pattern of the DREs is not taken into account.

The same question may also rise when altering the shape. Let us consider a 2D rectangular roughness element of height h and base b positioned on a straight line of length L . According

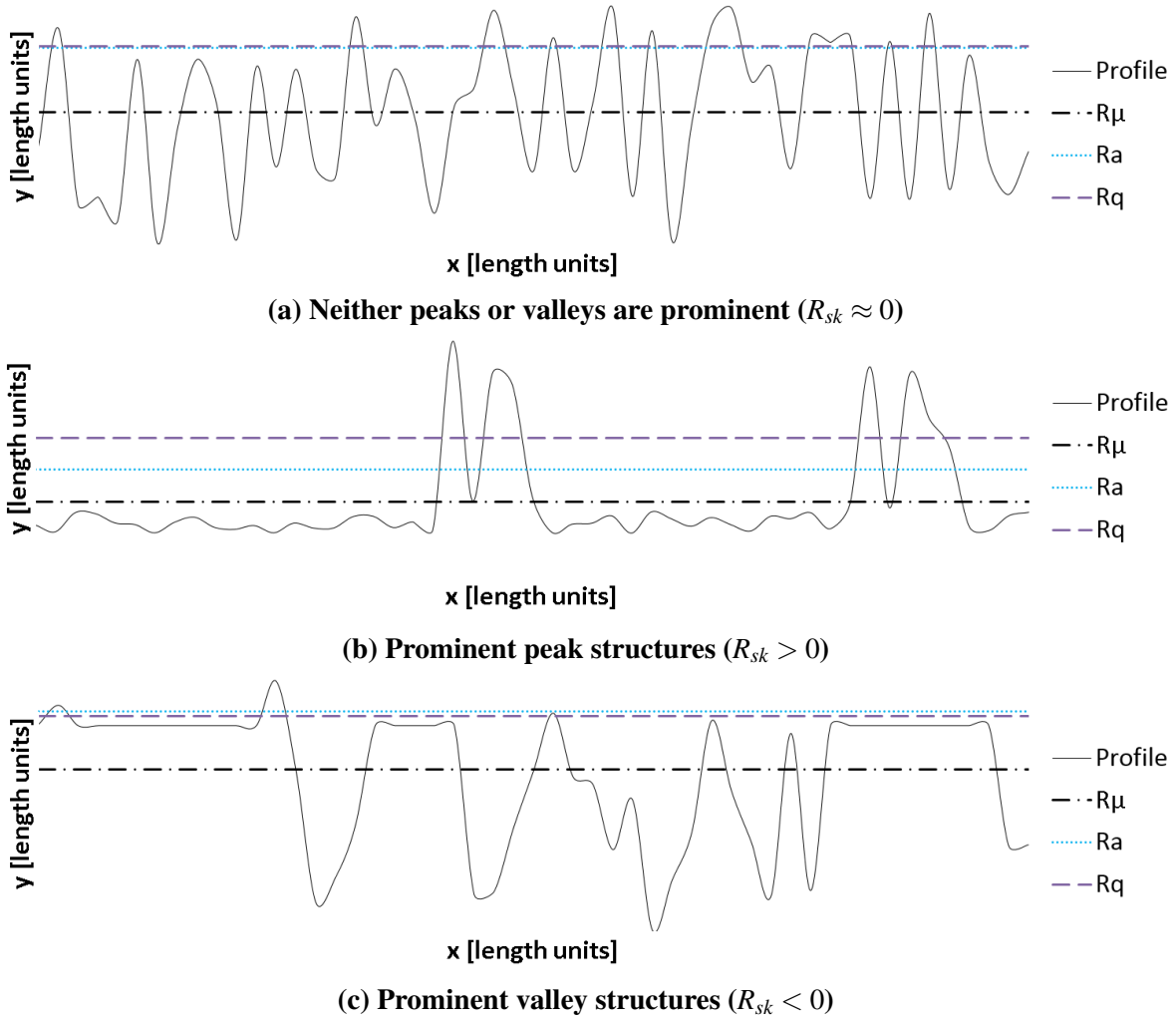


Figure 2.1 Examples of 2D stochastic roughness profiles. Axes reflect arbitrary length units

to the above-mentioned definitions:

$$R_{\mu} = \frac{bh}{L}$$

$$R_a = \frac{\int_0^{L-b} \frac{bh}{L} dx + \int_0^b (h - \frac{bh}{L}) dx}{L} = \frac{2bh}{L} (1 - \frac{b}{L}) \quad (2.6)$$

Solving for h shows that there is a range of b and h values that would generate the same R_a value. This means that similarly to altering the spacing of the DREs, one can alter the defining shape parameters of the micro-structure and maintain the calculated statistical roughness

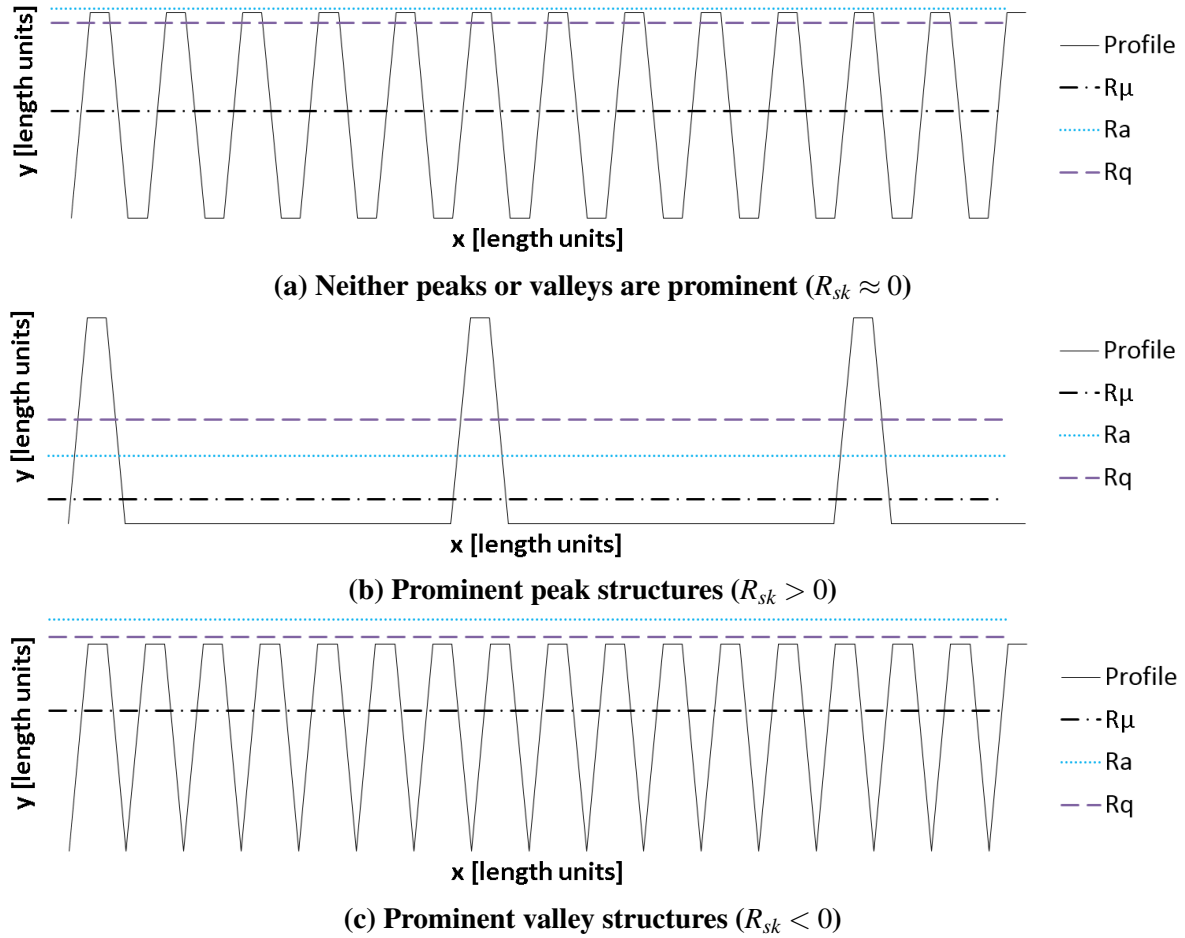


Figure 2.2 Examples of 2D regular roughness profiles. Axes reflect arbitrary length units

parameters at constant value. Naturally, this observation can be extended to having the shape of the DRE completely altered.

The pattern and shape effect of regular roughness are explored in the current and following chapter, but firstly the aerothermal importance of roughness is established to justify the following efforts.

2.2 Aerothermal Effect of Roughness

Over the past century, beginning from the research conducted by Nikuradse [25], followed by Schlichting [19], the substantial aerothermal impact of roughness finish has been widely studied.

For example, Hummel et al. [23] tested the effect of roughness on the performance of a turbine linear cascade. Losses would increase under roughness influence for various Reynolds numbers, reaching a 40% rise at the highest average roughness ($R_a = 11.8\mu m$) when compared to the hydraulically smooth reference blade.

Montomoli et al. [26] studied the separate and combined impact of roughness and wake passing in a high speed wind tunnel. They observed that roughness causes earlier transition at the suction side, delaying flow separation. They suggested a polished blade shows higher losses than an as-cast one.

Bons [27] measured heat transfer and skin friction on real turbine surfaces in a low-speed wind tunnel with no pressure gradient. It was shown realistic surfaces demonstrate fairly different behaviour than the typically used models. Specifically, roughness impact on skin friction was around three times as significant as on heat transfer, while the Stanton (St) numbers predicted by conventional correlations were often not in agreement with measurements. Also, the combination of turbulence and roughness led to a greater impact on skin friction and heat transfer when compared to the individual effects added together.

Bons and McClain [28] experimentally investigated the combined effect of roughness and pressure gradient (positive, zero and negative) on the heat transfer of turbine blades, reaching the conclusion that heat transfer is remarkably affected by these mechanisms combined. St was over-predicted by 10% when adding the heat transfer caused by those two conditions separately. When a favourable pressure gradient was considered, the smooth components with turbulent boundary layer showed higher St values than the roughened blade. Inversely, the opposite was observed when an adverse pressure gradient was applied.

Bons et al. [29] tried to observe the roughness effects through CFD when combined with freestream pressure gradients. Two methods were followed: the Discrete Element Method

(DEM) and a fully resolved three dimensional flow domain. DEM, which will be described later on, was implemented in a 2D finite difference solver with a mixing length turbulence model. The latter case was characterised by an adaptive Cartesian grid implemented using the Spalart-Allmaras (SA) turbulence model. Both computational methods were applied to surfaces that have been examined experimentally for comparison. Agreement between the experimental and computational results of C_f and St was less satisfactory in the cases where surfaces demonstrated larger roughness features. The numerically predicted C_f and St were within $\pm 8\%$ and $\pm 17\%$ of the experimental results, respectively.

Erickson et al. [30] investigated the impact of roughness on heat transfer including the effect of freestream turbulence and Reynolds number using a linear cascade rig with both smooth and realistically rough surfaces. The results showed that roughness did not seem to affect heat transfer in the presence of laminar boundary layers, however it caused early transition, with the transition onset being independent of freestream turbulence. The location of the transition was not influenced significantly by different Reynolds numbers.

Chu et al. [31] tried to increase the critical heat flux of fluids in MEMS cooling systems by using well defined micro-pillars with dimensions of less than $100\mu m$. It was found that machined micro-structures increase the critical heat flux (by about 160%) through augmented capillary forces.

Film cooling may be affected by roughness either positively or negatively depending on the blowing ratio. Cardwell et al. [32] measured the film cooling effectiveness with a rough endwall, measuring a 30% reduction of spanwise averaged effectiveness midway between cooling rows at the highest blowing ratio.

In general, surface finish acts in combination with various flow mechanisms, such as suction side separation, freestream turbulence and pressure gradients, secondary flows, film

cooling, and general unsteadiness, meaning surface finish impact is affected by such flow mechanisms and vice versa, making numerical predictions case-dependent. Thus, surface roughness needs to be examined under the appropriate flow conditions for accurate prediction.

2.3 Studies on Roughness Parameters

Given the substantial roughness aerothermal effects as discussed, researchers have tried to identify the key parameters of roughness that have been lacking in the conventional roughness definitions. Due to the inherent stochastic nature of non-machined roughness, the statistical parameters defined above are the ones mostly under investigation. However, some studies are executed utilising regular roughness. This allows researchers to explore parameters that would be difficult to define for stochastic surface finish. Such parameters are the frontal area of a DRE A_F , the wetted area of the DRE A_W , the surrounding smooth area of a DRE A_S , and of course the height K of the DRE. In Figure 2.3 some of these parameters are represented on a surface of cylindrical micro-structures. Since the interest lies in the discrete roughness aerothermal effect, this section considers mostly work performed on regular roughness.

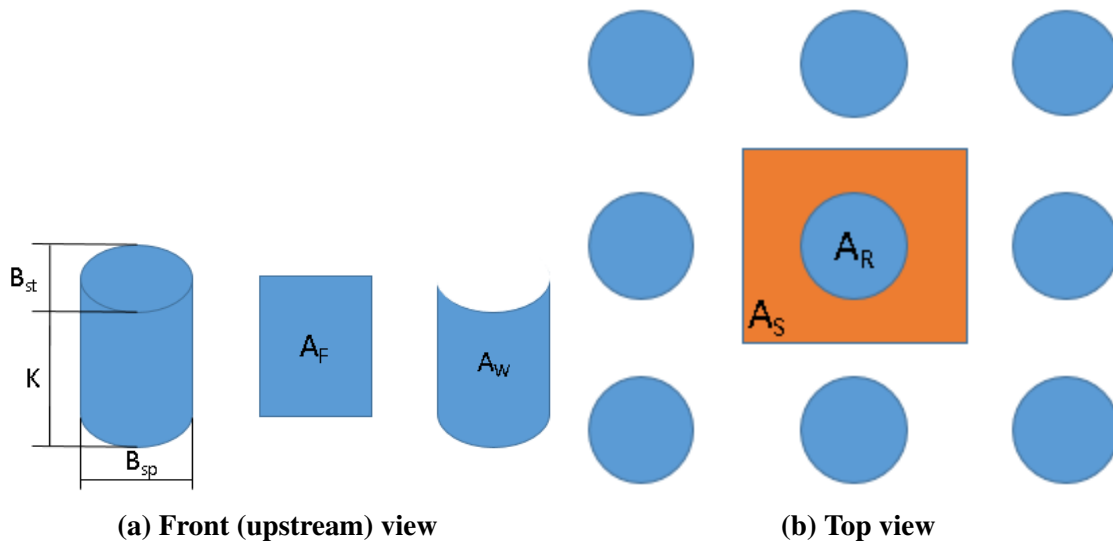


Figure 2.3 Regular roughness parameters

Goodhand et al. [33] questioned if simplified surface criteria such as R_a and k_s are adequate for C_d prediction. They introduced the ratio between centerline mean height and the wavelength of prominent structures (R_a/λ), which characterises the interaction between the largest roughness scales. By experimenting with differently polished blades, a strong relation between this ratio and the onset of rapid drag rise was observed. The drag increase inception, attributed to boundary layer thickening, was taken as a threshold at which the measured drag started to rapidly increase. C_d slope threshold did not appear to have strong relation to R_{sk} . This particular study will be the reference point in the next chapter, where the shape and pattern effects of regular roughness will be more systematically explored.

Chatzikiyriakou et al. [34] performed DNS and LES for a fully developed turbulent channel flow. The walls were fitted with hemispherical roughness elements in various spacing configurations to study the effect on C_f . The simulations were performed using a second order Central Difference (CD) scheme for the convective fluxes and the equations were explicitly time-integrated with a third order Runge-Kutta scheme. The SIMPLEC algorithm was implemented for the pressure-velocity coupling and a Cartesian mesh was created using the immersed surface technique. The results showed that the roughness element height has an impact on friction factor, as the mean velocity profile was increased and shifted towards the viscous wall. The existence of roughness alters the shape of the turbulent kinetic energy profiles, while element spacing affected boundary layer reattachment imposing a weak effect on C_f .

Micro-structures, such as pedestals, in cooling micro-channels is another area worthy of exploration. Thole et al. [35, 36] investigated the pedestal shape effect on turbine cooling passages. The test pieces were either cylindrical or oblong pins in two different orientations. The Re based on the pin diameter varied between $10^4 - 3 \times 10^4$ and three mechanisms were

shown to strongly affect heat transfer: transition on the surface of the oblong pins, wake separation and elevated turbulence downstream of the first row. The prominence of these mechanisms was dependent on shape orientation, Reynolds number and spanwise spacing.

Siw et al. [37] investigated the heat transfer effect of arrays of triangular, circular and semicircular pins of the same cross-sectional area in the regime of $Re_D = 1640 - 4100$. The triangular array showed the highest heat transfer enhancement, i.e. Nu/Nu_{smooth} was on average 7% higher than the second most efficient array, which was the array of circular pins. Also, the triangular arrays showed the highest pressure loss. C_f was 15% higher for the triangular pins than the semicircular pins, which ranked second in pressure losses. Moreover, the triangular set was tested under different spacing configurations, with the densest configuration causing the most prominent effects.

Lorenz et al. [38, 39] explored the effect of the eccentricity (e) of discrete roughness elements on aerothermal behaviour of low pressure turbine blade for various Re and Tu . The roughness elements were initially created on a thin metal foil and then fitted on the blade. The DRE height to blade chord ratio ranged between $2.5 \times 10^{-4} - 11.7 \times 10^{-4}$. It was found that higher eccentricity caused stronger turbulent fluctuations promoting increased heat transfer as well as increased C_f , as seen in Table 2.2.

Table 2.2 Effects on the suction side of the blade, as reported in [38, 39], when $Tu = 1.4\%$ and $Re_c = 150000$. $e > 1$ indicates that largest dimension is perpendicular to the upstream flow

e	Nu_{rough}/Nu_{smooth}	$\theta_{rough}/\theta_{smooth}$
1.00	1.60	2.05
2.80	2.80	2.60
0.38	1.30	1.60

Roberts and Yaras [40] performed an experimental investigation on a stochastically roughened flat plate with streamwise pressure distribution. The variable space consisted of

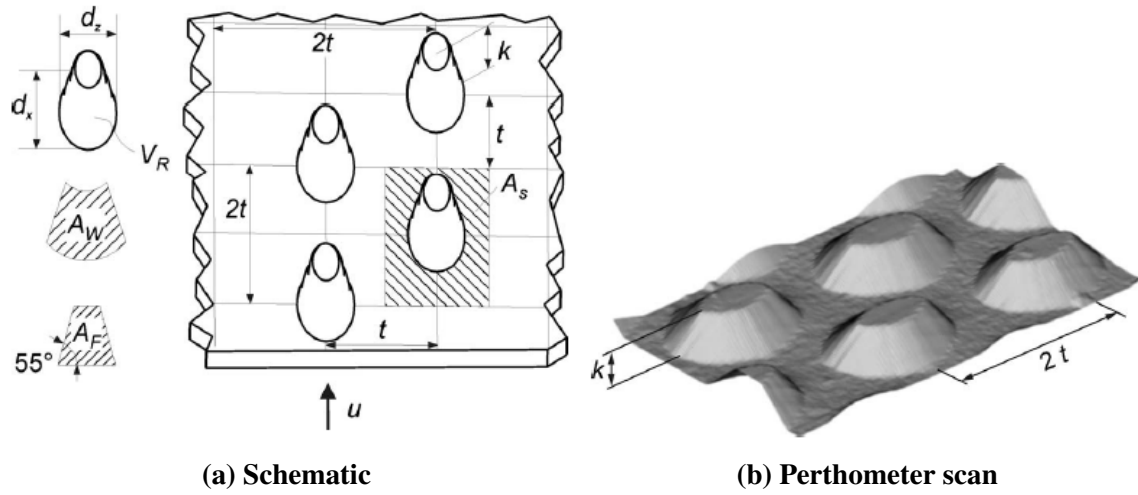


Figure 2.4 Elliptical surface features used in [38, 39]

$R_q/\theta_{t,smooth}$, R_{sk} and the spacing factor, λ/R_q , where θ_t is the boundary layer momentum thickness. Increasing the spacing factor by 8 times caused the distance of separation-transition points to drop from 66mm to 46mm. Increasing the spacing factor 10 times, for a different value of R_{sk} caused the distance to drop from 66mm to 59mm. Likewise, doubling $R_q/\theta_{t,smooth}$, while keeping similar skewness and spacing values, resulted in a reduction from 59mm to 46mm. For similar spacing and height attributes but opposite skewness (0.4 and -0.1) the transition point moved closer to the separation point by less than 5mm. The authors concluded that the primary effect is caused by the roughness height and the secondary by the spacing and skewness.

Stripf et al. [41] explored the turbulent heat transfer and transition inception on a highly loaded turbine vane with truncated conical roughness elements in combination with freestream turbulence. Base and top radius, height ($10\mu m - 80\mu m$) and spacing were varied in the study. Transition on the suction side moved upstream as a result of different size and spacing, with the authors stating that the height had a more prominent effect on transition than spacing, especially at low Reynolds number. k_s proved inadequate for predicting transition inception trends as the earliest inception point did not correspond to the highest k_s . In general, they

observed that the transition location was sensitive even to the finest roughness, affecting heat transfer.

Waigh and Kind [42] gathered a wide range of experimental data of discrete roughness elements (such as cubes, cylinders, cones and hemispheres) to determine the essential parameters for calculating the roughness coefficient used in high Re models. The logarithmic region law of the wall, characterised by:

$$u^+ = \frac{1}{\kappa} \ln(y^+) + C^+ \quad (2.7)$$

where $\kappa \sim 0.41$ is the von Karman constant, and $C^+ \sim 5$ for a smooth wall, is modified in order to introduce a downward shift of the dimensionless velocity profile. For their study they defined a spacing parameter as A_S/A_F , a cell aspect ratio as $\lambda_{st}/\lambda_{sp}$, a bluntness of the DRE parameter as A_W/A_F , a DRE spanwise aspect ratio as K/B_{sp} , and a DRE streamwise aspect ratio as K/B_{st} . They distinguished two DRE spacing regimes by calculating the Λ parameter as:

$$\Lambda = \frac{A_S}{A_F} \frac{K}{B_{st}} \quad (2.8)$$

They concluded that A_W/A_F and K/B_{st} are key parameters while the positioning of primary and staggered elements was of secondary importance. The streamwise element dimension appears only in the roughness density criterion, however it is identified that there were not an adequate number of cases with different streamwise and spanwise aspect ratios.

$$C^+ = \begin{cases} 5 - \left(10.56 \log_{10} \left[\frac{A_S}{A_F} (K/B_{sp})^{0.87} (A_W/A_F)^{0.44} \right] - 7.59 \right), & \text{if } \Lambda < 6 \\ 5 - \left(-5.75 \log_{10} \left[\frac{A_S}{A_F} (K/B_{sp})^{0.55} (A_W/A_F)^{1.38} \right] + 5.78 \right), & \text{if } \Lambda \geq 6 \end{cases} \quad (2.9)$$

Bogard et al. [24] as well as Sigal and Danberg [43] correlated k_s with a function of A_F , A_W , A_S and A_R . They used a similar approach to Waigh and Kind [42], that is in splitting their data into two spacing regimes. By this approach it is suggested that the correlated attributes are substantial for the interaction between roughness and bulk flow. Also, Bogard et al. [24] observed R_a and R_{p-v} are inadequate for heat transfer and skin friction prediction.

Hosni et al. [44] compared the shape effect of truncated conical and hemispherical roughness elements of the same height, spacing and aspect ratio. The DREs were fitted on a flat plate and studied in the regime of $Re_L = 10^5 - 10^7$. Two spacing configurations were examined for $\lambda/D = 2$ and 4. For conical elements the St increased when compared to the smooth case by 55% and 30% for $\lambda/D = 2$ and 4, respectively, while for the hemispherical elements the St increased by 75% and 40% for $\lambda/D = 2$ and 4, respectively, when compared to the smooth case.

Table 2.3 includes information regarding which attributes were found to affect aerothermal behaviour, as well as those surface characterisation variables that did not demonstrate satisfactory prediction. Since researchers have not used the same variables (or ratio of variables) to describe roughness attributes, the grouping of the parameters was based on the judgement of the present author.

Considering the above review, it may be observed that discrete roughness investigations have been designed primarily to imitate stochastic roughness, which is randomly created and relatively uniformly and smoothly shaped when compared to the regular, discrete roughness. Exploration of the aerothermal impact of machined micro-structures is lacking. The shape and spacing of such structures and their effect on the interaction between DREs may turn out to be of high importance, thus making their computational resolution imperative. However,

Table 2.3 Overview of past investigations of micro-structure key parameters (D-Discrete Roughness, S-Stochastic Roughness). ✓ and x denote a parameter that was found to provide consistent or inconsistent, respectively, characterisation of the aerothermal behaviour of roughness

	Shape				Topology			Surface			
	K	B_{st}	B_{sp}	$\frac{A_W}{A_F}$	$\frac{A_S}{A_R}$	$\frac{R_a}{\lambda}$	λ	R_a	R_{p-v}	k_s	R_{sk}
Goodhand (2015) (S)						✓		x			x
Chatzi/kou et al. (2015) (D)	✓						✓				
Thole et al. (2015) (D)		✓	✓				✓				
Siw et al. (2012) (D)				✓			✓				
Lorenz et al. (2012) (D)		✓	✓								
Roberts and Yaras (2006) (S)	✓						✓				✓
Stripf et al. (1998) (D)	✓						✓			x	
Waigh and Kind (1998) (D)	✓	✓		✓			✓				
Bogard et al. (1996) (D)				✓	✓			x	x		
Hosni et al. (1993) (D)	✓	✓	✓	✓			✓				
Sigal and Danberg (1990) (D)				✓	✓						

resolving the wake interaction of regular roughness would in turn add difficulties due to CFD limitations, as discussed in the next section.

2.4 Conventional CFD Modelling of Roughness Effects

The importance of considering roughness effects in numerical predictions can be justified by the roughness aerothermal effects. Such roughness effects have been incorporated in computational modelling and below the most common approaches with variable fidelities are discussed.

Firstly, the least costly way for heat transfer predictions is to apply the Reynolds analogy. When considering gases, the relation is expressed as

$$Nu = Re \frac{C_f}{2} \quad (2.10)$$

and can provide an initial insight for cases of simple geometries without a pressure gradient. However, Reynolds analogy is found to be inadequate in complex flows, especially when pressure gradient is present, despite the qualitative indications it may offer (e.g. [20, 45]).

Secondly, effort has been put into developing turbulence models that account for the effects of roughness on the bulk flow, with significant progress resulting into models that provide fair agreement between experiments and numerical simulations. These models account for the roughness effects by using a k_s function, which usually alters the turbulent viscosity ν_t (e.g. Aupoix and Spalart [46]) or the k_t , ω or ε terms in the $k - \omega/\varepsilon$ turbulence models (e.g. Knopp et al. [47], Eca and Hoekstra [48]). Similarly, for high Re models this can be achieved by altering the wall functions (e.g. Suga et al. [49]). The logarithmic area of the law of the wall is shifted downwards by a value determined by either stochastic or deterministic roughness traits. A simple example was presented in the previous section.

These promising results are mostly accurate in few cases, and when the time comes for implementation into more complex flow domains consistency is often unachievable without recalibrating the models [20]. Such issues are introduced due to the k_s lacking universal consensus when defined, and also due to being associated only to the C_f and not the St [20, 50–52].

Another alternative is the Discrete Element Method (DEM). Here, the roughness impact appears as additional terms in the governing flow equations in the boundary layer. These additional terms account for the blockage by the roughness elements, drag and heat transfer [20, 28]. The total drag is assumed to be the summation of the shear drag from the smooth surface and the form drag of the roughness elements. Similarly, the total heat transfer is the summation of the heat transfer from the smooth surface and the roughness elements. Hence, the governing equations are formulated including blockage factors, while the momentum and

energy equations are augmented within the boundary layer by functions of C_d or C_d and Nu , respectively. The empiricism is introduced by semi-empirical correlations for C_d and Nu . For example, in [50] they are functions of local Re and Pr . Similar approach is followed in various other DEM implementations (e.g. [52]). Despite the positive results, DEM cannot be applied on 3D time-dependent flows since it is still formulated only for steady 2D cases [20].

The next step would be to include the geometric features of the DREs in the computational domain. Utilising RANS, though, would not suffice. Wake shedding from bluff bodies, like the DREs, is a highly unsteady phenomenon, affected by the fine turbulence scales. The oscillatory repositioning of the separation and reattachment locations is difficult to be captured by turbulence modelling [51, 53]. Extensive mesh refinement in the DREs vicinity could mitigate the issue; however, this could introduce double consideration of turbulence, from behalf of both the model and the grid resolution.

To bypass the uncertainties introduced by turbulence modelling, LES or DNS can be deployed. There are case studies that explore the aerothermal impact of DREs (e.g [34, 54–59]), however most of such studies consider a small number of DREs, rarely more than a single DRE, due to the mesh size requirements.

Therefore, there are two contradicting requirements. The first one is the need for high fidelity computational modelling near the wall. The second is the adequacy of low fidelity in the bulk domain, allowing lower grid density. Combining them would require a split into two coupled problems. The former would resolve the fine length and time scales whilst the latter would resolve the coarse domain, constituting a multi-scale solution to the problem.

2.5 Multi-Scale Methods

Multi-scale methods have witnessed significant growth, particularly due to their suitability for fluid mechanical problems that are characterised by flow regions that do not obey continuum medium mechanics. Of course, a multi-scale approach can be adopted for problems that do not demonstrate scale-dependent governing equations, yet are characterised by a wide range of scales. The multi-scale approach allows all significant scales to be resolved accurately and, at the same time, the collective effect of fine scales are introduced to the low fidelity domain and vice versa.

For example, coupled convection-conduction simulations should be performed when high temperature gradients are involved. However, the typically high time-scale discrepancies between solid and fluid domains introduce either significant rounding errors in the solid domain or computationally expensive simulations, depending on which domain determines the physical time-step. He and Fadl [60] attempted to solve this issue by introducing a triple-nested timing formulation. Based on the Dual Time Stepping (DTS) [61], the internal pseudo-time marching consists of two parts: a) the time-marching for the fluid domain, which is performed with a small time-step, and b) the calculation of the solid domain properties in a much larger time-step for the solid part, coupled with the time-averaged fluid part.

Lad and He [62] applied the Immersed Mesh Block (IMB) method to resolve the flow inside and around cooling holes. Typical Nozzle Guide Vanes without cooling holes have low grid density requirements. This contradicts the highly fine mesh needed to resolve the flow in the vicinity of the cooling holes. Thus, they inserted fine mesh blocks that resolved the cooling holes and the near-wall flow structures. In order to couple the coarse and fine mesh

domains, mass conservation at the interface between coarse grid and fine mesh was enforced, i.e. the net mass fluxes at the interface are corrected to be the same for both domains.

He developed the steady multi-scale Block Spectral Method (BSM) aiming to accelerate high fidelity CFD simulations that are characterised by wide spectra of spatial scales [63–66]. In terms of regular roughness, this method aims to provide efficient resolution of the detailed local flow variations in space and time around the large scale micro-structures. This resolution is provided without resorting to modelling every single micro-structure in detail, as a conventional large scale CFD simulation would demand, but still demonstrating similar time-accurate and time-averaged flow properties. The method was implemented in a serial finite volume solver.

The BSM was initially applied mostly on steady or quasi-steady (large scale unsteadiness) cases, mostly of film cooling. Firstly, the method was tested on a simplified steady 2D film cooling case where the hot gas and the cold gas were separated by a thin metal layer [63]. Also, an annulus sector with roughness elements on the inner isothermal surface was considered [63]. A simple effusion cooling configuration was simulated [64], with the inlet stagnation temperature in both the hot and cold sector varying at a low frequency. The simulations showed fair agreement between the direct and the Block Spectral solutions. Regarding discrete micro-structures, He performed a coarse mesh study [65] by applying the method on a surface fitted with dimples.

Recently, the Block Spectral Method was reformulated for self-excited flows [66]. The method was applied to 2D jets (20 in total) distributed equidistantly in the lateral direction with smooth inlet stagnation pressure variation. By refining fewer than half of them a significant agreement of the time-averaged flow between the direct and the BS multi-scale solutions was observed. Similarly, He considered a matrix of 50×3 2D cylindrical pins (50 pins in the

freestream direction). Finally, He performed a multi-scale 3D simulation of film cooling flow. The domain consisted of groups of three holes in the freestream direction, repeated in the lateral direction 15 times. By applying a fine mesh for only 6 arrays of cooling holes good matching of the time-averaged solution between the BS and direct simulations was achieved.

Still, to the author's knowledge, multi-scale modelling applied to resolve the interaction between the fine flow scales introduced by roughness and the bulk flow has not been widely considered.

2.6 Remarks

In this chapter the statistical parameters that are typically employed to describe roughness were discussed, as well as the aerothermal impact of roughness and the underlying key roughness parameters. Next, the conventional CFD modelling approaches were presented and the multi-scale technique was discussed.

Parameters such as R_a , R_{sk} and k_s were presented. These parameters are commonly used in correlations to predict surface finish aerothermal effects. Following this, the implications of their definitions when it comes to regular roughness were briefly explored. These implications provide the first indication that the regular roughness shape and pattern effects need to be studied.

Then, the roughness aerothermal effects were discussed in order to justify the importance of predicting its impact. Surface finish has been found to promote both negative and positive aerothermal component behaviour, depending on the application and the surrounding flow conditions. The effects of roughness and other flow mechanisms were found to combine in a non-linear fashion. This suggests the significance of properly accounting for roughness effects in computational modelling.

Having established the importance of roughness impact on aerothermal characteristics, the next step was to identify the most influential roughness parameters. There have been studies considering either stochastic or regular roughness and, in both cases, the results suggest that the interaction between roughness elements is a key feature. For example, in Goodhand et al. [33] this interaction is performed between the most prominent roughness features. The prominence of the wake interactions is determined by a relative spacing factor, e.g. R_a/λ . This can be extended to the regular roughness, with the capability of considering individual DRE traits as well.

Considering the significant aerothermal impact of roughness, designers have been trying to incorporate its effects in numerical predictions. Correlations, usually based on k_s , alter the defining turbulence properties of the turbulence models near the wall regions. Still, this approach is not adequately accurate. Additionally, applying RANS whilst resolving the geometric properties of roughness within the computational domain is still problematic, as it is widely accepted that RANS fails to accurately predict the wake shedding downstream of a bluff obstacle.

Applying LES/DNS in the whole domain would be immensely computationally expensive. Thus, a multi-scale approach should be sought, as it allows for the high fidelity computational modelling to occur in the area of interest. Still, multi-scale methods have been not widely applied to stochastic or regular roughness.

Considering the above analysis, roughness is an important feature that should be included in computational modelling. The literature indicates that the shape and pattern affect component performance, although at which extent remains unclear. Hence, the next chapter will be focused on answering the first two research questions, as mentioned in the introductory

chapter. If the hypothesis is confirmed, then it will be considered what can be done to accurately introduce roughness shape and pattern effects to the computational modelling.

Chapter 3

Study of Shape and Pattern Effects of Manufacturable Micro-Structures

In this chapter the aerothermal effects of the shape and pattern of regular roughness are investigated. Initially, the motivation for this investigation is outlined based on the existing literature and general information regarding the setup of the cases is presented. The computational case studies, which may be regarded as ‘external flow’ cases, are presented in an attempt to tackle the core questions of this chapter. In the end, the implications of this study are discussed. It is noted that the work of this chapter has been also presented in Kapsis and He [67].

As introduced earlier, there are two distinctive flow regimes of interest, a) a well-established turbulent flow and b) a flow regime subject to a transition due to unsteady disturbances arising from the shear layer instabilities. Firstly, a fully turbulent flow subject to a typical moderate freestream turbulence level of practical interest is examined by using RANS. The first part investigates the drag rise inception when discrete elements are considered, following Goodhand et al. [33]. In the second part the effects of the spacing directionality and the

micro-structure shape are studied. Laminar-to-turbulent transition and how it affects the drag rise inception is investigated in the third subsection utilising a LES solver.

3.1 Preamble

In the introductory chapter, the potential capabilities of AM in generating regular roughness shapes and patterns were discussed. Also, in the second chapter it was shown that fine structures on a surface, i.e. roughness, can play a significant role, either individually or in coexistence with other flow mechanisms. Various researches seem to indicate that conventional rough surface parameters, such as k_s , R_a , R_μ , R_{p-v} and R_{sk} , that are used in correlations may not provide consistent aerothermal characterisations. Instead, the spacing between the elements was shown to be closely relevant when more consistent correlations were sought in the context of stochastic roughness (e.g. Goodhand et al. [33]). The interpretation of this recent finding for stochastic roughness and its implications for manufacturable roughness are of the present interest. In particular, the spacing for the machined roughness can be dependent on the orientation of the pattern, thus the directionality relative to the primary flow now needs to be considered. In addition to the spacing effect, there are indications that the shape of the regular roughness is of a primary concern as well.

There is clearly the need for developing an understanding and predictability of aerothermal behaviour for surface micro-structures. The modelling and methods development for resolving the machined roughness elements are challenging, nevertheless efforts to progress will be motivated if the geometric features of these elements cause meaningful differences. Hence, the purpose of this chapter is to demonstrate, if existent, the collective impact of regular, discrete roughness and its dependence on the pattern and shape of the roughness elements. Such impact should be apparent even when stochastic parameters, as utilised for the conventional

roughness, are matched in order to justify the need for resolving in CFD such fine geometric structures. In turn, this proves the clear motivation for the development of numerical methods that can deal with such multi-scale problems, as discussed in future chapters.

Considering the engineering background and the state of art in understanding and predicting aerothermal behaviour of machined roughness, two main questions arise:

- Do shape and configuration patterns of micro-elements matter in the relevant flow regime?
- How do the roughness elements interact and what is the impact of shape and pattern (and their interplay) on the interaction, in the context of stochastic versus machined roughness?

A particular aspect of interest stems from the previous findings (e.g. [20] and [33]). Roughness is known to cause large increases in flow losses under two distinctive regimes, i.e. a) laminar-to-turbulent transition or b) a fully turbulent flow. Although these findings were only made for stochastic roughness, it is recognised that machined roughness can be regarded as an extreme form of stochastic roughness. Thus, first of all, it would be asked if the analysis method used here can capture these known stochastic roughness behaviours for validation purposes. Furthermore, can the analyses also offer any new insightful interpretations for the behaviours in these two distinctive flow regimes?

The present chapter is motivated by the need to identify and understand the general aerothermal characteristics of micro-elements under flow conditions of interest. At this stage, the main emphasis is on the understanding of the generic aerothermal behaviour without being restricted to any specific type of flow. For instance, ‘external flow’ around a blade surface, or ‘internal flow’ associated with internal cooling configurations have different design objectives in terms of heat transfer versus the flow loss.

3.2 Computational Configuration

Computational Domains

The studied geometries (Table 3.1, also shown in Figure 3.1) consist of in-line roughness elements of nominal spacing $\lambda_{st,sp} = 6r$. In all cases, the rectangular roughness elements have the same streamwise (x direction) and spanwise (z direction) base blockage as the elements of Circ, i.e. the bases are $2r \times 2r$, and thus only the height is altered for matching the statistical geometric properties. Fillets are fitted at the edges ($r_{fillet} = 0.1r$) as this would reflect elements in real environments where sharp edges would be unrealistic. It is noted that cases Circ $_{Ra/\lambda}$ and Circ $_{Ra}$ are considered only in Section 3.4. Also, case Rect $_{kmax}$ is only considered when the directionality of the spacing effects is studied. In the studies of the directionality of the spacing effects the number of micro-structures (N) considered is either 3, 5 or 9.

Table 3.1 Geometries (nominal spacing) considered. Circ is regarded as the baseline geometry

Case Id	Shape	λ_{st}/r	λ_{sp}/r	height	Matched with Circ
Circ	Hemispherical	6	6	$r = 0.5mm$	-
Circ $_{Ra/\lambda}$	Hemispherical	3	6	$r' = 0.626r$	R_a/λ_{st}
Circ $_{Ra}$	Hemispherical	16	6	$r' = 1.254r$	R_a (only)
Rect $_{Ra/\lambda}$	Rectangular	6	6	$r' = 0.668r$	Spacing, R_a and R_a/λ_{st}
Rect $_{kmax}$	Rectangular	6	6	$r' = r$	Spacing and height

In Figure 3.2b the side and top view of the domain are presented. The contribution to the performance parameters from the first roughness element (hump) is omitted to remove the significance of entrance-related flow development effects. In these test cases the physical model is represented by multiple parallel in-line arrays of roughness elements. Therefore,

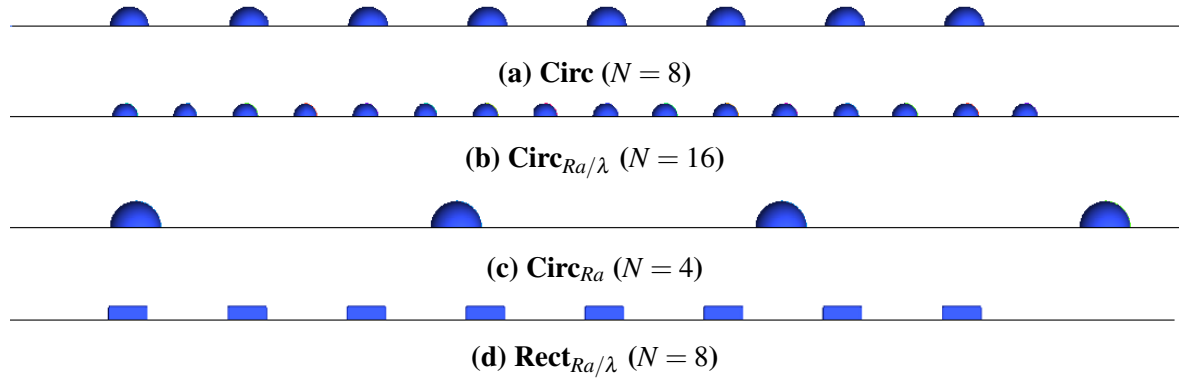
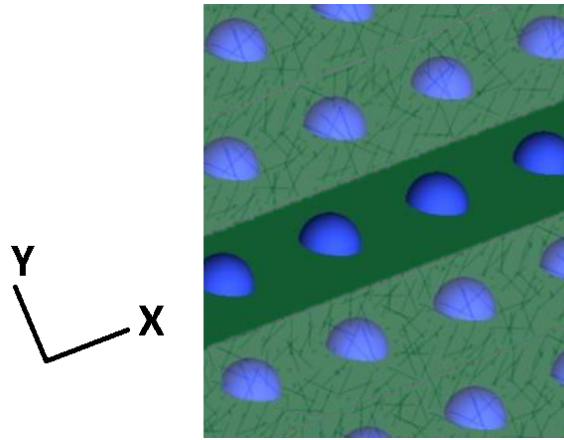
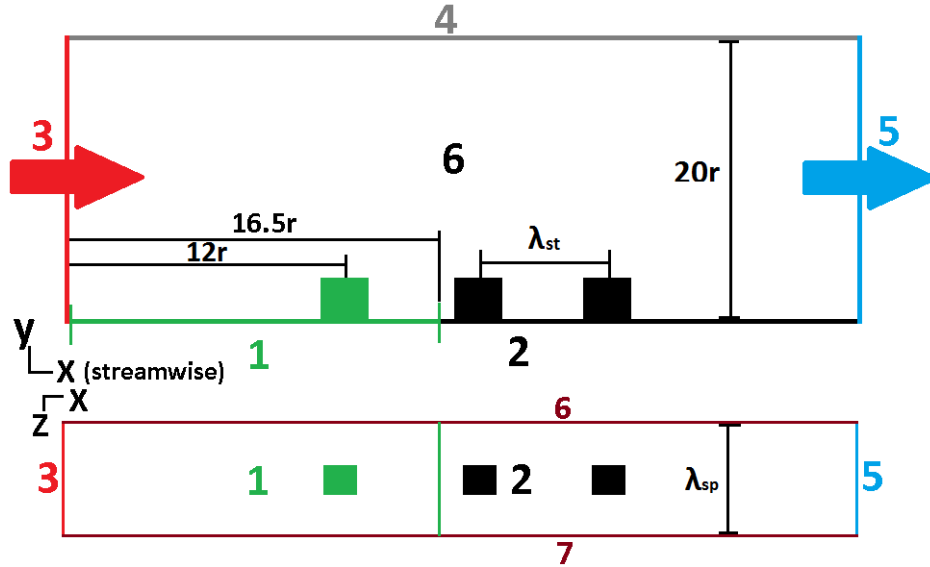


Figure 3.1 Side view of the geometries from Table 3.1



(a) Physical model and computational domain (highlighted strip)



(b) Computational domain and boundaries

Figure 3.2 Physical and computational domain of the considered cases (chosen number of micro-structures is for illustration purposes only)

it is symmetrical with respect to the left/right side. In the physical model, complete spatial flow periodicity is unrealistic; however, this is a common assumption in CFD modelling to reduce computational cost; the key for a meaningful outcome is the consistency between the compared cases.

Considering the notation in Figure 3.2b, the boundary conditions are set as follows: no slip, isothermal for walls 1 and 2, velocity profile, T_0 and Tu at the inlet (3), adiabatic slip conditions for symmetry plane (4), exit (5) static pressure of 1 atm and periodicity for side walls 6 and 7. The time-averaged (steady) turbulent inlet velocity profile, shown in Figure 3.3, is produced utilising the Prandtl-Schlichting formula:

$$u^+ = y^+ \text{ if viscous sublayer, otherwise}$$

$$u^+ = U_b \sqrt{0.2275 \log^{-2.58} \left(\frac{U_b x}{\nu} \right)} \quad (3.1)$$

where $x = 0.0509m$, meaning the following ratio of Re for the same reference velocity U_b reduces to:

$$\frac{Re_r}{Re_x} = \frac{r}{x} \approx 0.01 \quad (3.2)$$

The flow ($Pr \approx 0.73$) is considered compressible ($Ma_{inlet} \approx 0.34$ at $Re_r = 4200$) with constant thermal conductivity and specific heat due to the small temperature differences ($T_{wall} - T_{0,inlet} = 20K$). The kinematic viscosity is computed based on Sutherland's law [68].

Ansys ICEM 14.5 and FLUENT 14.5 are used for mesh generation and flow simulation, respectively. In Appendix A a validation study of the commercial software is presented, considering pins in the relative Re regime. The second order Advection Upstream Splitting Method (AUSM) numerical scheme is utilised.

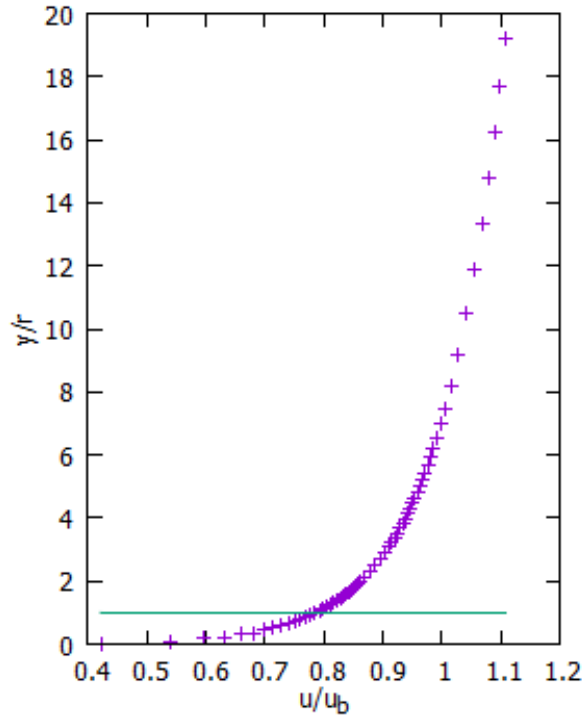


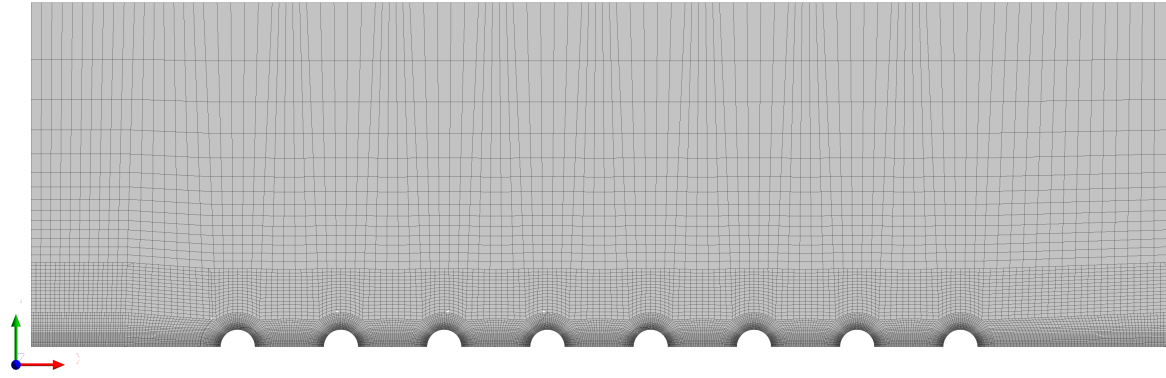
Figure 3.3 Example of inlet velocity profile at $Re_r = 4200$. Horizontal line (green) denotes the height of the baseline micro-structure (Circ)

For the mesh generation the non-conformal multi-block approach is selected as it allows for consistent refinement close to the wall and for creation of symmetric meshes, as shown in Figure 3.4. O-grids are fitted around the roughness elements and mesh refinements during mesh dependency studies are carried out with emphasis on the near-wall regions.

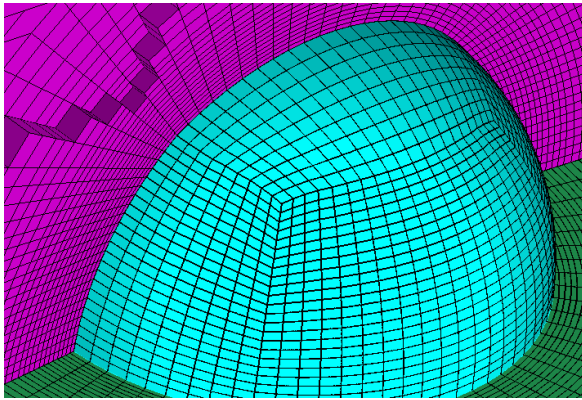
Turbulence Modelling

To examine the behaviours for the two distinctive flow regimes as stated above, two kinds of flow conditions are configured with two turbulence modelling (RANS and LES) approaches applied. For a fully turbulent flow subject to a moderate level of inflow turbulence ($Tu = 4\%$), RANS models are used and three turbulence models are tested: the SA (strain/vorticity based) one equation model, the realizable $k - \epsilon$ (with thermal and pressure effects) and the $k - \omega$ SST (with Low- Re and curvature corrections) two equations models.

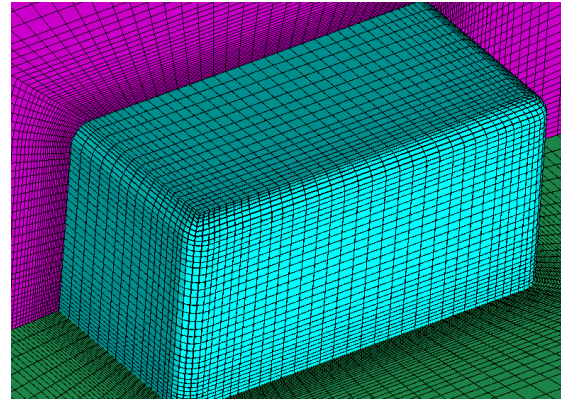
For investigating the transitional flow behaviour, RANS models are not suitable, as the physical flow unsteadiness is caused by self-excited instabilities in the shear layers.



(a) Example of mesh at midspan



(b) Hemispherical roughness element (Circ)

(c) Rectangular roughness element ($\text{Rect}_{k_{max}}$)**Figure 3.4 Employed mesh for RANS solutions**

Thus, a higher flow fidelity with fewer modelling uncertainties would be preferred. For the corresponding flow condition with a clean inflow ($Tu = 0\%$), an implicit LES (ILES) method is adopted. This means no sub-grid scale (SGS) turbulence model is used, avoiding the uncertainty introduced by the SGS model for a sensitive transitional flow. Thus, when the ILES results are mesh independent (as discussed in the relevant section), they could approach those of DNS.

The ILES are performed with a second order integration in the real-time regime. For the laminar-to-turbulent flow regime study (Section 3.6), both steady laminar and implicit LES (ILES), when flow unsteadiness is observed, are employed. ILES is utilised with Dual Time Stepping (DTS), and is applied for $Re_r \geq 838$ and $Re_r \geq 1022$ for the cases Circ and $\text{Rect}_{Ra/\lambda}$, respectively.

Metrics

For monitoring the aerothermal behaviour of the cases the following metrics are used, where the chosen characteristic length is the radius r of the baseline micro-structure shape:

- For the drag coefficient C_d a constant reference area is used.
- For the heat transfer coefficient

$$Nu_r = \frac{qr}{(T_{wall} - T_{inlet})K_{cond}} \quad (3.3)$$

where K_{cond} is the thermal conductivity of the fluid. Due to the low Ma the recovery factor is considered minimal, hence this Nu definition is chosen.

- $C_{d,ratio} = C_d/C_{d,smooth}$ and $Nu_{r,ratio} = Nu_r/Nu_{r,smooth}$, where smooth indicates the parameter for the equivalent case with no roughness elements.
- $\Delta C_{d,ratio} = C_{d,ratio} - C_{d,ratio,ref}$.

For the unsteady simulations, the time-averaged values of the metrics are reported. The Reynolds number is defined based on the inlet bulk velocity as:

$$Re_r = \frac{U_b r}{\nu} \quad (3.4)$$

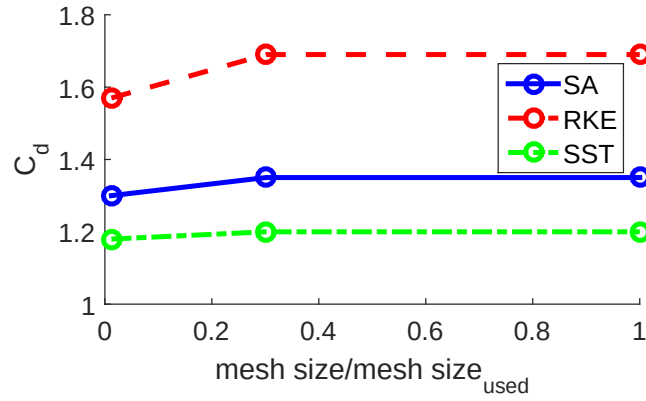
3.3 Mesh/Turbulence Modelling Sensitivity

This sensitivity study is performed for the highest investigated Re for each studied flow regime, i.e. $Re_r = 4200$ and 2700 for the RANS and ILES, respectively.

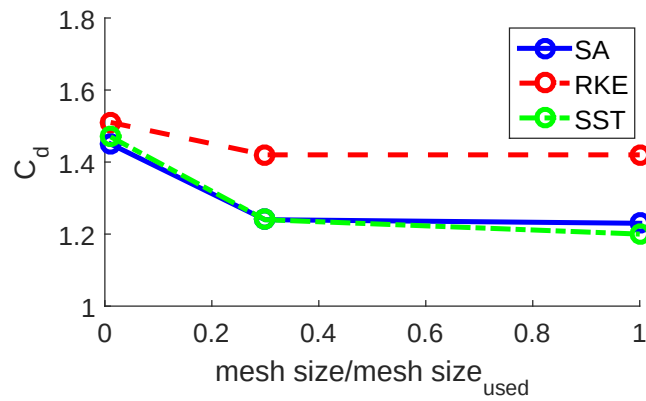
Convergence is assumed when the normalised residuals of the conservative variables reach a plateau (five orders of reduction). For the transient simulations, where DTS is employed,

the same criterion is applied, aiming at observing a flat residual area at each time-step and at least a drop of three orders of magnitude in the normalised residuals.

The mesh and turbulence modelling study results for the RANS simulations are shown in Figure 3.5. These results show that both the $k - \omega$ and SA models are quite consistent (maximum relative error was 2.5% and 12.5% for C_d and Nu_r , respectively) when it comes to magnitude, whilst the $Rk - \varepsilon$ seems to be constantly inconsistent (maximum relative error was 18.3% and 40.8% for C_d and Nu_r , respectively, when compared to the SST $k - \omega$). Thus, the SST $k - \omega$ turbulence model is used for all RANS simulations. Higher fidelity turbulence models are not considered as they are more mesh sensitive.



(a) Hemispherical roughness elements (Circ)



(b) Rectangular roughness elements (Rect_{kmax})

Figure 3.5 Mesh dependency study for $\lambda_{sp} = \lambda_{st} = 6r$. Reference (used) mesh size is $2M$

The consistency of the ILES results is ascertained through a mesh dependency study. The normal mesh count used was ~ 7 million computational mesh cells. The drag variations

computed using different mesh counts are compared in Figure 3.6 for the two shapes of the elements at $Re_r = 2700$ when the turbulent state is well-established. The relative error between the densest and the nominal mesh configurations was 0.3% and 1.8% for the cases Circ and $\text{Rect}_{Ra/\lambda}$, respectively. Although the mesh used is relatively coarse and the results are not completely grid-independent, they are regarded as sufficiently grid-converged for the present study.

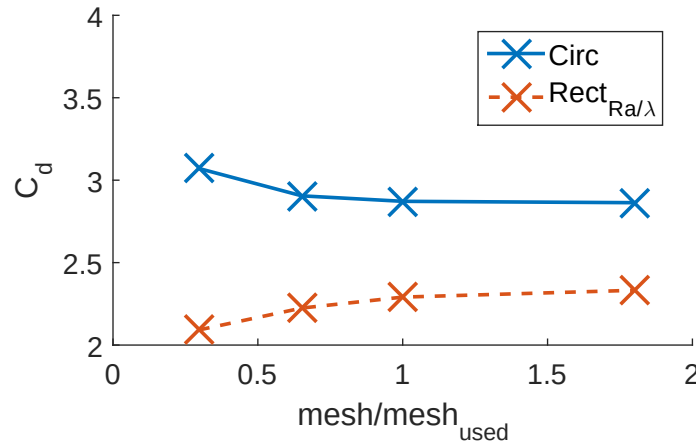


Figure 3.6 Mesh dependency study for Circ and $\text{Rect}_{Ra/\lambda}$ at $Re_r = 2700$ for ILES. Reference (used) mesh size $7M$

For the ILES the nominal time-step is $\Delta t = 1 \times 10^{-6} s$ (for $Re_r = 2700$), which resulted in a sampling rate of 40 samples per shedding period. As derived from Table 3.2, the relative error of the C_d when increasing the nominal physical time-step by 30% was 0.7% and 0.4% for the cases Circ and $\text{Rect}_{Ra/\lambda}$, respectively.

Table 3.2 ILES: Time-averaged C_d values for different time-steps using the nominal mesh

	Δt nominal	$1.3\Delta t$ nominal	$1.7\Delta t$ nominal
Circ	2.88	2.90	2.90
$\text{Rect}_{Ra/\lambda}$	2.29	2.30	2.30

A further indication of a well-established turbulent flow at $Re_r = 2700$ is given by the axial velocity spectrum at a point near the domain exit as shown in Figure 3.7, which shows broadband turbulence disturbances, also qualitatively in agreement with the standard $-5/3$ distribution.

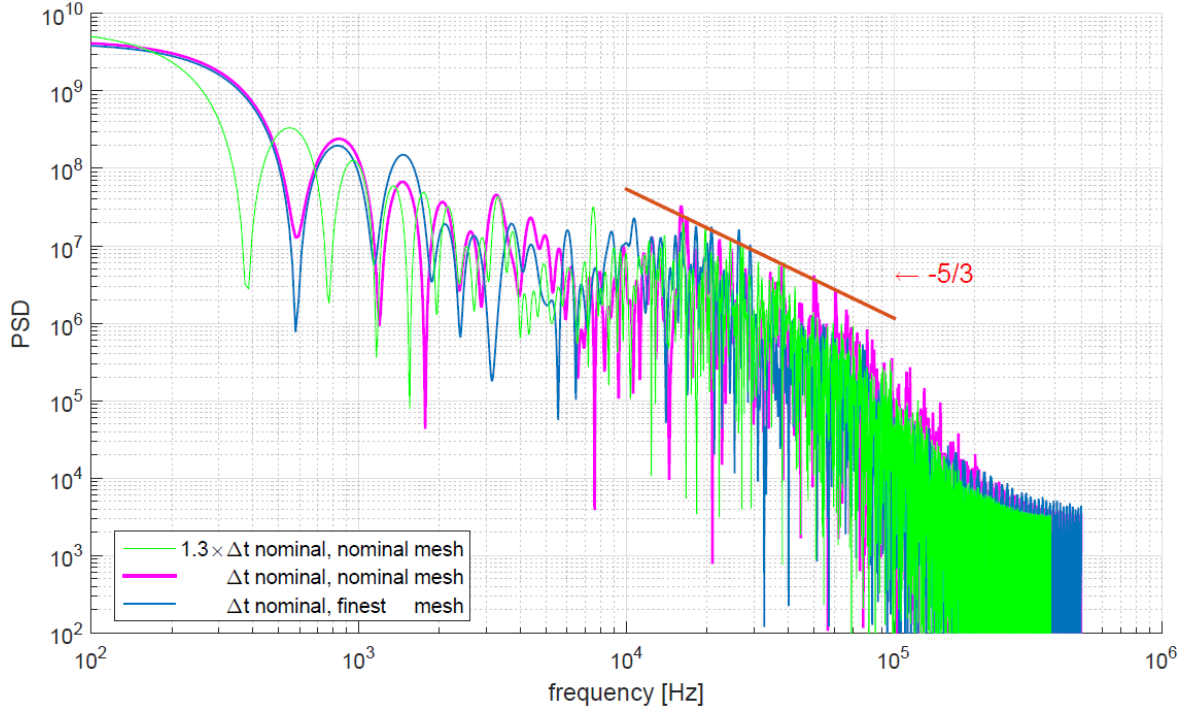


Figure 3.7 Power spectra density of axial velocity downstream¹ of the last element of Circ at $Re_r = 2700$ (ILES)

To quantify the numerical error of the computational analysis the Grid Convergence Method (GCI), which is based on the Richardson extrapolation² [69], is employed. From Table 3.3 it is deduced that the numerical uncertainty introduced by the mesh for the C_d is 1.75% and 1.67% for the SST $k - \omega$ simulations and ILES, respectively.

¹Fast Fourier Transform is applied to the axial velocity. The figure is produced by the dot product between the resulted vector and its conjugate

²This method was introduced to assist with the acceleration of the arithmetic solution of partial differential equations, by choosing appropriate steps, introducing numerical errors within the desired tolerance

Table 3.3 Post-processed C_d mesh dependency data as shown in Figures 3.5 and 3.6. The nominal mesh is used as the finest mesh for these calculations

	SST $k - \omega$		ILES	
	Circ	Rect	Circ	Rect
GCI [%]	1.75	0.00	0.21	1.67
e_a [%]	0.68	0.00	1.16	2.90

3.4 Role of Non-Dimensional Spacing R_a/λ

Goodhand et al. [33] questioned if simplified surface criteria such as R_a and k_s are adequate for C_d prediction of stochastic roughness. Through their experiments on differently polished blades, a strong relation between the ratio of the centerline mean height to the dimensionless spacing (R_a/λ) and the inception of rapid drag increase were observed. Their data reduction along with supporting experimental evidence show this relative spacing parameter to correlate strongly with a critical Reynolds number beyond which the drag starts significantly increasing. The results seem to indicate the role of the relative spacing for both a transitional flow regime and a fully turbulent one.

As an extended validation study, it is firstly explored if the impact of the relative spacing as proposed by Goodhand [33] can be predicted. A fully turbulent (RANS) flow subject to $Tu = 4\%$ at the inlet is considered. To be more consistent with a typical ‘averaged’ shape of stochastic roughness for which the relevant research has been carried out so far, hemispherical elements are utilised in this case. For stochastic roughness, λ is the wavelength of the most prominent roughness elements. For regular roughness, this wavelength expresses the spacing of the DREs.

Given that the primary interest of the present work is to examine the performance characteristics of large arrays of micro-structures, effort should be put into minimising the inlet velocity profile and entrance effects. The drag contributions from each of the

roughness elements are firstly plotted in Figure 3.8 for the two different streamwise spacings. The trends are consistent regardless of the flow regime. It is clear that the contribution of the roughness elements is largely settled after the first one or two most upstream elements (depending on the spacing). Hence, the performance parameters for comparison are taken as such to ignore those from the first one or two elements.

For all cases λ_{sp} is kept the same and its effects will be examined later. A key flow loss characteristic is how the drag ratio varies with the Re . The inception of $C_{d,ratio}$ rise is considered to be at the condition when the slope of the drag ratio with respect to the Reynolds number would significantly increase.

The inadequacy of simply using R_a is illustrated in Figure 3.9. Clearly, the two cases with a matched R_a show very different Re values (at ‘A’ and ‘B’) for the onset of a rapid rise of the drag coefficient. This inadequacy of using R_a only is contrasted with the two closely correlated cases when R_a/λ_{st} is matched. The drag of both cases with the matched R_a/λ_{st} now start to strongly increase at $Re_r \approx 2200$ (at ‘A’), which is in line with the observation made by Goodhand et al. [33]. The change is quite apparent, as at this critical Reynolds number the changes in the curvatures of $C_{d,ratio} - Re$ curves are sharp, creating a roll-up (‘knee’). This behaviour is not shown for those cases where only the R_a is matched; on the contrary, the curvature seems to be largely constant and the horizontal area is absent, with the drag ratio strongly increasing at a much lower Re (at ‘B’).

To help identify and understand the underlying flow physics, the flow patterns before and after the critical Reynolds number are examined, as shown in Figure 3.10. At a low Reynolds number ($Re_r = 1000$), a large and open separation is seen, which is effectively ended by the impingement of the separated shear layer on the upper area of the downstream roughness

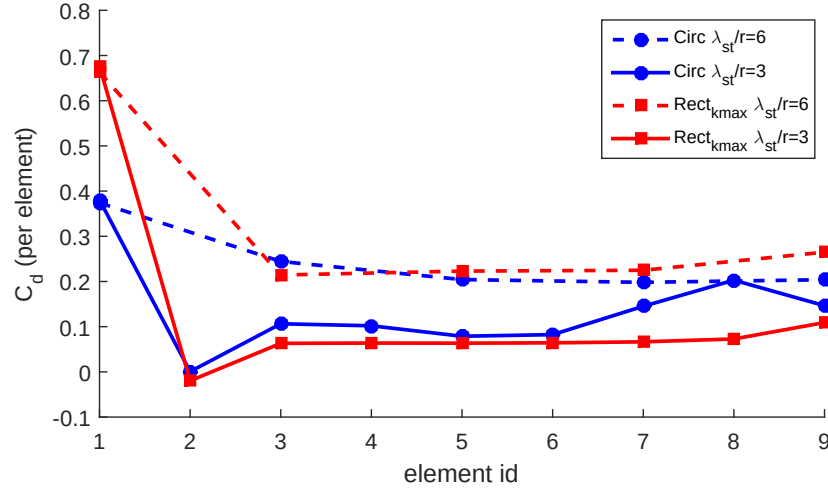
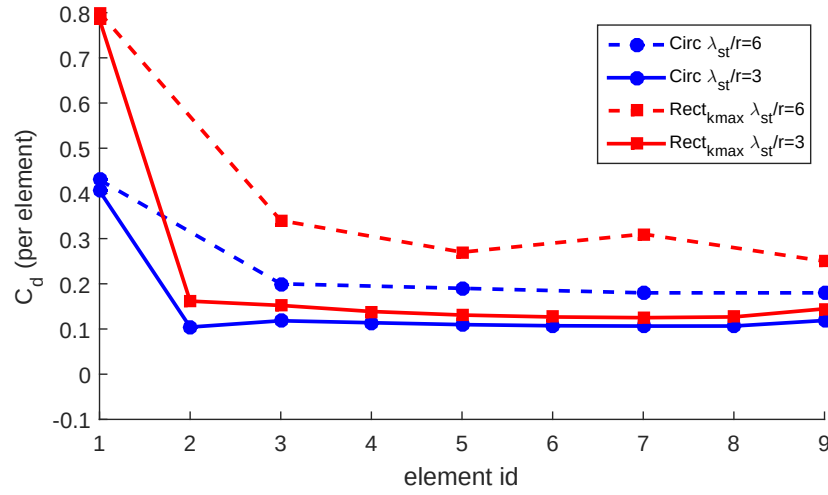
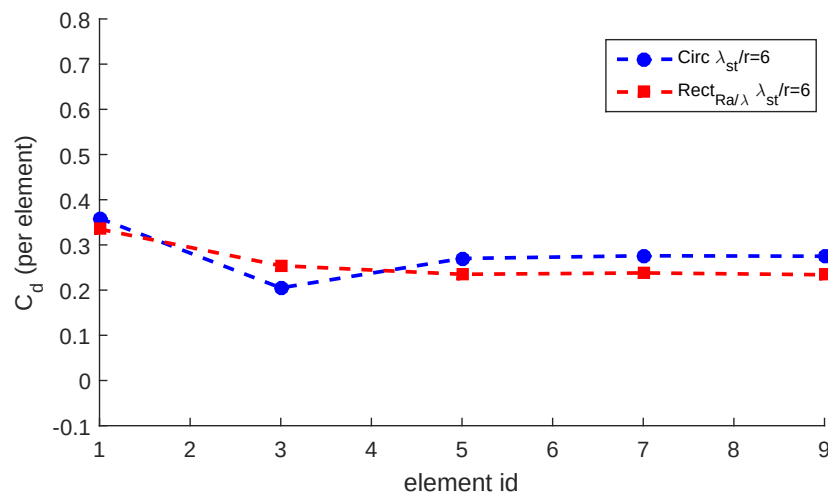
(a) $Re_r = 4200$ ($k - \omega$ SST)(b) $Re_r = 210$ (laminar)(c) $Re_r = 2700$ (ILES)

Figure 3.8 Drag contribution of each roughness element. Each marker corresponds to a roughness element

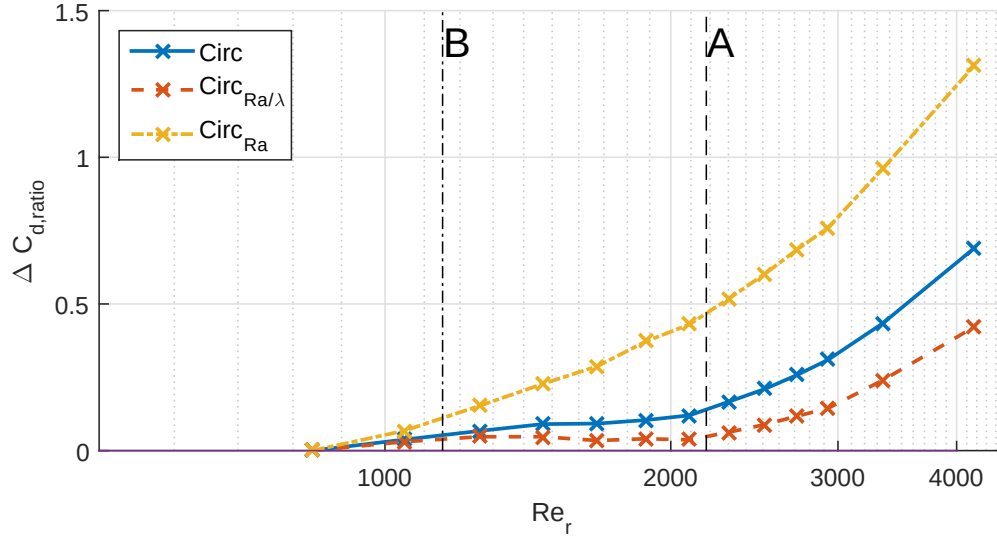


Figure 3.9 Drag increase with Reynolds number ($k - \omega$ SST)

element (Figure 3.10a). This in effect has a considerable shielding impact on the downstream element, thus a low drag should be expected.

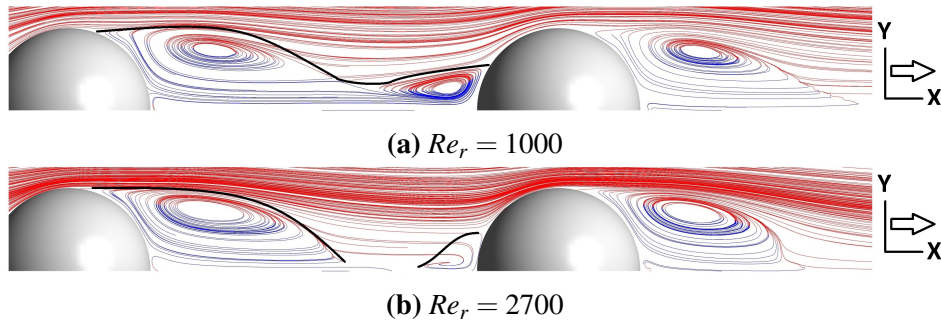


Figure 3.10 Flow patterns at midspan of Circ around the last two elements downstream ($k - \omega$ SST)

A high Reynolds number, above the critical value (e.g. $Re_r = 2700$ as shown in Figure 3.10b), leads to increased turbulence mixing, and subsequently a reattachment, creating a shorter separation bubble. A new boundary layer is thus formed after the reattachment. This kind of flow pattern is expected to be associated with a higher overall drag due to both a higher friction drag of the newly formed boundary layer and a largely inviscid flow behaviour of the frontal region of the downstream element (higher local pressure due to the stagnation, thus higher form drag). The above comparison reveals the distinctive flow pattern change,

attributed to being responsible for the distinctive transition from a low loss regime to a high loss one.

3.5 Effect of Shape and Spacing

3.5.1 Shape Effect for Matched Relative Spacing R_a/λ

Given the key role played by the relative spacing in correlating the increase in the drag with the Re , as shown in the previous section for the hemispherical elements, the shape effect on this behaviour is examined next. To this end, the results of the two shapes are compared, hemispherical and rectangular elements, with matched R_a/λ_{st} . The results are shown in Figure 3.11. The change of the slope is clear for the hemispherical shape of elements, but it is not as obvious for $Rect_{Ra/\lambda}$. There appears to be a much subtler roll up for the rectangular case. Thus, for a matched relative spacing, the shape of the micro-structure does seem to influence the loss characteristics.

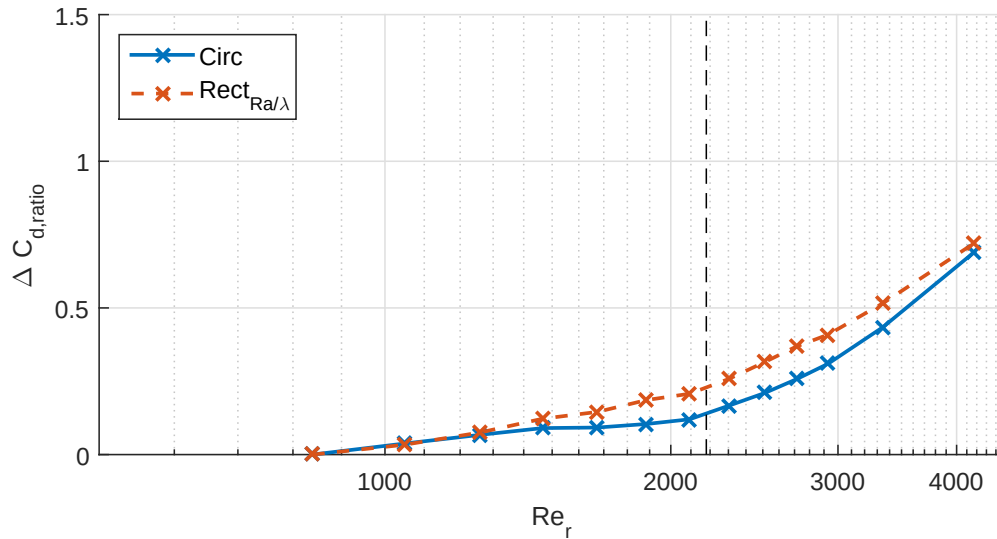


Figure 3.11 Drag increase with Reynolds number for two shapes of DREs ($k - \omega$ SST)

3.5.2 Directionality of Spacing Effect

Having observed the shape dependence of the flow drag parameters, the effect of the directionality of the element pattern relative to the primary flow is explored. For this, two independent spacing alterations are examined in the spanwise and streamwise directions. In addition to the drag forces, the characteristics of the heat transfer are also examined. Because in this investigation the projected area is altered (as spanwise spacing varies), to ensure a fair comparison, the metrics are considered only for a section of wall 2 (as illustrated in Figure 3.2b). This section consists of the roughness elements and the stripe of flat surface between them in the axial direction.

Impact of spanwise spacing:

Firstly, the effect of spanwise spacing is investigated and the results are presented in Figure 3.12, where the trends are shown to be fairly consistent. Drag and heat transfer are largely constant throughout the whole spanwise spacing range. Only for the shortest spacing a subtle rise is shown for Circ and Rect_{k_{max}}, as for these configurations blockage is more severe due to the element height. Overall, the trends for both drag and heat transfer show a clear insensitivity to spanwise spacing alterations. The observed insensitivity to spanwise spacing would also serve as an indication that the element-to-element interactions are dominated by the component of the streamwise interaction in steady solutions (as it will be discussed below), thus mitigating the impact of the assumed periodicity at the spanwise boundaries.

Impact of streamwise spacing:

Regarding the impact of streamwise spacing, wall 2 is kept constant, therefore the number of elements varies depending on the spacing as 9, 5 and 3 number of elements for $\lambda_{st}/r = 3, 6$ and 12, respectively. From Figure 3.13 it is observed that drag follows a non-monotonic trend,

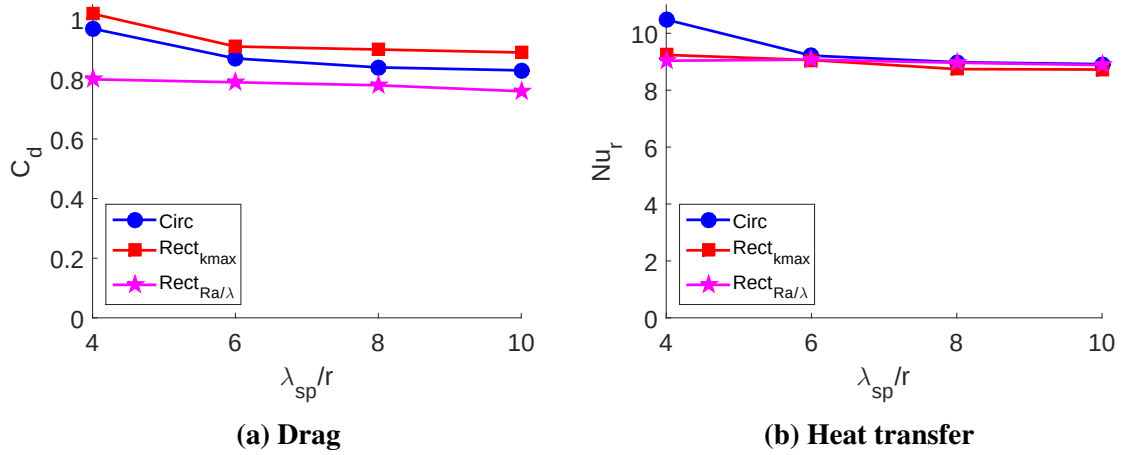


Figure 3.12 Spanwise spacing effect at $Re_r = 4200$ ($k - \omega$ SST)

as for the minimum λ_{st} the hemispherical elements demonstrate the highest drag despite of the lower frontal area, while both rectangular cases demonstrate a peak of drag at the middle spacing value. These qualitatively different trends imply significant sensitivity when streamwise spacing varies for different element shapes.

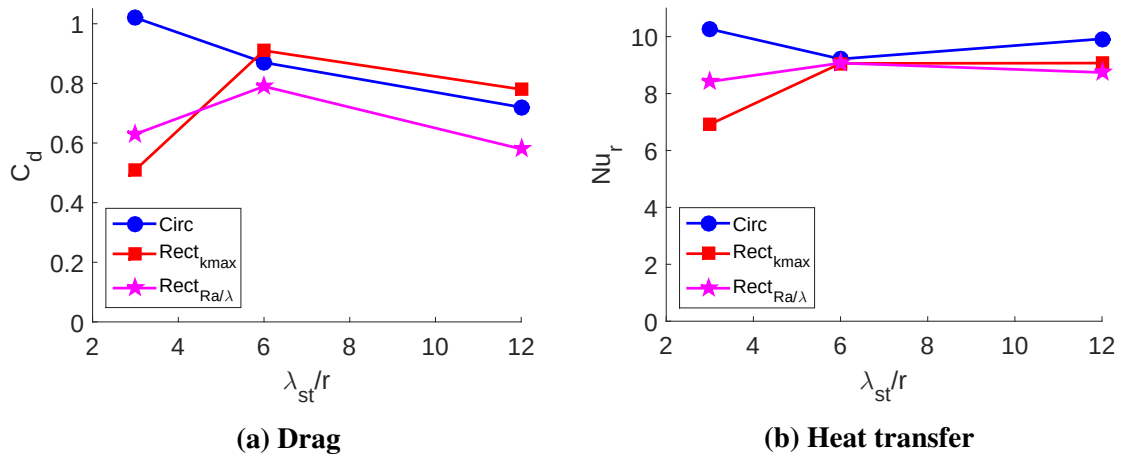


Figure 3.13 Streamwise spacing effect at $Re_r = 4200$ ($k - \omega$ SST)

Figures 3.14, 3.15, 3.16 and 3.17 present the flow patterns with different streamwise spacings for the two shapes. The vortical flow structures are seen to be altered as the elements are closely packed, sheltering the flow downstream. Especially for the rectangular elements, the space between adjacent elements is completely occupied by the separated flow structures. Also, rectangular elements demonstrate less severe flow separation when the

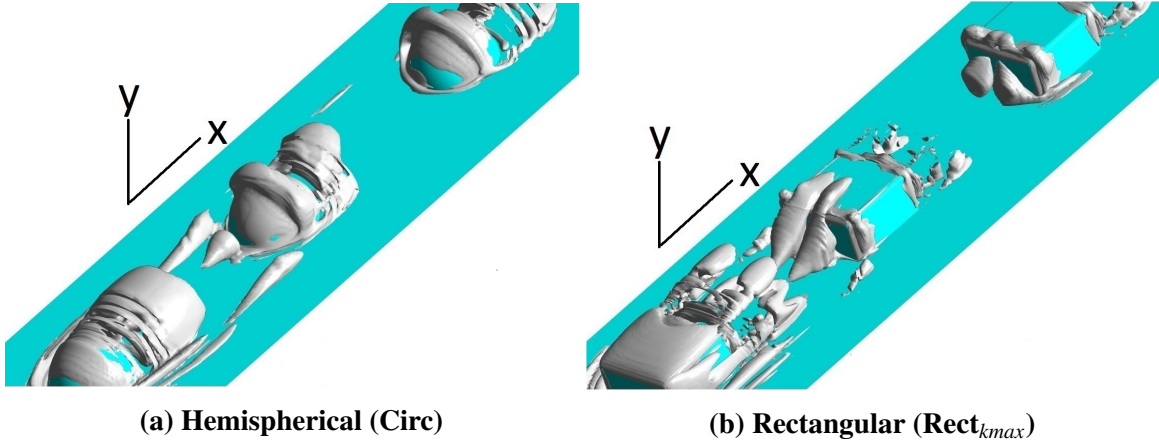


Figure 3.14 Iso-surfaces of Λ_2 criterion $\lambda_{st} = 6r$ at $Re_r = 4200$ ($k - \omega$ SST)

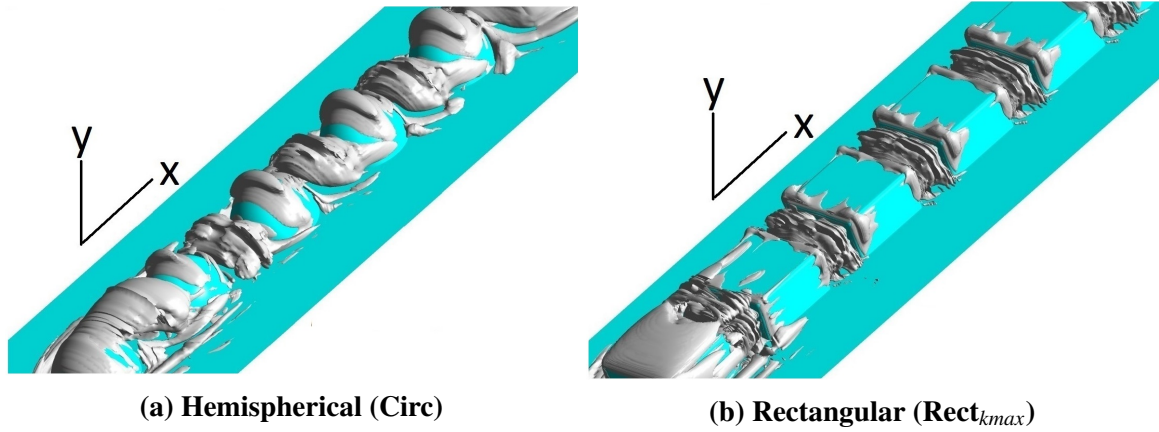


Figure 3.15 Iso-surfaces of Λ_2 criterion $\lambda_{st} = 3r$ at $Re_r = 4200$ ($k - \omega$ SST)

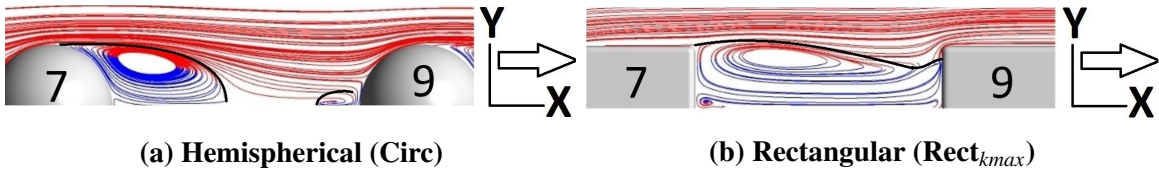


Figure 3.16 Flow pattern at midspan when $\lambda_{st} = 6r$ at $Re_r = 4200$. Element numbering reflects element id as presented in Figure 3.8a ($k - \omega$ SST)

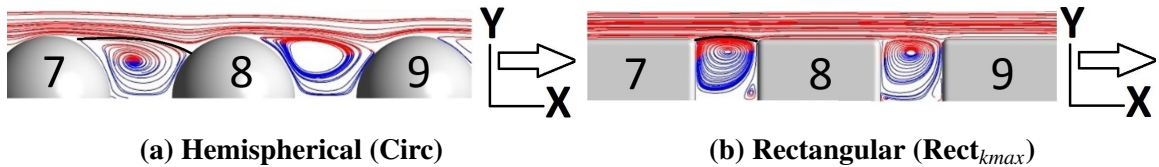


Figure 3.17 Flow pattern at midspan when $\lambda_{st} = 3r$ at $Re_r = 4200$. Element numbering reflects element id as presented in Figure 3.8a ($k - \omega$ SST)

streamwise spacing is reduced, since the flow is almost completely sheltered in the space between the roughness elements. This trapped flow of low momentum obstructs the flow from reattaching, reducing the form drag exerted on the downstream elements. For the hemispherical elements, where the separation due to the blockage is not as severe, the flow is more prone to reattachment, introducing some form drag at the upper area of each element.

Overall, in relation to the spacing effects, the main observation is that the streamwise spacing is far more influential than the spanwise spacing. This directionality of the spacing effects relative to the flow direction implies the limit of using a single relative spacing parameter for machined roughness.

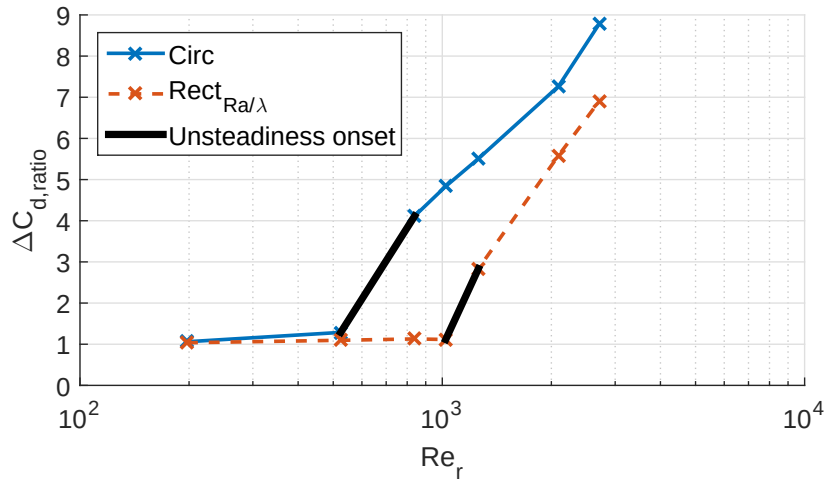
3.6 ILES Analysis for Transitional Flow Regime

The cases analysed so far are all for well-established turbulent flows subject to a 4% freestream turbulence intensity. In the final part of the study, the aerothermal behaviour of the micro-structures around a laminar-to-turbulent transition is examined.

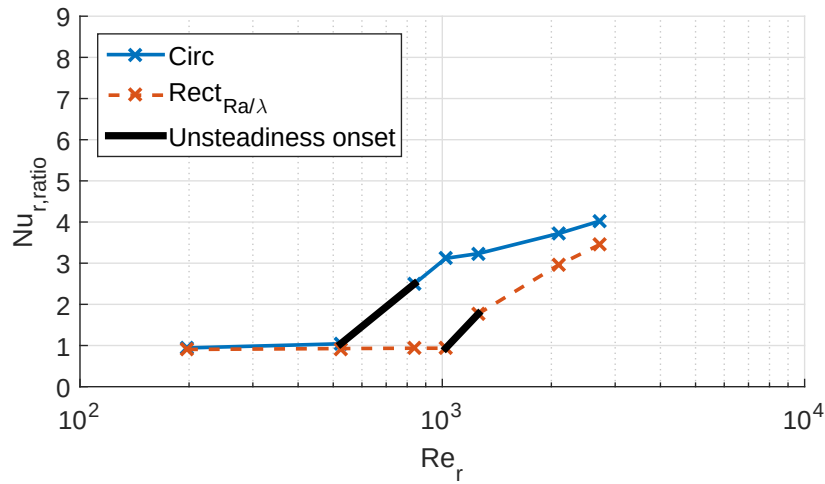
Given the above context and also the present interest in the shape effect of machined roughness elements, the hemispherical and rectangular configurations with a matched R_a/λ_{st} are now compared. The freestream inflow is completely laminar, with all other computational parameters being the same as in the preceding sections. An implicit LES (i.e. no sub-grid scale modelling) method is utilised to capture self-excited unsteadiness due to shear layer instabilities of the shed vortices. As the results will show, this unsteadiness plays a substantial role in triggering the laminar turbulent transition for the configurations and conditions concerned.

Firstly, it is of interest to get the overall picture of the performance characteristics across a transitional condition. This is given in Figure 3.18 where the drag and heat transfer variations

with relation to the Re are plotted. The transitional regime flow solutions remain completely



(a) Drag



(b) Heat transfer

Figure 3.18 Drag and heat transfer increase with Reynolds number for two shapes of DREs (transitional regime)

steady for $Re_r \leq 500$ (laminar). For this purely steady laminar flow regime, it is also shown that the aerothermal performances are largely insensitive to the streamwise spacing changes (Figure 3.19) when compared to that of a fully turbulent flow (Figure 3.13). In Figures 3.20 and 3.21 the laminar flow patterns for the two shapes are shown for two different spacings. For the small spacing (Figure 3.21), the laminar flow patterns for the two shapes are similar to each other, as well as to their turbulent counterpart (Figure 3.17). For the large spacing ($\lambda_{st} = 6r$), however, the flow patterns of the two shapes become more similar to each other

(Figure 3.20) rather than to their turbulent counterpart (Figure 3.16). The low sensitivity of the flow pattern to the element shape in the laminar regime is in agreement with the conventional expectation that the laminar flow should be largely insensitive to roughness, which also explains the low sensitivity of the two shapes to the spacing in this regime.

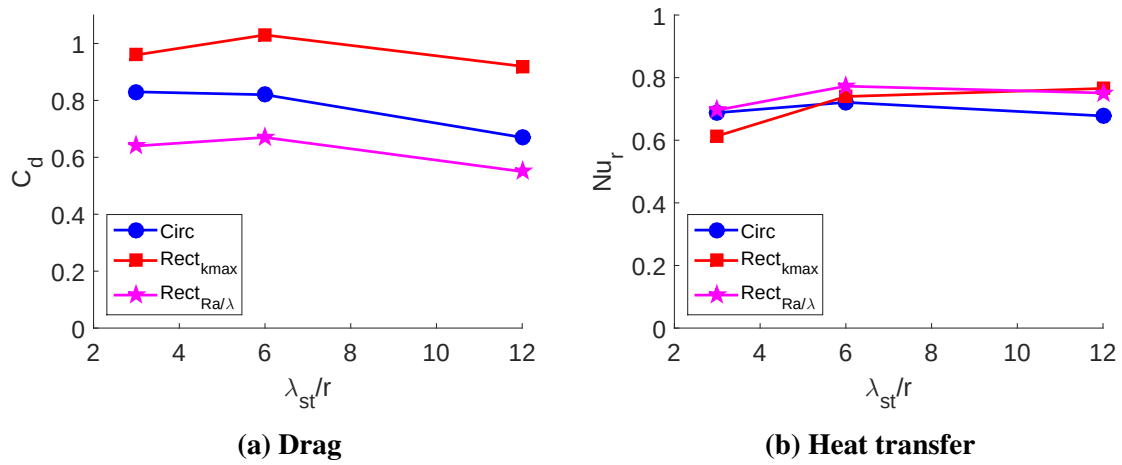


Figure 3.19 Streamwise spacing effect at $Re_r = 200$ (laminar)

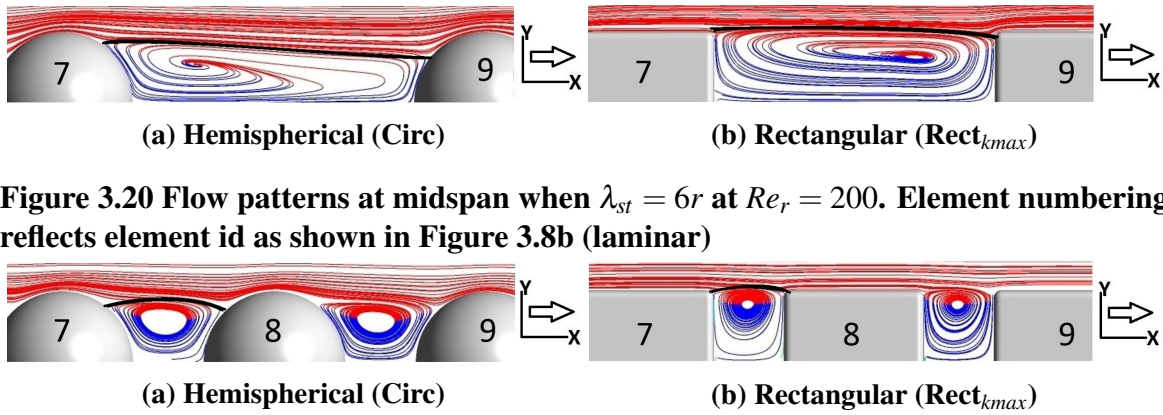


Figure 3.20 Flow patterns at midspan when $\lambda_{st} = 6r$ at $Re_r = 200$. Element numbering reflects element id as shown in Figure 3.8b (laminar)

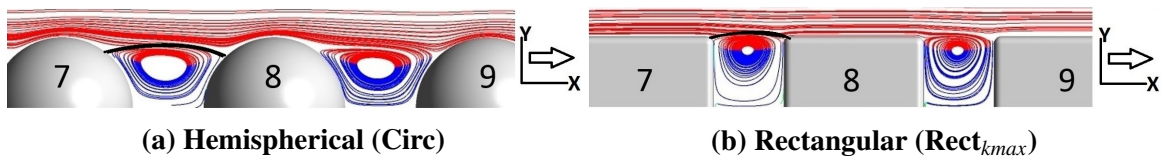


Figure 3.21 Flow patterns at midspan when $\lambda_{st} = 6r$ at $Re_r = 200$. Element numbering reflects element id as shown in Figure 3.8b (laminar)

Now the impact of the transition is considered. Interestingly, the onsets of the unsteadiness, as identified by the thick lines in Figure 3.18, seemingly mark the start of the high drag/heat transfer regime. The changes can be comprehended more clearly by comparing the flow patterns at $Re_r = 200$ and $Re_r = 2700$, i.e. before and after the unsteadiness onset, respectively. The induced high unsteadiness can be seen in Figure 3.22, while the steady and instantaneous

vortical flow patterns are compared in Figures 3.23 and 3.24, respectively. Clearly, there are two qualitatively different flow patterns at these two Reynolds numbers. The results suggest the onset of unsteadiness should mark the start of the transition, even though resolving a detailed transition process is not of the current interest. At the condition of $Re_r = 2700$, the established turbulent flow patterns emerge for both shapes (Figure 3.24).

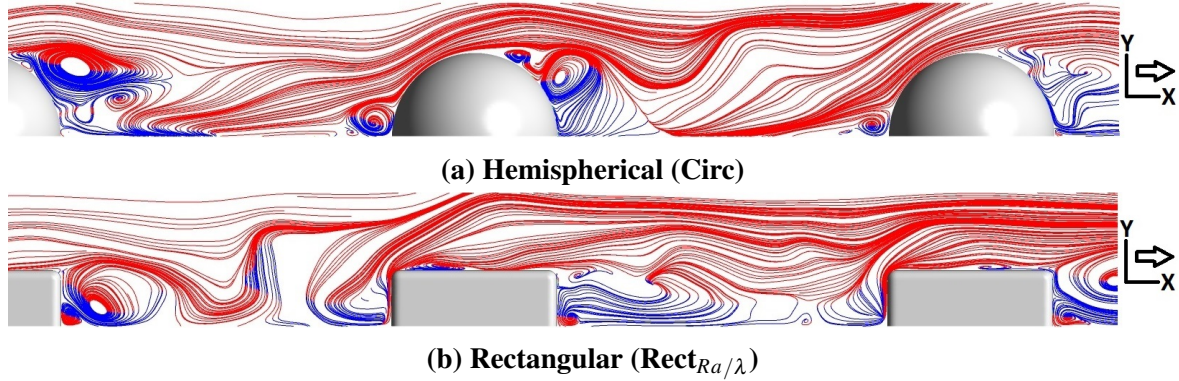
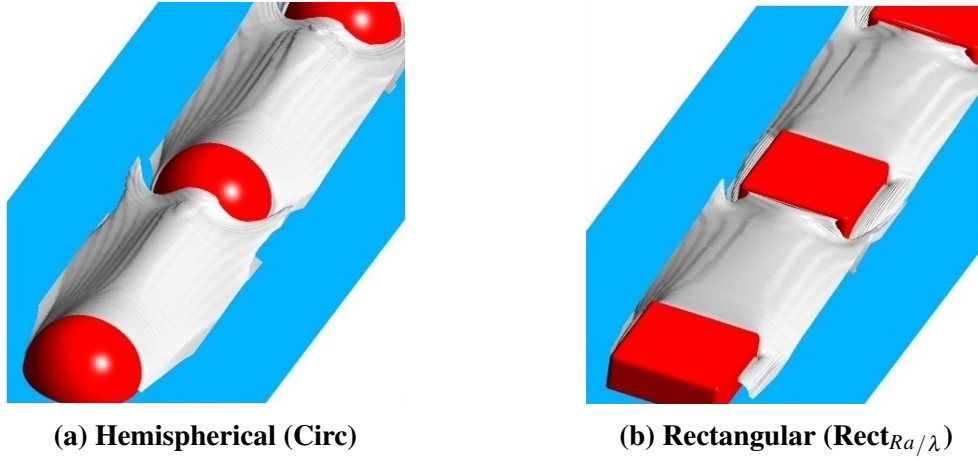
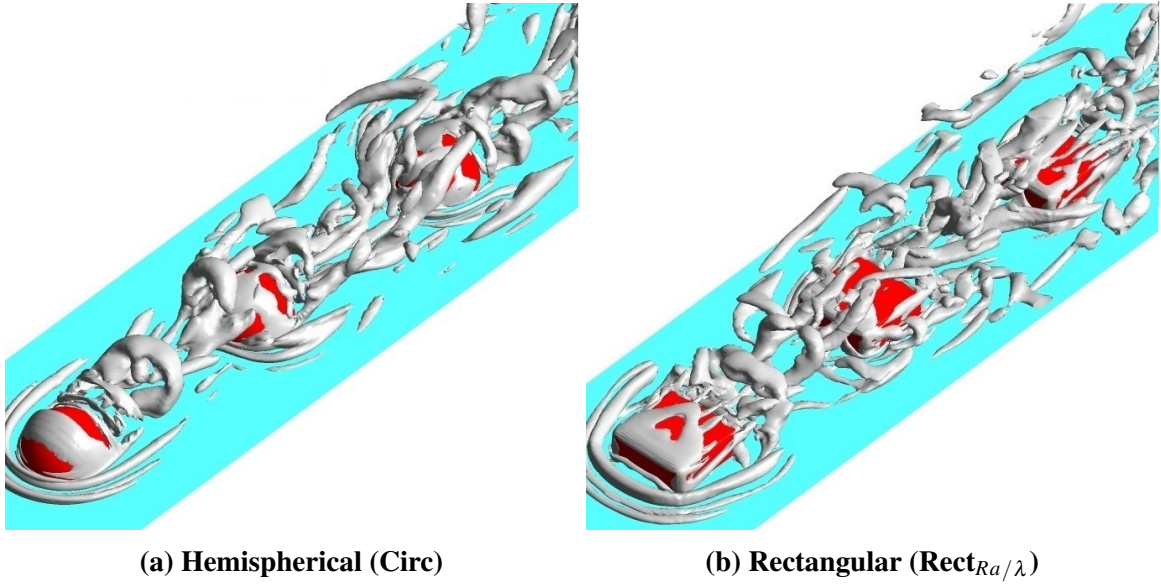
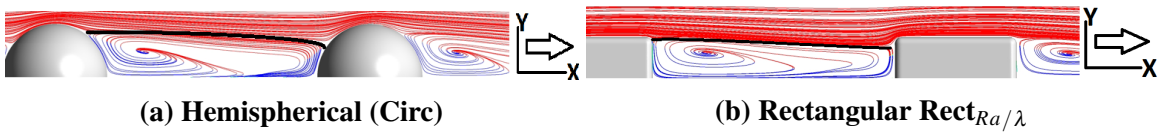
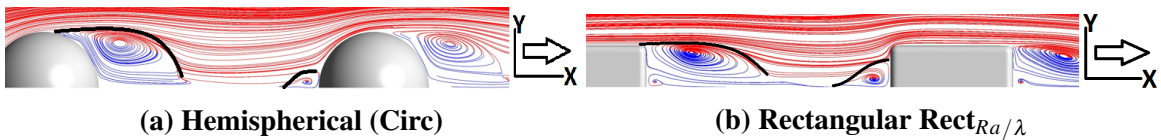


Figure 3.22 Flow patterns (instantaneous) at midspan at $Re_r = 2700$ (ILES)

A more revealing comparison is in the time-averaged flow patterns for the two Reynolds numbers before and after the transition, as shown in Figures 3.25 and 3.26, where a common and insightful pattern change seems to emerge. For a low drag pattern at a low Reynolds number (Figure 3.25), an open-type separation occurs with separated shear layers impinging on the upper part of the downstream elements. On the other hand, for a high loss pattern (Figure 3.26) the separated flow reattaches before reaching the downstream element, which is now subject to a newly formed boundary layer.

It is quite remarkable to see the similarity in the flow pattern change between this transitional flow regime and a fully turbulent one, as analysed earlier in Figure 3.10. This similarity, though only qualitative (given the marked differences in the flow conditions and turbulence models used), seems to indicate that the reattached flow pattern is the key feature marking a rapid rise in drag and heat transfer. It is suggested that the two distinctively different flow regimes might be simply two different paths to reach the same pattern with

Figure 3.23 Iso-surfaces of vorticity at $Re_r = 200$ (Laminar)Figure 3.24 Iso-surfaces of Λ_2 criterion (instantaneous) at $Re_r = 2700$ (ILES)Figure 3.25 Flow patterns (time averaged) at midspan at $Re_r = 200$ -last two elements downstream-(ILES)Figure 3.26 Flow patterns (time averaged) at midspan at $Re_r = 2700$ -last two elements downstream-(ILES)

a reattached separation flow by enhancing turbulent mixing. For an already turbulent flow, increased turbulent mixing at a higher Reynolds number may be one path to reach the high loss flow pattern. Similarly, for a laminar transitional flow, the unsteadiness due to shear layer instabilities and the subsequent transition may well be another path to do so, as seen here.

Finally, in Figure 3.27 the drag coefficient C_d for the transitional and the fully turbulent flow regimes is presented for the cases Circ and Rect_{Ra/λ}. For the Re range where both flow regimes are studied ($Re_r = 8 \times 10^2 - 3 \times 10^3$) there are distinctive differences in the values and trends of the C_d , which are case-dependent, between the two flow regimes. For both cases, C_d drops monotonically as Re increases when fully turbulent flow is assumed. However, for the transitional flow regime more complex behaviours are shown. Firstly, the case consisting of hemispherical micro-structures (Circ) demonstrates an unsteady behaviour (ILES) for the Re regime where the fully turbulent (RANS) regime is also examined. However, the drag coefficient is not monotonically decreasing, as it is the behaviour in the fully turbulent regime. Initially, there is a subtle increase before the C_d values start dropping with increasing Re , and the maximum relative difference of the predicted C_d between the ILES and the RANS solutions is 55%. For the Rect_{Ra/λ} case, the flow is still transitioning from laminar to unsteady. With increasing Re , the values of C_d are dropping before the onset of the unsteadiness, then start increasing rapidly until they start decreasing again. For $Re_r < 1.2 \times 10^3$ the maximum relative difference of the predicted C_d between the ILES and the RANS solutions is 277%, whereas for $Re_r > 1.2 \times 10^3$ the maximum relative difference is 28%. These observations serve as an example of the sensitivity of the flow predictions to this Re_r regime and, therefore, they demonstrate the need for detailed resolution of such micro-structures when numerical predictions are sought.

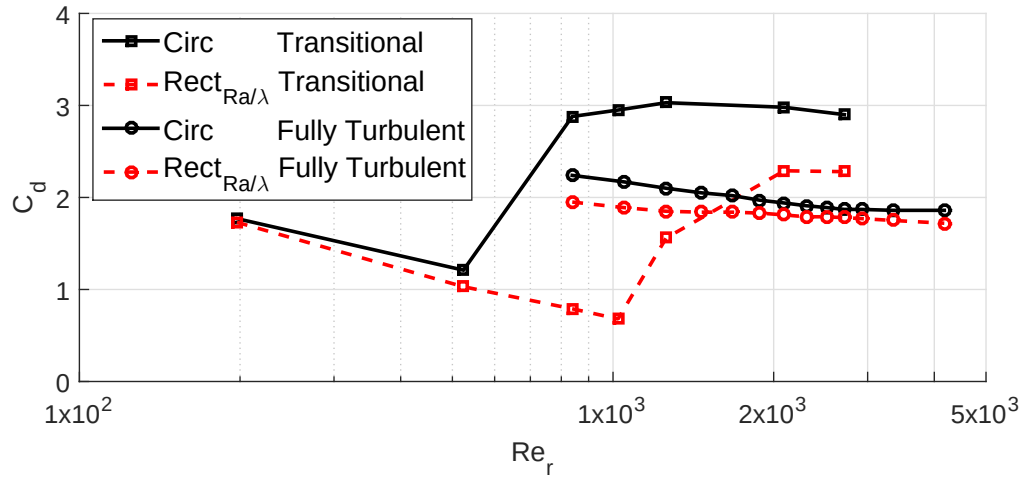


Figure 3.27 Comparison of the C_d between the transitional and fully turbulent flow regimes

3.7 Remarks

In recognition of the potential of manufacturable micro-structures, a computational study was carried out to address how machined roughness patterns and shapes would affect the aerothermal behaviour.

Care has been taken in examining grid-dependency and turbulence sensitivity of the results. Several turbulence models were utilised for RANS in a higher Reynolds number regime, and additional laminar and LES analyses were conducted for lower Reynolds number conditions. The contrast between higher and lower Reynolds number regimes and between different modelling fidelities assisted in distinguishing driving aerothermal mechanisms and modelling requirements in a fully turbulent flow or in a transitional flow.

As part of both the validation and the investigation of the effects of roughness pattern, a recently established roughness parameter was also examined. The present results show that the non-dimensional roughness spacing parameter R_a/λ , recently put forward for correlating flow losses subject to stochastic roughness by Goodhand et al. [33], appears to be qualitatively also effective for regular micro-structures in a well-established turbulent regime.

The RANS models, though inherently less consistent for bluff body flows, are shown to be able to capture the loss features qualitatively in agreement with the previous experimental observation. However, the transitional behaviour and its impact can only be adequately captured by LES. In this case, the absence of a sub-grid model would effectively lead the implicit LES towards coarse mesh DNS.

When transition is considered, detailed shapes are shown to alter the onset of the unsteady flow, thus the onset of shear layer instabilities and transition. Consequently, the critical Reynolds number for which the flow is tripped into an unsteady flow is shape dependent.

To distinguish between stochastic and regular roughness, the present results reveal a clear and strong dependence of the aerothermal behaviour on the element pattern orientation relative to the primary flow. This flow-roughness pattern directionality is not surprising but it is something that needs to be considered for regular roughness. This directional dependence clearly challenges the adequacy of the single roughness spacing parameter R_a/λ .

The present analyses reveal an insightful common flow structure which marks the change from a low loss flow pattern to a high loss one. The key feature appears to be the downstream reattachment of the separated flow from the upstream element. When the upstream shed shear layer impinges on the downstream element, the whole downstream separated region is effectively shielded, thus the overall drag/loss is low. On the other hand, when the separated shear layer reattaches to the base surface before the downstream element, a new boundary layer starts to grow, leading to extra loss/drag generation mechanisms. A possible common link between a fully turbulent flow and a transitional flow in establishing this high loss flow pattern was suggested.

It was shown that the shape and pattern of regular roughness elements affect aerothermal behaviour due to their interactive wakes. To accurately capture the interaction between the

DREs and their unsteady wakes the computational mesh needs to be adequately fine. Such a fine mesh for solving micro-structures and the main flow domain would be far beyond what is currently adopted in common CFD analyses. Hence, the question remains: how can the resolution of such fine scale be incorporated in the computational solutions without increasing the size of the problem significantly?

A multi-scale approach seems to be the most appropriate, as this will allow for the mesh size to increase only at the vicinity of interest. Yet, when considering a large number of such micro-structures, as a realistic application would require, the multi-scale mesh size will become unmanageable. In the rest of this thesis the adopted computational modelling approach that aims at solving this problem is discussed.

Chapter 4

Present Computational Methodology

So far, it has been demonstrated that roughness can introduce significant aerothermal effects. These effects stem from the interacting nature of the wakes shed by the individual roughness micro-features, which in turn collectively affect the bulk flow. Now the following questions need to be addressed:

- How can accurate numerical predictions be generated?
- How can both the large flow scales of the bulk flow and the fine turbulence induced by the roughness features be accommodated?
- How can numerical predictions of large multi-scale domains be performed while consuming realistic amount of computational resources?

The first question is effectively answered in Chapter 2. There, it was discussed how conventional computational methods would fail to accurately capture regular roughness effects. Hence, a multi-scale approach is suggested. Employing two coupled problems, one with a fine mesh and one with a coarse mesh, could bridge the need for low consumption of computational resources and appropriate resolution of the flow scales of interest.

This chapter discusses how the numerical prediction of the aerothermal effects of large number of DREs could be performed. Initially, the baseline solver is described. Then, the following material is split into two parts: the Embedded Dual-Mesh (DM) part and the Block Spectral Method (BSM) part. The former presents the adopted baseline multi-scale approach, whereas the latter discusses the method that could accelerate the baseline multi-scale configuration, both based on He [63–66].

4.1 Baseline Solver

In this section the main characteristics of the baseline flow solver are described. Initially, the general traits of the serial solver, e.g. mesh, numerical scheme, and governing equations, are presented. Next, the limitation of the employed programming language when it comes to memory consumption and the adopted solution is discussed. In the end, an overview of the algorithm of the solver is presented.

4.1.1 General Information

The baseline finite-volume cell-centered Navier-Stokes solver has been developed over many years, initially based on the Jameson central difference type discretisation [70, 71]. It was later extended to a multi-block mesh solver with the option to overlay embedded domain for local mesh refinement [72]. The solver allows for arbitrary base block connectivity, e.g. a single block face interfacing with multiple block faces or O-grids.

In addition, the density-based $AUSM^+$ up-winding discretisation (see appendix B) was implemented [64]. It accommodates the second order limiter MUSCL (hence 5 mesh points are required for flux calculations) and an extension of it to a third order based on Cada et al. [73]. The third order limiter requires additional calculations with the same amount of 5 mesh cell centres employed by MUSCL.

Since this code was primarily developed for turbomachinery applications, the code utilises cylindrical coordinates. Hence, for Cartesian problems the radius is set arbitrarily high, such that numerical truncation error is not introduced, yet the domain resembles a Cartesian one. The flow is governed by the 3D unsteady conservation equations of mass, momentum and energy expressed as:

$$R(\Phi) = 0 \quad (4.1)$$

where $\Phi = [\rho, \rho u, \rho r v, \rho w, \rho E]^T$.

Fluid viscosity follows the Sutherland's law [68] while specific heat capacity is constant per mesh block. The boundary conditions include the common option used in turbomachinery flow solvers: the stagnation pressure, stagnation temperature and the flow angle at the inlet, the static pressure at the outlet, non-slip condition for viscous walls and the symmetry condition for inviscid walls.

Turbulence modelling is performed either with the one equation Spalart-Allmaras (SA) [74] or the two-equation SST $k - \omega$ [75] models. For LES the Smagorinsky subgrid turbulence modelling [76] can be employed.

An explicit four-stage Runge-Kutta scheme is used for time integration through the pseudo-time regime. For unsteady flows implicit Dual Time Stepping has been implemented as described by Jameson [61]. So, from Equation 4.1 the system to solve is transformed into:

$$\frac{\partial \Phi}{\partial \tau} + R(\Phi) = 0 \quad (4.2)$$

where τ is the pseudo-time. For each time-step in the real time regime, a steady problem is solved in the pseudo-time regime. The temporal derivative is expressed by a backward

second-order scheme:

$$\frac{\partial \Phi_k}{\partial t} = \frac{1.5\Phi_k - 2\Phi_{k-1} + 0.5\Phi_{k-2}}{\Delta t} \quad (4.3)$$

Though the Dual Time Stepping has been implemented, an explicit method is used for the real-time marching, necessitating a reduced CFL number to ensure numerical stability. The numerical stability CFL number is defined as:

$$CFL = \frac{U_{ref} \Delta t}{\min \text{ cell dimension}} \quad (4.4)$$

For the dual-mesh configuration, there is also the option to perform the time-marching with a multi-step approach, i.e. the fine mesh performs multiple sub-time-steps at each single coarse mesh time-step. Alternatively, a single time-step, determined by the fine mesh, applied on both the coarse and fine mesh can be utilised.

In Figure 4.1 the main actions of the solver are illustrated. In the initial step **Setup** the domain mesh and properties are configured and the embedded mesh is generated. In the following action, the mesh blocks connectivity is identified and stored so that the information is readily available. In the **Calculate viscous terms** the stress and heat terms of the governing equations are calculated. Finally, the **Update flow properties** step evaluates the fluxes and updates the flow properties, with more details being discussed in the following sections. There are additional minor actions performed as well, e.g. checking convergence, that are not mentioned for simplicity.

4.1.2 Data Storage

As mentioned, the solver works with multi-block dual-meshes, with the whole domain consisting of multiple mesh blocks. Considering the baseline solver is written in Fortran 77, information can be stored only in static matrices. Conventionally, each mesh block is

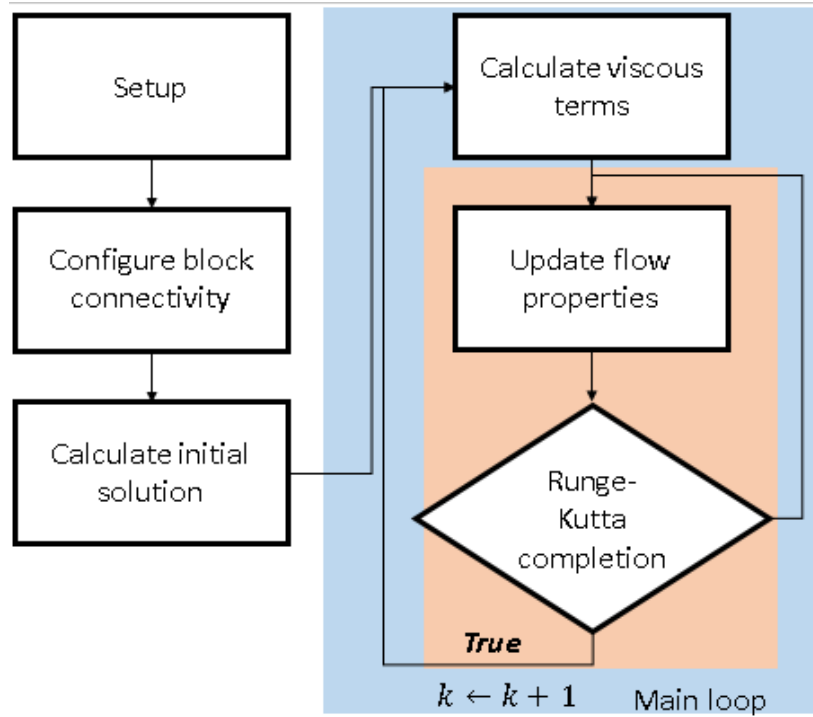


Figure 4.1 Schematic of the solver algorithm

associated to a single data block. Therefore, memory shortage issues may appear for domains consisting of large blocks along with multiple small blocks. The need for defining a maximum matrix size based on the largest mesh block leads to excessive, unused allocated memory from storing the small blocks.

For example, consider a large base block of the size M and N small blocks of the size $M/1000$. For this mesh $(1 + N) \times M$ data cells need to be allocated, while the total number of mesh cells is equal to $(1 + N/1000) \times M$, as illustrated in Figure 4.2a. The ratio of used allocated memory over the total allocated memory would be $(1 + N/1000)/(1 + N)$. For $N = 1$ the value would be 50%, while for a high number of N small blocks the ratio could be 0.1%.

Therefore, to avoid domain size restrictions due to computational memory shortage, the mesh blocks are stored in a more compact manner. Here, each data block does not correspond to a mesh block, but multiple small blocks can be stored in one data block, as shown in Figure 4.2b. This approach may add challenges to the parallelisation of the serial solver, yet it allows

for a near-wall region with a large number of DREs to be fitted with embedded blocks at discrete regions with sensible memory consumption.

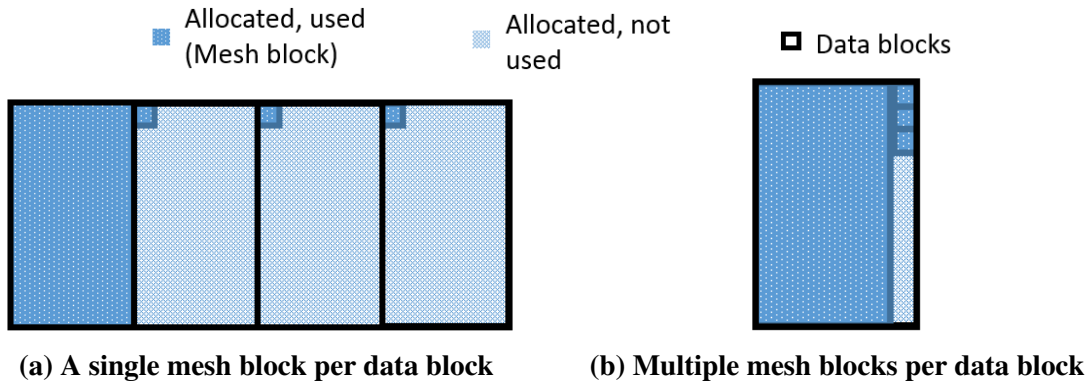


Figure 4.2 Schematic of data and mesh blocks

4.2 Embedded Dual-Mesh Multi-Scale Approach

Initially, the implications of the dual-mesh approach on the mesh generation are discussed. Following this, the coupling of the two-scale domains is presented.

4.2.1 Mesh Generation

Let us consider a 2D flat plate fitted with rectangular DREs, as illustrated in Figure 4.3. The objective is to adequately resolve the flow scales in the bulk and near-wall regions. As briefly mentioned, a way to do so is by constructing a multi-scale problem.

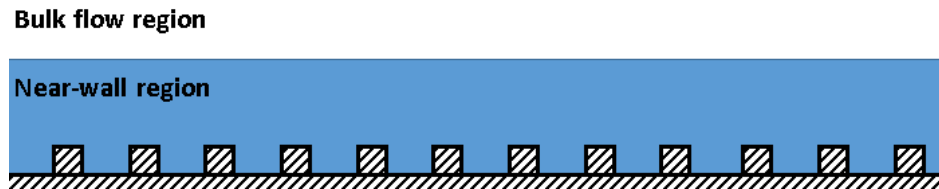


Figure 4.3 2D flat plate with fitted micro-structures

Here the embedded dual-mesh is adopted. Initially, a coarse (base) domain is generated that is adequately dense for capturing the bulk flow structures. The coarse mesh spans across both the bulk and near-wall region. The next step is to generate a fine domain that spans across

only the near-wall region. The two decoupled meshes are illustrated in Figure 4.4a where the near-wall mesh refinement is clearly needed. The bulk, coarse mesh in this particular example is unable to even capture the geometric properties of the DREs, thus the fine domain is placed on top of the coarse domain, as shown in Figure 4.4b. In other words, the fine mesh is embedded into the coarse mesh domain, similarly to the IMB approach [62]. It is important to note that the coarse mesh at the near-wall region is not discarded.

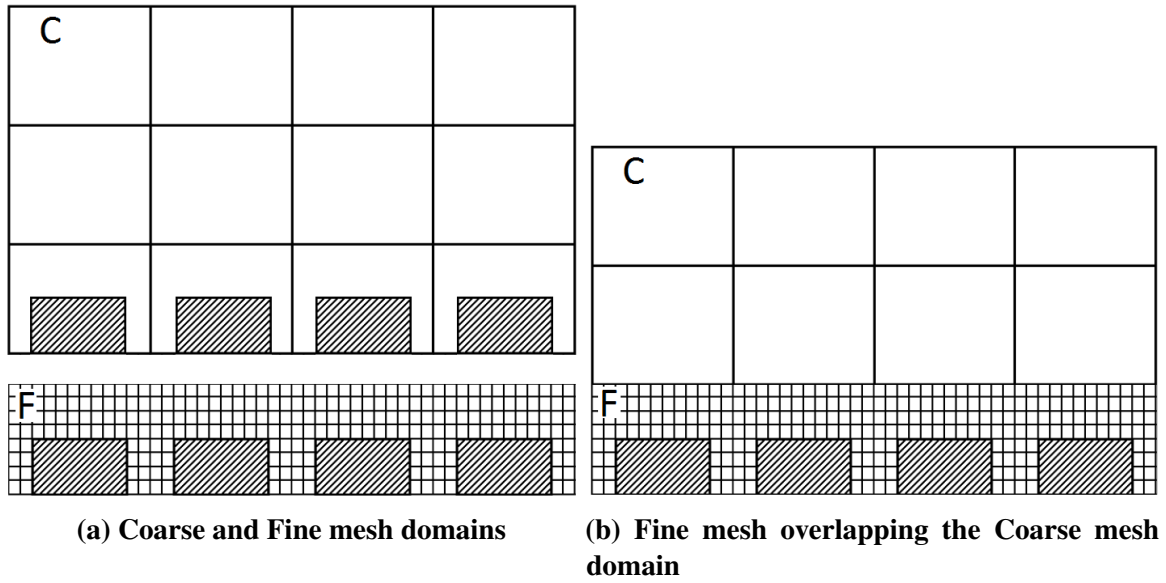


Figure 4.4 Dual-Mesh. C and F denote Coarse and Fine mesh, respectively

The scope of the presented work focuses on regular roughness. Since the in-house solver handles structured meshes, the above approach is quite suitable when refining the mesh at an area of interest. The fine (embedded) mesh is created by dividing the already existent coarse (base) mesh cells. For a repetitive spatial configuration, the fine domain may be generated for a single spatial period. Following this, the fine domain is copied and translated until the whole near-wall region is covered. Furthermore, this approach demonstrates the specific practicality that locally increased resolution may be achieved without altering the whole mesh blocking strategy. In Figure 4.5, a comparison between the multi-block (a) and the dual-mesh (b) domain configurations is demonstrated. Assuming that refinement is required in a certain

region of the domain, the multi-block approach would require re-blocking of the structured mesh. On the other hand, the dual-mesh approach would simply create an embedded block with fine mesh overlapping the initial coarse/base mesh to cover the area of interest.

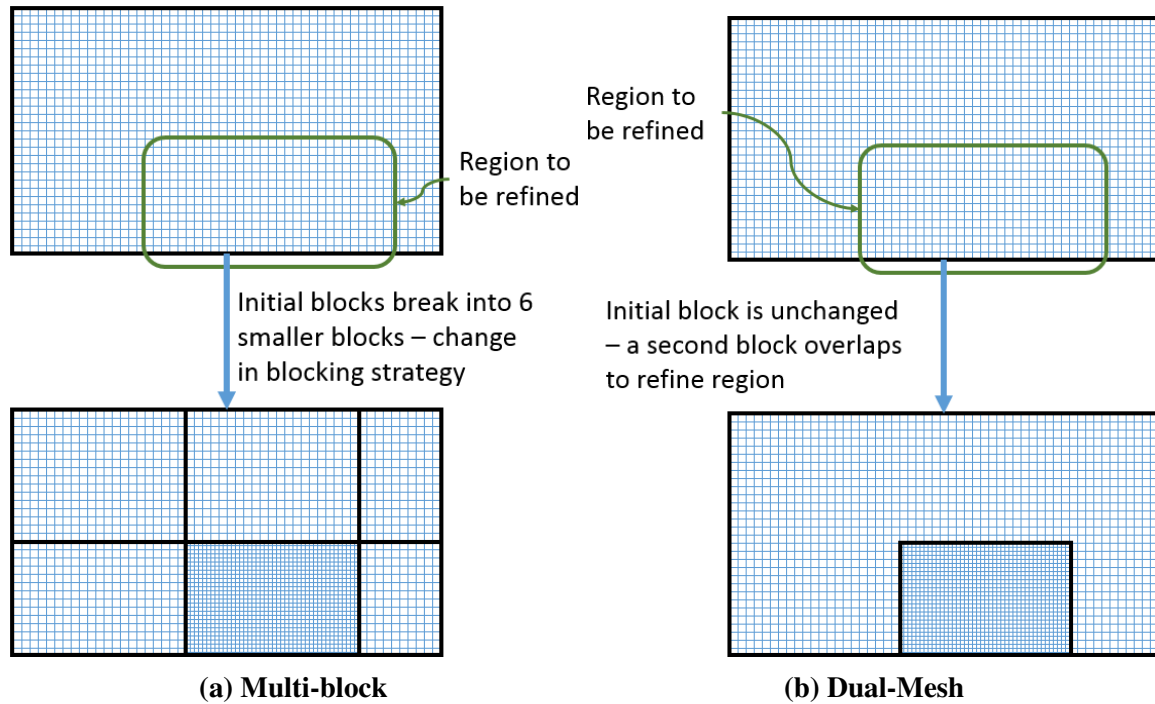


Figure 4.5 Comparison between the multi-block meshing approach and the dual-mesh (embedded mesh) approach

Another feature utilised in the presented dual-mesh approach is the embedded wall. A solid domain can be introduced in the existing mesh by specifying viscous walls along grid lines. It is imperative that the specified solid-cells create a closed volume so that the enveloped cells are identified as to be solid. The boundaries of the solid domains are treated as viscous walls, i.e. with no-slip conditions, and the enveloped solid regions are passive to the solution. In future work, this feature can be extended to accommodate conjugate heat transfer for the embedded solid domains. Again, the benefit of this feature can be understood from Figure 4.6, in combination with the embedded block feature, as the embedded mesh and embedded solid combination allows for the efficient creation of a near-wall region capable of capturing micro-geometric features.

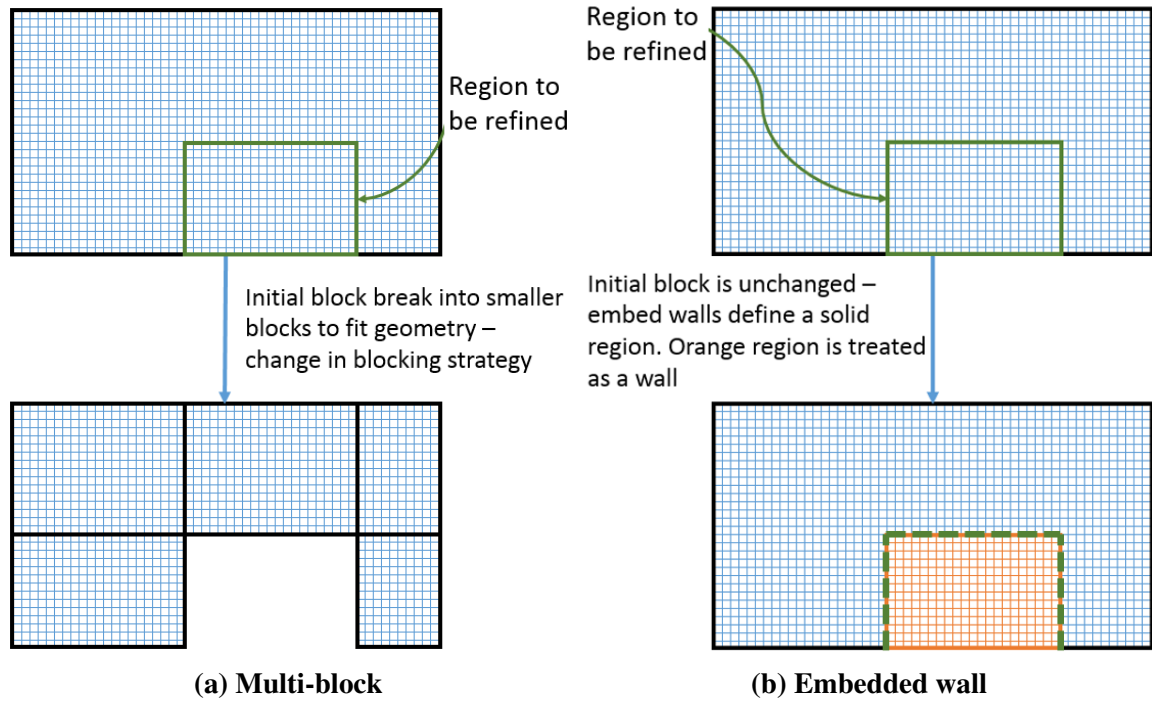


Figure 4.6 Comparison between the multi-block meshing approach and the embedded wall (solid) approach. Dashed lines denote the embedded viscous walls

4.2.2 Dual-Mesh Coupling

Having described the mesh generation, now the coupling of the two-scale problem needs to be established. For this, a two-way communication is applied:

- The fine domain updates the primary flow properties of the bulk domain in the near-wall region, as well as the calculated fluxes at the coarse-fine domains interface.
- The coarse domain updates the boundary flow properties of the fine domain.

The affected domain regions are illustrated in Figure 4.7. Here, a third mesh characterisation is introduced, the C_F . This region is a subregion of the coarse C domain and it is updated by the fine F domain.

As mentioned in the previous subsection, the fine mesh domain is generated by splitting the existing computational cells of the coarse mesh domain. Therefore, the coarse mesh primary flow properties located at the near-wall region can be simply updated based on

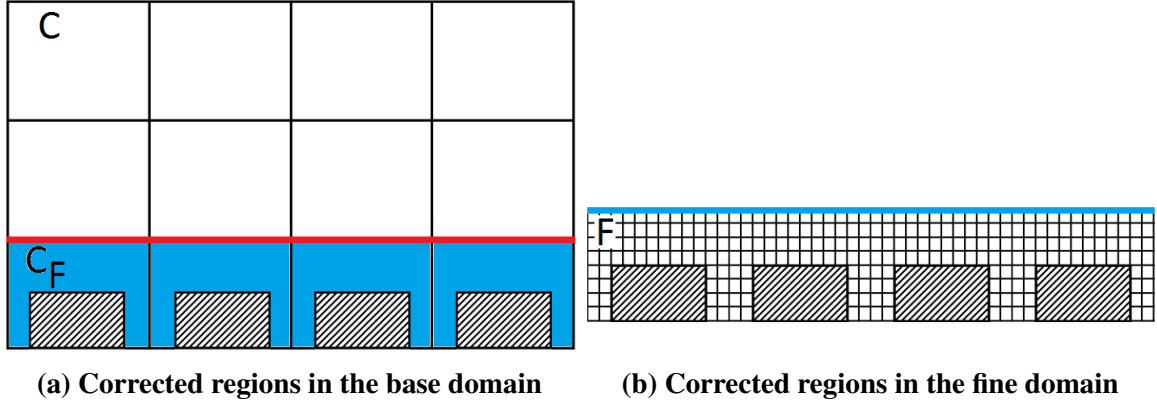


Figure 4.7 Corrected regions in each domain due to coupling. Blue and red indicate corrected flow properties and corrected face fluxes, respectively

the collective effect of the fine flow structures. This is performed by volume-averaging the flow properties of the corresponding fine mesh computational cells. Thus, for a single computational cell of the C_F domain the primary flow properties Φ are updated by N_F^V fine cells, where N_F^V the number of the fine cells refining the coarse cell, as:

$$\Phi_{C_F} = \frac{\sum_{n=1}^{N_F^V} V_F(n) \Phi_F(n)}{\sum_{n=1}^{N_F^V} V_F(n)} \quad (4.5)$$

where $V_F(n)$ is the volume of each n fine mesh cell.

Similarly, the fluxes of the coarse domain at the base-embedded domains interface are updated by the fluxes of the fine domain at the interface through area-averaging as:

$$flux_{C_F} = \frac{\sum_{n=1}^{N_F^A} A_F(n) flux_F(n)}{\sum_{n=1}^{N_F^A} A_F(n)} \quad (4.6)$$

where $A_F(n)$ is the face area of each n fine mesh cell at the base-embedded mesh interface and N_F^A the number of fine mesh faces refining a single coarse mesh face.

The ratio of the coarse cell size to the fine cell size can be significantly greater than unity. Thus, when updating the boundary flow properties of the fine mesh domain, a careful consideration of the chosen coarse mesh properties is required. Hence, to ensure sensible

flow variable values are used as boundary conditions of the embedded domain, the base information is linearly interpolated at the interfaces of the embedded blocks. Consider Figure 4.8, where a 2D dual-mesh is shown, with the coarse mesh being refined 4×4 times by the embedded mesh. $\times$ denotes the assumed location of the flow property of a coarse mesh cell, while $\circ$ denotes the virtual location of the boundary values of the embedded mesh. The boundary values are determined by second order interpolation of the coarse mesh values. In the presented example, F_1 and F_2 values are the interpolated values of $C_{1,1}$, $C_{2,1}$, $C_{1,2}$ and $C_{2,2}$ and the F_3 and F_4 values are the interpolated values of $C_{2,1}$, $C_{3,1}$, $C_{2,2}$ and $C_{3,2}$. For 3D meshes, the interpolation is of third order. In the present implementation, the accuracy of the evaluated fine domain fluxes at the embedded-base interface is of the first order.

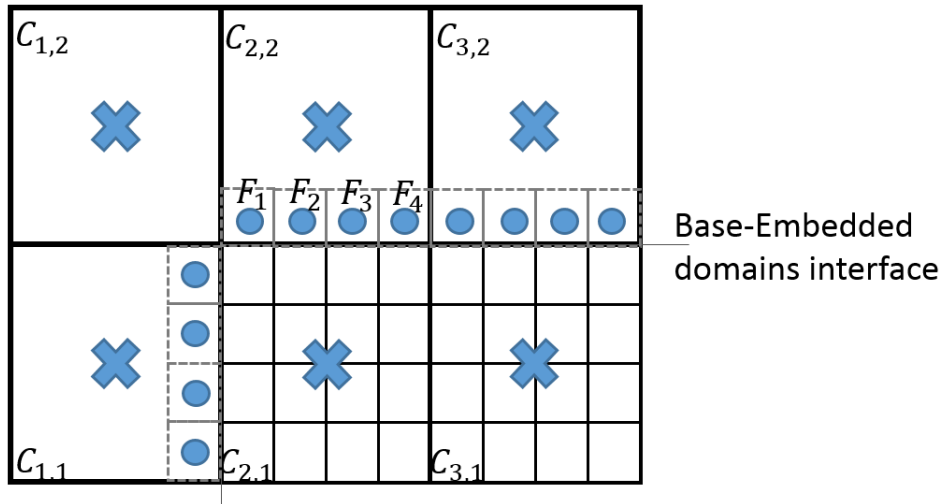


Figure 4.8 Schematic of embedded block boundary treatment. Gray (dashed) squares denote halo cells

So far it has been assumed that the whole near-wall region is resolved by an embedded domain. However, for large domains this approach may require a significant amount of computational resources. Still, instead of refining the whole near-wall region, it is possible for only some discrete parts of the near-wall regions to be assigned an embedded domain, as shown in Figure 4.9. A fully refined and a discretely refined near-wall region will be also referred to as fully embedded and partially embedded configurations, respectively. Partial

embedding can be problematic, as a significant area of the embedded F mesh blocks are not subject to the same scales of flow structures; the boundaries will only receive coarse mesh flow properties, affecting the prediction accuracy of the fine mesh domains.

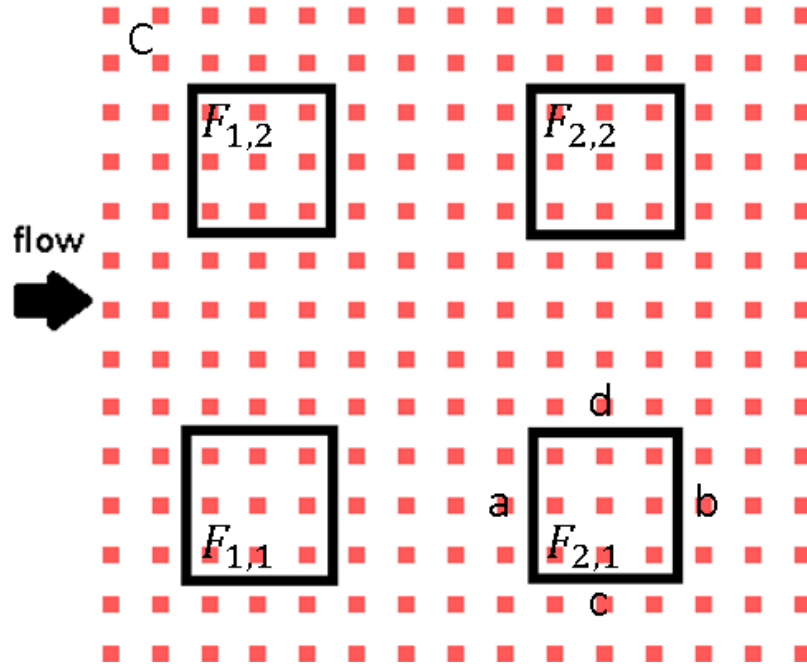


Figure 4.9 Top view of a flat-plate fitted with cubic DREs. Four discrete regions, each resolving 9 DREs, are refined by embedded mesh blocks

This can be remedied by recycling flow fluctuations of the fine domain. Considering the primary flow property Φ at the boundary of the embedded domain, the following decomposition into a coarse mesh component Φ_C and a fine mesh fluctuation Φ'_F can be assumed:

$$\Phi = \Phi_C + \Phi'_F \quad (4.7)$$

The Φ_C component is the interpolated coarse mesh value, as discussed above. The Φ'_F fluctuation information can be acquired in two ways:

- From the same embedded mesh block. For example, from the above figure block $F_{2,1}$ could have its upstream boundary a fluctuations copied from the downstream boundary b . Similarly, boundary c could pass its fluctuations to boundary d and vice versa.

- From a neighbouring embedded mesh block. Choosing a different embedded mesh block would decorrelate the mesh points of the same mesh block. For example, from the above figure block $F_{2,1}$ could have its upstream boundary a fluctuations copied from the upstream boundary a of the $F_{1,1}$, $F_{1,2}$ or the $F_{2,2}$ blocks.

This is a similar approach to the enforcement of inlet turbulence based on the turbulence at the exit of the domain in LES of a fully developed channel flow (e.g. [34]).

Summing up the above analysis, the **Update flow properties** step for the coupled two-scale problem is presented in Figure 4.10, where the coupling actions are performed within this loop. For the coarse domain, firstly the fluxes are corrected at the **Update flux_{C_F}**, so that the correct primary flow properties are evaluated for the cells adjacent to the base-fine interface. Then the flow properties at the **Update Φ_{C_F}** step are calculated via volume-averaging. The coarse flow properties interpolation and the exchange of the fine boundary fluctuations takes place at the **Update boundaries of F domain** step.

4.3 Block Spectral Method

In practical situations, a large number of micro-structures (e.g. $\sim 10^3 - 10^4$) on a typical surface of interest would require an extremely large number of mesh points in order to capture the unsteady interaction between the roughness elements themselves and between the roughness elements and the main flow, making this multi-scale problem prohibitive. One could argue that this can be tackled in the near future when computers will demonstrate higher computational performance, based on Moore's law. According to Moore's law the amount of transistors in an integrated circuit, thus the computational power, doubles in a period of time of roughly eighteen months, a trend that has been followed for almost a half century. However, there are indications that this trend is reaching the top of the curve due to material

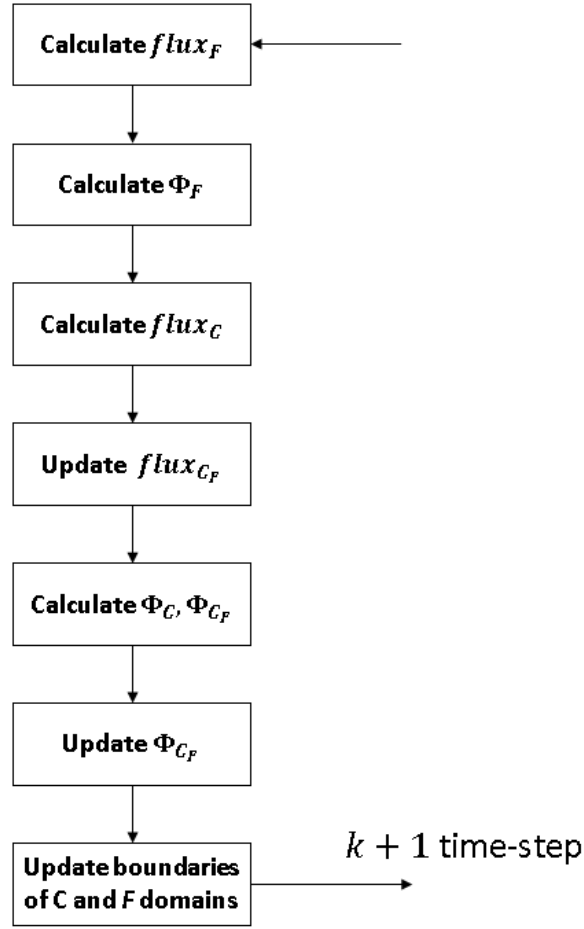


Figure 4.10 Update flow properties: Dual-Mesh

limitations, while quantum computers are still in an early stage of development. Hence, it would be useful to explore other options, rather than to simply wait for the evolution of HPC, which, even if it becomes feasible, it would be expected to be costly.

Previously, it was discussed how the near-wall region can be refined at discrete regions (partial embedding). Although this reduces the mesh size, there are implications to the accuracy of the prediction, as the non-refinement of all the fine scale regions would introduce a history error. This is due to the absence of a fine mesh that corrects the coarse domain with the collective effect of the fine scales, meaning the embedded domains would be subject to the incorrect environmental conditions. Therefore, the adoption of a different approach is followed.

In the last part of this chapter the theoretical background of the Block Spectral Method (BSM) is discussed. Firstly, the theoretical framework of the time-averaged flow decomposition is analysed. Then, the main BSM purpose and methodology are discussed, including the reformulation of the governing equations and the propagation of the flow corrections. Finally, the potential computational gain of the BSM is considered.

4.3.1 Time-Averaged Flow Decomposition

The pioneering work of Reynolds [77] is the basic foundation of the Block Spectral Method. In his work it was demonstrated that there is a distinctive value of a dimensionless number, defined as $Re = \frac{U_b \ell}{\nu}$, that determines whether flow disturbances cannot or can be maintained. In the case of flow unsteadiness, Reynolds showed that it is plausible to acquire a time-averaged flow field. However, to achieve this the closure issue needs to be tackled as the temporal averaging creates additional unknowns (Reynolds stresses), the terms consisting of the time-averaged products of the turbulent fluctuations $\tilde{u}_i$, expressed as:

$$R(\overline{\Phi}) + \overline{(\rho \tilde{u}_i) \tilde{u}_j} = 0 \quad (4.8)$$

In other words, a time-averaged solution of a transient case could be obtained by utilising a steady flow approach while correcting the residual vector of convective and diffusive terms R with the Reynolds stresses, if they have been calculated a priori. Again, turbulence modelling attempts to approximate them since their values are unknown.

Ning and He [78] showed that it is possible to generate a time-independent solution of a transient case by using Reynolds stresses. They focused on matching a time-independent solution for vortex-shedding using corresponding unsteady stresses against a time-averaged vortex-shedding solution. Finer turbulence fluctuations were taken into account by applying

the mixing-length turbulence modelling. This was demonstrated on a 2D cylinder case as well as on a 2D turbine blade trailing edge vortex shedding case.

The instantaneous conservative properties Φ can be broken down into a mean component $\bar{\Phi}$ and an unsteady component Φ' . In their work, the unsteady component consisted of the large scale turbulence, that is the vortex-shedding component $\tilde{\Phi}$, and the finer turbulent scales (handled by turbulence modelling) Φ^* . Thus, a flow variable can be expressed as:

$$\Phi = \bar{\Phi} + \Phi' = \bar{\Phi} + \tilde{\Phi} + \Phi^* \quad (4.9)$$

They obtained the vortex-shedding unsteady stresses by time-averaging the $(\rho \tilde{u}_i) \tilde{u}_j$ terms over a shedding period as:

$$\overline{(\rho \tilde{u}_i) \tilde{u}_j} = \frac{\sum_1^{N_m} (\rho u_i - \bar{\rho u_i})(u_j - \bar{u}_j)}{N_m} \quad (4.10)$$

$$\bar{\rho u_i} = \frac{\sum_1^{N_m} \rho u_i}{N_m} \quad (4.11)$$

$$\bar{u_i} = \frac{\sum_1^{N_m} u_i}{N_m} \quad (4.12)$$

where N_m is the number of measurements over a period. Firstly, a transient simulation was executed to acquire the unsteady stresses, followed by a steady simulation with updated governing equations that includes the pre-calculated Reynolds stresses.

The purpose of the aforementioned analysis is to come to the realisation that the time-mean flow equations are equivalent to the flow equations of the time-averaged flow variables corrected by some extra terms. The extra terms due to temporal averaging are now lumped together as adopted in [66, 72] and are called Unsteady Source Terms (UST). The time-averaged equations become:

$$\overline{R(\Phi)} = R(\bar{\Phi}) + \text{UST} = 0 \quad (4.13)$$

The lumped UST emerge due to the non-linearity of the governing equations. The non-linear behaviour can be attributed to self-excited structures, such as periodic vortex shedding and turbulence eddies. For obtaining a time-independent solution, the governing equations (Equation 4.2) is reformulated as:

$$\frac{\partial \Phi}{\partial \tau} + R(\Phi) = -UST \quad (4.14)$$

4.3.2 Correction Terms

In the previous chapter, both instantaneous and time-averaged flow fields around DREs were illustrated. Although the flow is highly unsteady at each moment, in the time-averaged regime it appears to be spatially periodic. Considering the analysis of the previous subsection, this time-averaged flow trait could be potentially exploited to accelerate numerical prediction. The Block Spectral Method is based on this idea.

The generic approach of the multi-scale method is outlined in the following example. Let us consider a simple two dimensional flat plate with roughness elements, similar to the analysis of the previous section.

1. Figure 4.11a: A coarse-mesh domain (C) that resolves bulk flow structures and a fine-mesh domain (F) that resolves near-wall flow structures are defined. It was mentioned in the previous section that due to the geometrical spatial periodicity of the regular roughness, it is possible to define a fine mesh that extends at a single wavelength. Then, it can be simply copied to the rest of the identical regions.
2. Figure 4.11b: These two problems are combined by embedding the fine mesh into the coarse mesh. It is chosen to refine specific areas to maintain the mesh size within acceptable limits.

3. Figure 4.11c: Therefore, there is a region of the coarse domain with corrected values notated as C_F . In the conventional dual-mesh approach the algorithm would be complete by this step. However, by looking at the second and third step it is realised that the near-wall region has not been wholly refined, as the middle DREs have not been assigned an embedded mesh.
4. Figure 4.11d: The aim is to correct the remaining unrefined near-wall regions, notated as C_C , in order to generate a time-averaged solution as if there was a fine mesh updating these regions. If whole near-wall resolution should be pursued, as a fully embedded configuration would require, the grid size would be prohibitively high, a constraint desired to be mitigated by utilising the BSM.

Caution is required when correcting the C_C regions, as the correction has to be applied in a meaningful way to generate the correct flow field. Earlier it was discussed how the UST can be utilised as source terms, providing the correct time-averaged flow field. In a similar way to turbulence modelling, time-independent correction sources can be introduced.

When applying these constructed source terms to the governing equations of the C_C region, there are two things that need to be considered. Firstly, the C_F flow properties, after they have been updated by the fine mesh, are not a solution of the coarse domain governing equations. Hence, the governing equations of the C_F regions need to be modified with additional instantaneous source terms that account for the spatial averaging, called instantaneous Spatial Resolution Source Term ($\text{SRST}'(t)$). The spatial-averaging introduces source terms similar to the UST of the time-averaging process, i.e. when the $\text{SRST}'(t)$ are available a priori, the volume-averaged solution can be obtained without the need to update the coarse domain with the fine domain properties. Secondly, any potential transient residuals at the C_C region have to be taken into account and be subtracted. The reason is the generated

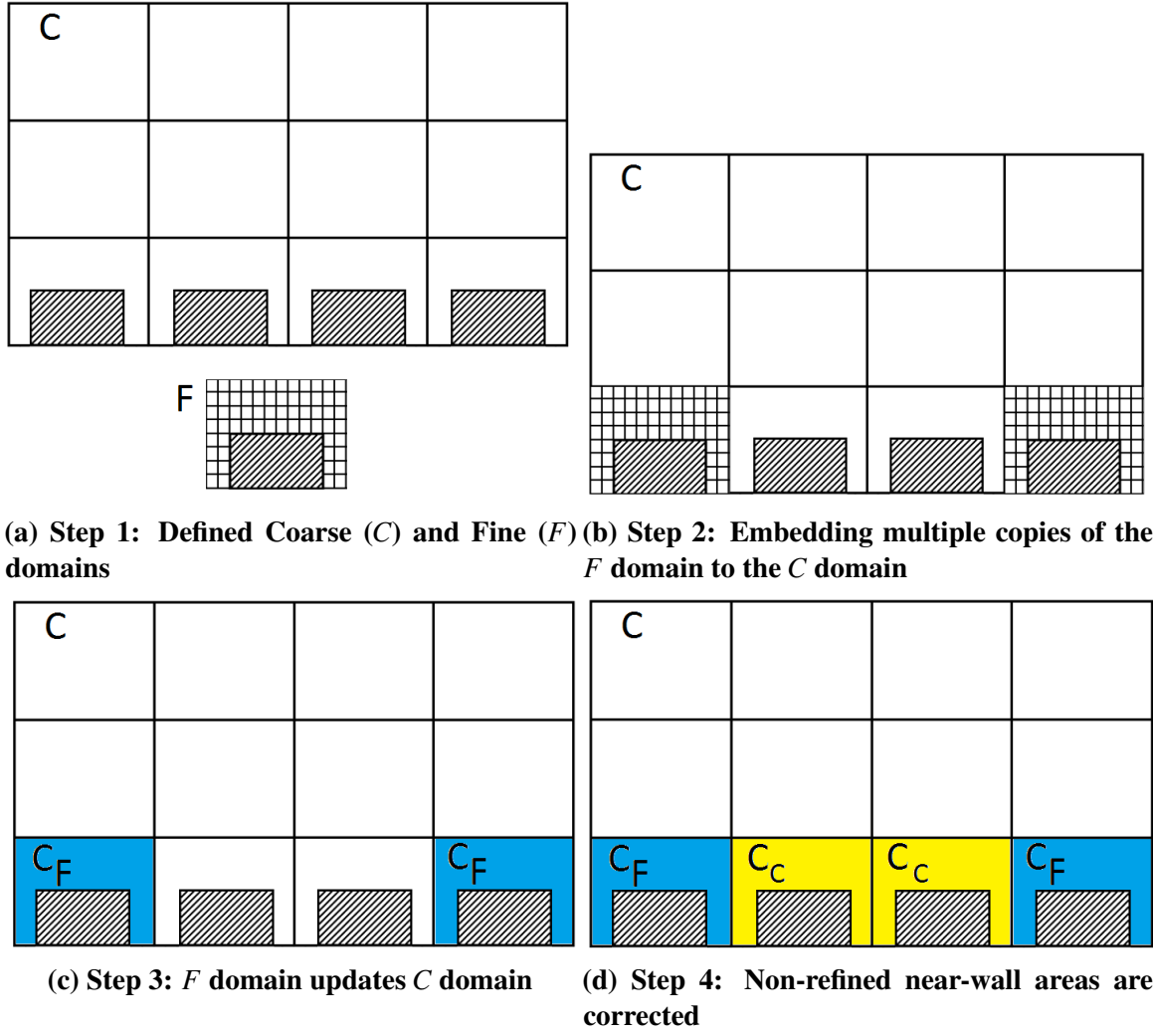


Figure 4.11 Outline of the proposed multi-scale approach

coarse mesh UST at the C_C regions would not correspond to a coarse mesh solution that is updated by a fine domain. Let us consider the governing equations of each region of the coarse and fine domain:

- Fine mesh (domain F):

$$\frac{\partial \Phi_F}{\partial t} + R_F(\Phi_F) = 0 \quad (4.15)$$

- Coarse mesh (domain C):

$$\frac{\partial \Phi_C}{\partial t} + R_C(\Phi_C) = 0 \quad (4.16)$$

- Coarse mesh that is updated by the fine mesh (domain C_F):

$$\frac{\partial \Phi_{C_F}}{\partial t} + R_C(\Phi_{C_F}) + \text{SRST}'(t) = 0 \quad (4.17)$$

- Coarse mesh that is updated by source terms (domain C_C). This equation refers to the time-independent corrections that need to be applied so that the targeted time-averaged flow field is generated:

$$\frac{\partial \Phi_{C_C}}{\partial t} + R_C(\Phi_{C_C}) + \text{corrections} = 0 \quad (4.18)$$

Time-averaging Equation 4.17 and taking into account Equation 4.13 will result into:

$$R_C(\overline{\Phi_{C_F}}) + \text{UST}_{C_F} + \text{SRST} = 0 \quad (4.19)$$

where SRST denotes the time-averaged $\text{SRST}'(t)$. The UST_{C_F} , SRST terms are lumped together in a single term, the Total Source Terms (TST) as:

$$\text{TST}_{C_F} = \text{UST}_{C_F} + \text{SRST} \quad (4.20)$$

Thus, for the base cells that are updated by the fine blocks (C_F region) the TST is calculated as:

$$\text{TST}_{C_F} = -R_C(\overline{\Phi_{C_F}}) \quad (4.21)$$

As an example, let us consider the approach described in the previous subsection. If the time-averaged flow field of a dual-mesh configuration is available, the dual-mesh time-averaged solution could be generated by only a coarse mesh domain, with the governing equations of the C_F regions balanced with the available TST_{C_F} .

By time-averaging equation 4.18:

$$R_C(\overline{\Phi_{C_C}}) + \text{UST}_{C_C} + \text{corrections} = 0 \quad (4.22)$$

The target is for the C_C region to provide a time-averaged flow field *as if* there were a fine mesh updating it. In other words, Equation 4.22 should converge to the target Equation 4.21, thus:

$$\text{corrections} = \text{TST}_{C_C} - \text{UST}_{C_C} \quad (4.23)$$

The TST of the C_C is of unknown values for now. From equation 4.13:

$$\text{UST}_{C_C} = \overline{R_C(\Phi_{C_C})} - R_C(\overline{\Phi_{C_C}}) \quad (4.24)$$

Therefore, the governing equations of the C_C regions are corrected as:

$$\frac{\partial \Phi_{C_C}}{\partial t} + R_C(\Phi_{C_C}) = -\text{TST}_{C_C} + \overline{R_C(\Phi_{C_C})} - R_C(\overline{\Phi_{C_C}}) \quad (4.25)$$

However, the known values of TST_{C_F} can be mapped, providing values for the TST_{C_C} . How this could be achieved is presented in the following subsection.

Considering the above, the purpose of the BSM is twofold:

- To avoid fully resolving all areas that demonstrate similar geometry and flow properties, yet accounting for their collective effect. For unsteady flow, such repetitiveness in the flow pattern is rare. However, such a pattern could appear in the time-averaged regime.
- To provide accurate boundary conditions to the embedded region. Considering a matrix of multiple micro-structures in a partially embedded configuration, the embedded regions will produce erroneous predictions since the boundary conditions of the embedded mesh at the base-embedded domains interface are incorrect.

Note that the additional calculations are applied to the coarse mesh domain, which is useful when the dual-mesh domain is distributed across multiple computational cores. On a final note, the solution of the unsteady BSM still converges to a steady flow field if the case configuration is of steady nature (e.g. steady laminar). In this scenario, the last two terms at the right hand side of Equation 4.25 cancel each other out as $UST_{C_C} = 0$.

4.3.3 Propagation of the Total Source Terms

The regions that demonstrate time-averaged flow similarity, like regular roughness, would be distributed on a surface. This means that the extrapolation of information from the C_F regions across the C_C regions would take place in a two dimensional way. For example, let us consider the top view of a surface with fitted DREs. The surface is characterised by square regions of similar time-averaged flow patterns, as pictured in Figure 4.12. α and β are the phase angles of each C_F and C_C regions in each phase direction. The aim is to build a representation using information from the C_F regions $((\alpha_1, \beta_1), (\alpha_1, \beta_5), (\alpha_1, \beta_9), (\alpha_4, \beta_1), (\alpha_4, \beta_5)$ and $(\alpha_4, \beta_9))$ and then having it extrapolated to the C_C regions.

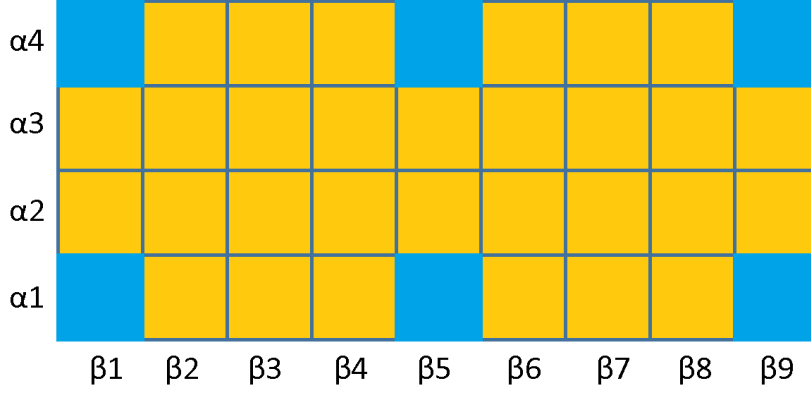


Figure 4.12 Schematic of BSM mapping. Source (C_F) and mapped (C_C) regions are denoted by the blue and yellow squares, respectively

The spectral extrapolation is performed by a pointwise region-to-region association for all the C_F and C_C regions. A general form of the spectral representation is shown below:

$$\text{TST}_{i,j,k}(\alpha_I, \beta_J) = \sum_{I=1, J=1}^{M_S, N_S} G_{i,j,k} \Psi(\alpha_I, \beta_J) \quad (4.26)$$

where Ψ , G are a basis function and its coefficients, respectively. M_S , N_S are the number of source regions in each phase direction. In the above example, that number would be two in the α direction and three in the β direction.

For example, let us consider the simple case of 1D mapping, as shown in Figure 4.13. The region-to-region variation, where the regions are indexed as I , is being established for each (i, j) point. α_1, α_2 and α_3 are the phase angles that correspond to the three shown spatially repeated regions, with α_1, α_3 referring to the C_F regions and α_2 denoting the C_C region. For each (i, j) mesh point the $G(i, j)$ values are calculated based on the basis function Ψ and the TST values of the C_F regions. With the constructed spectral representation (Equation 4.26) the TST at each (i, j) mesh point of the α_2 region is found. Again, it is of high importance that the C_F and C_C regions demonstrate similar geometry that leads to the same meshing topology, hence the identical grids. The computational cost of the calculation of the

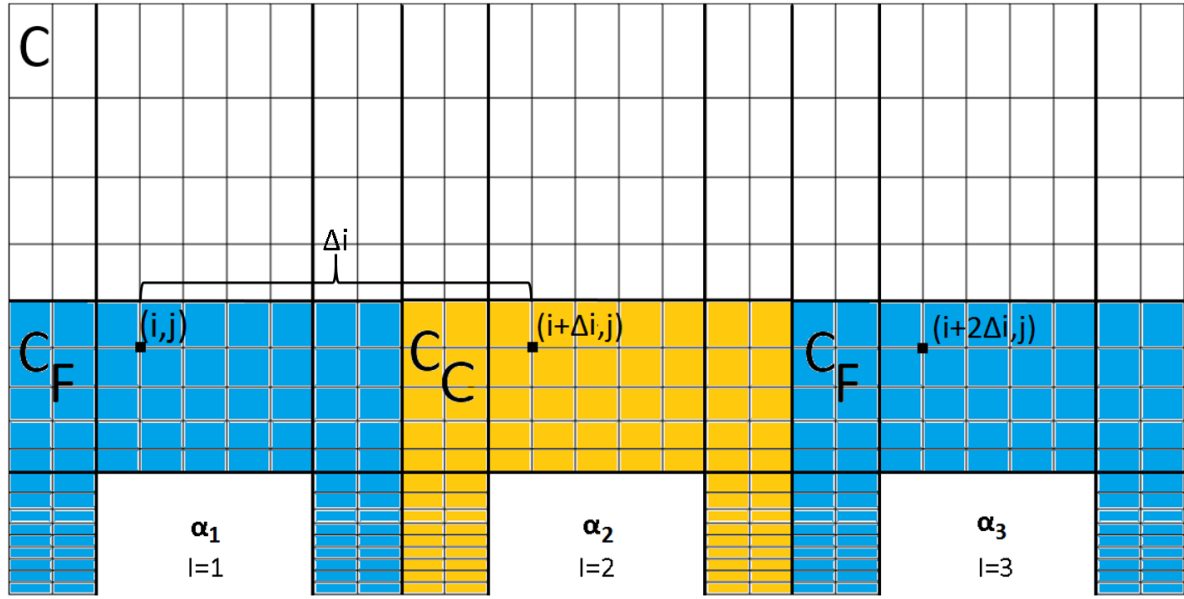


Figure 4.13 Example of 1D BS mapping

spectral representation must not outweigh the mesh size reduction achieved by negating full embedding.

At this point it should be clear how neatly this multi-scale method fits case studies of machined roughness elements. The same mesh topologies arising from the similar geometric features and the similar flow structures observed downstream of a few starting DREs are giving the opportunity to establish a region-to-region variation, as described above. It is clarified that the C_F regions do not necessarily need to include one roughness element each as shown in Figure 4.13, but also areas of interest around them, even several roughness elements per region without violating the concept of the BSM.

Care is required when choosing the regions that will be treated with the BSM. For the method to produce meaningful results, the TST_{C_C} need to acquire realistic values for these regions. A smooth variation of the time-averaged flow field requires a smoothly varying bulk flow. Hence, the BS regions cannot include flow areas with high flow gradients, such as shock waves (Figure 4.14a), areas of highly accelerating flow (Figure 4.14b), and areas of alternating flow directions (Figure 4.14c). However, the BSM can be applied in the surrounding areas.

This is implemented by applying a BS group either side of the high-gradient area but without crossing it, as shown in Figure 4.14, and without sharing any information between the BS groups.

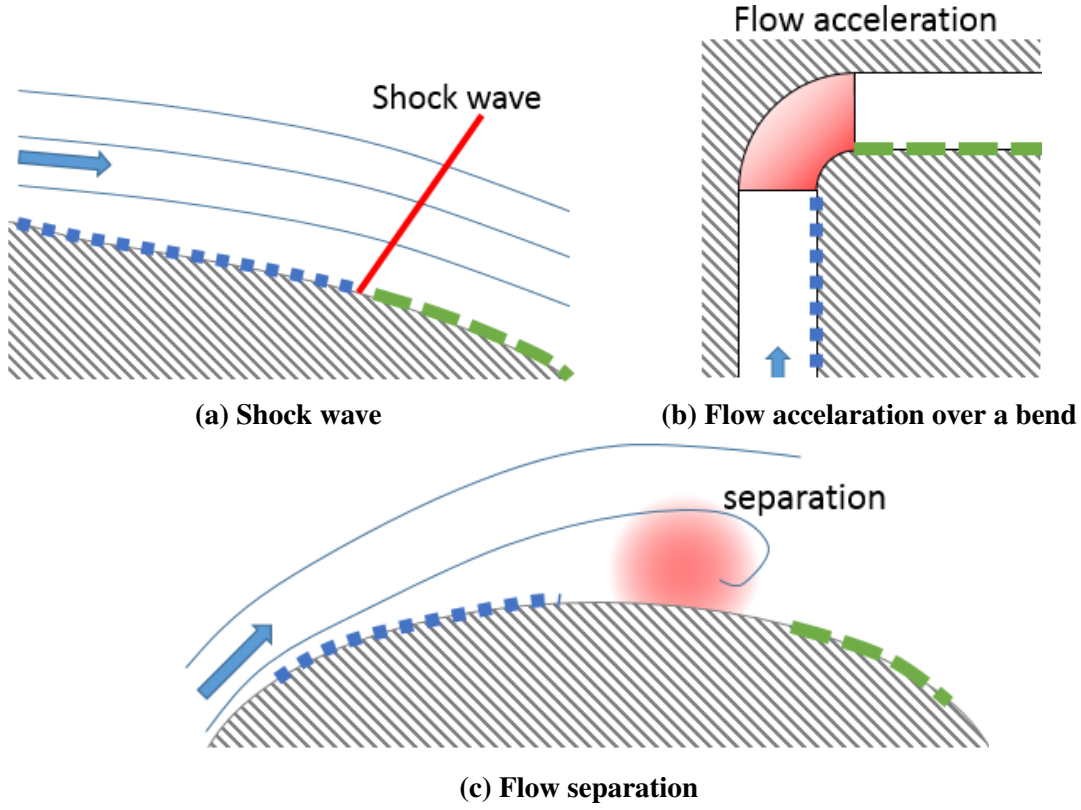


Figure 4.14 Examples of high gradient regions (red) that should not be considered for the BSM. Differently coloured/dashed lines (blue and green) denote regions where the BSM can be applied through separate spectra groups

A desired trait of the basis function is the fast convergence as the number of source regions increases. The Fourier series is a function of such behaviour, hence it has been selected as the basis function. Therefore, the spectral representation acquires the following form:

$$\text{TST}_{Ci,j,k}(\alpha_I, \beta_J) = \sum_{m=0, n=0}^{M_h, N_h} \lambda_{m,n} [A_{i,j,k}^{m,n} \cos(m\alpha_I) \cos(n\beta_J) + B_{i,j,k}^{m,n} \sin(m\alpha_I) \cos(n\beta_J) + C_{i,j,k}^{m,n} \cos(m\alpha_I) \sin(n\beta_J) + D_{i,j,k}^{m,n} \sin(m\alpha_I) \sin(n\beta_J)] \quad (4.27)$$

where

$$\lambda_{m,n} = \begin{cases} \frac{1}{4} & \text{if } m = n = 0 \\ \frac{1}{2} & \text{if } m > 0, n = 0 \text{ or } m = 0, n > 0 \\ 1 & \text{if } m, n > 0 \end{cases} \quad (4.28)$$

and M_h, N_h are the number of harmonics in each phase direction. The Fourier coefficients are calculated as the mathematical theory dictates:

$$\begin{aligned} A_{i,j,k}^{m,n} &= \sum_{I=1, J=1}^{M_S, N_S} \text{TST}_{C_F i,j,k}(\alpha_I, \beta_J) \cos(m\alpha_I) \cos(n\beta_J) \\ B_{i,j,k}^{m,n} &= \sum_{I=1, J=1}^{M_S, N_S} \text{TST}_{C_F i,j,k}(\alpha_I, \beta_J) \sin(m\alpha_I) \cos(n\beta_J) \\ C_{i,j,k}^{m,n} &= \sum_{I=1, J=1}^{M_S, N_S} \text{TST}_{C_F i,j,k}(\alpha_I, \beta_J) \cos(m\alpha_I) \sin(n\beta_J) \\ D_{i,j,k}^{m,n} &= \sum_{I=1, J=1}^{M_S, N_S} \text{TST}_{C_F i,j,k}(\alpha_I, \beta_J) \sin(m\alpha_I) \sin(n\beta_J) \end{aligned} \quad (4.29)$$

meaning that according to Equation 4.26 the following assignments are performed:

$$\Psi \rightarrow \cos/\sin(\alpha_I, \beta_J) \quad G_{i,j,k} \rightarrow \lambda_{m,n}(A, B, C, D)_{i,j,k}^{m,n}$$

The calculation of the Fourier spectral representation with $M_h \times N_h$ harmonics is subject to $4 \times (M_h + 1) \times (N_h + 1)$ unknown coefficients, but since $\sin(0) = 0$ the number is reduced to $(2M_h + 1) \times (2N_h + 1)$. The number of unknown coefficients determines the number of the C_F regions required as sources. In fact, an accurate solution can be achieved with the number of harmonics being of the order of 10 or even less [63, 64, 66]. This characteristic highlights the advantage of the Block Spectral Method when applied to large multi-scale

problems. In the case where a non-periodic distribution of flow properties is desired, the non-periodic distribution is considered as half of a periodic distribution with mirrored source regions. Examples of how this is configured will be presented in a future chapter.

Finally, the **Update flow properties** step is presented in Figure 4.15. For acquiring time-averaged results the marching time-averaging is adopted, as it is shown to be effective for flow disturbances (as shown by He [72]). The time-averaged flow variable at a physical time-step k can be calculated after sufficient $k - 1$ time-steps as:

$$\bar{\Phi}|_k = \frac{1}{k} \left(\sum_{\kappa=1}^k \Phi_{\kappa} \right) = \frac{1}{k} \left(\Phi_k + \sum_{\kappa=1}^{k-1} \Phi_{\kappa} \right) = \frac{1}{k} (\Phi_k + (k-1)\bar{\Phi}|_{k-1}) \quad (4.30)$$

where k is the current physical time-step.

4.3.4 Computational Gain

In general, the effectiveness of the solution of the Block Spectral Method over the direct, fully embedded solution where all micro-structures are refined can be roughly measured by a ratio of mesh-counts. The mesh-count ratio is in turn related to the ratio of the number of micro-structures retained in the fully embedded domain, M_{full} to that of the partially embedded solution domain M_{BS} . Alternatively, M_{full} reflects the number of micro-structures in the coarse/base domain, i.e. the number of micro-structures in the coarse domain will be the same as the number of micro-structures in the embedded domain for the fully embedded setup. Additionally, there is extra computational time required by the BS mapping, hence the C_{map} coefficient is introduced. It may be desired for the source regions to be surrounded by refined regions. N_{extra} is the extra number of C_F regions included to provide more realistic boundary conditions to the C_F regions from which the source terms are generated. Considering the

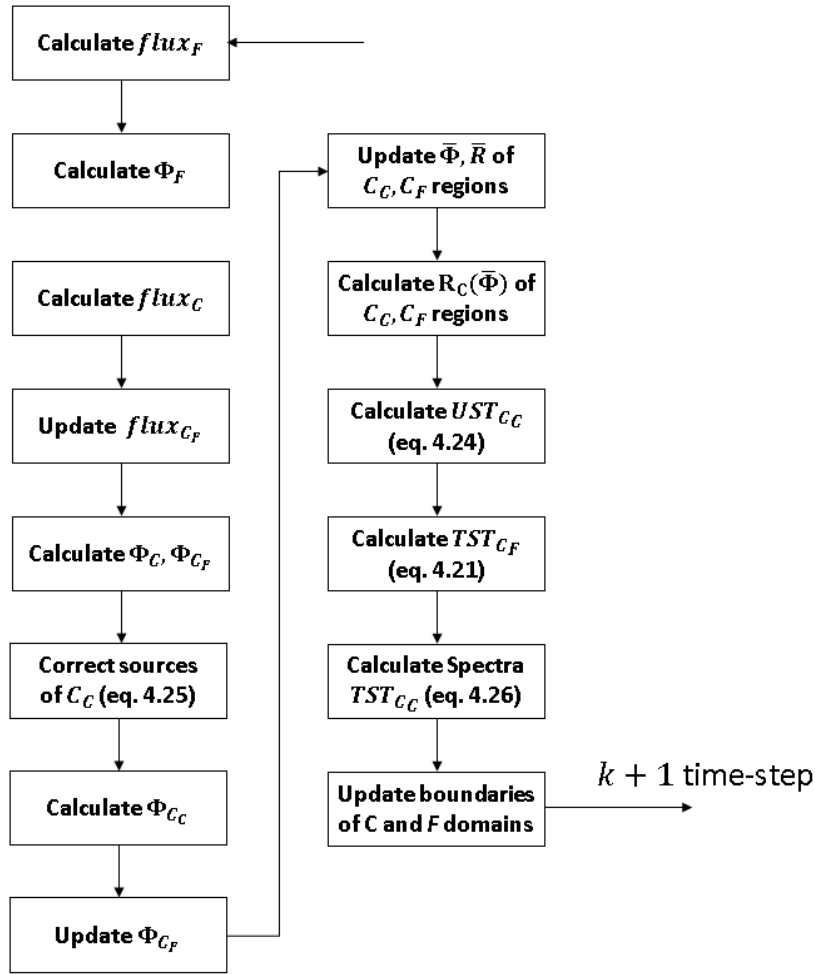


Figure 4.15 Updating flow properties: Block Spectral Method

above, the computational gain is defined as:

$$c_e = \frac{M_{full}}{C_{map} N_{extra} M_{BS}} \quad (4.31)$$

The sensitivity to the number of extra refined regions was examined in [66], and N_{extra} is estimated to be from 3 to 9 per source region. The number of the refined regions needed for generating the region-to-region spectrum, M_{BS} , is estimated to be 10-30 [63, 64, 66]. C_{map} will be quantified in a future chapter, where it will be shown to acquire sensible values (i.e. 1.1-1.2). The key point is that C_{map} , N_{extra} and M_{BS} would remain largely constant regardless of the size of the problem, (i.e. M_{full}) to be solved. This is the scalability effect of the BSM: the same number of micro-structures can be fully resolved and have their impact on the flow

mapped for an arbitrary number (M_{full}) of micro-structures. Thus, the effectiveness of the Block Spectral Method will be directly dictated by the number of micro-structures in the fully embedded domain to be solved, M_{full} . This is the primary appeal of the present multi-scale method in tackling potential problems that would be otherwise most likely intractable.

4.4 Parallelisation

The serial in-house solver was parallelised to allow for the simulation of large domains. The parallelisation was performed using blocking commands, and the parallelised version can accommodate domains with arbitrary mesh blocking. For dual-mesh configurations, only coarse mesh flow properties are required to be exchanged to perform the dual-mesh coupling, thus this property is exploited to prevent parallel efficiency from significantly dropping. Details on the parallelisation of the baseline solver are documented in Appendix C, along with a parallel efficiency study.

Also, in Appendix D the parallelisation of the BSM subroutine is analysed. The propagation of the source terms TST_{Cc} requires global information, due to the nature of the chosen basis function. Fortunately, the BS calculations are directly applied only to the coarse mesh domain. It is feasible for the coarse mesh domain, where the BSM is applied, to be handled by a single core. In this case, the BS solver in parallel demonstrates the parallel efficiency of the dual-mesh configuration. However, it is possible for multiple nodes to handle the coarse mesh; this scenario was investigated by transferring coarse mesh information to perform the propagation of the BS source terms through collective blocking commands.

4.5 Remarks

Initially, in this chapter the baseline finite-volume solver, its distinctive features and the backbone algorithm were described. The solver handles structured dual-mesh domains that

are stored in a compact manner to tackle the limitations of the used programming language. Also, two multi-scale methodologies were discussed, with the latter being an extension to the former.

Firstly, the embedded dual-mesh approach was described. Its benefits in easily generating a two-scale mesh were presented and the two-way coupling between the coarse and the fine domains were established. The spatial coupling is performed by a) volume-averaging of the fine mesh properties and then transferring the reduced properties to the coarse domain and b) by updating the fine domain boundaries through linear interpolation of the coarse domain properties. For transient simulations, the fine domain may undergo sub-time iterations for a single coarse domain time iteration.

Secondly, the Block Spectral Method for self-excited flows was discussed. It was argued that a partially embedded near-wall region would reduce the computational cost, allowing larger matrices of DREs to be modelled. However, partial embedding would impact the prediction accuracy due to the incorrect flow properties surrounding the fine domains. The Block Spectral Method tackles this issue, generating a time-averaged flow field equivalent to the fully embedded time-averaged solution. This is achieved by augmenting the governing equations with source terms that account for the fine domain impact on the coarse domain. The source terms for the unrefined near-wall regions are calculated through Fourier mapping of the source terms of the coarse domain regions overlapped by the fine domain. The low number of regions required to achieve accurate mapping allows for a desirable computational gain.

It is noted that the above analysis could be equally applied to an unstructured mesh. The requirement is for the coarse mesh topology to be spatially periodic, i.e. to demonstrate identical mesh topology among all near-wall regions for both the coarse and the fine domain.

Chapter 5

Validation of the Baseline Solver

This chapter is focused on the validation study of the multi-block and dual-mesh setups of the baseline solver. Firstly, the computational configuration is presented, followed by a study of multi-block cases to determine the accuracy of the shear stress and heat transfer predictions. Secondly, a study with a dual-mesh configuration is performed to ensure proper dual-mesh coupling. In the third and final section, a test case with available experimental data is simulated. The experimental configuration consists of a surface fitted with positive (pimples) and negative (dimples) micro-structures. For this study three turbulence modelling approaches are compared to experimental data and the underlying flow mechanisms are identified.

5.1 Computational Configuration

The computational domain of the majority of the case studies is illustrated in Figure 5.1 and the boundary conditions are set as follows: stagnation pressure P_0 and temperature T_0 at inlet (1), static pressure P at the exit (2). Inviscid wall conditions are applied at the two sides and the top boundary (4, 5 and 6). Finally, the bottom boundary (3) is a viscous wall

(with or without micro-structures). When heat transfer results are obtained, the viscous walls are set to be isothermal. For any domain that does not follow this setup, its configuration is explicitly described. Similarly, no turbulence modelling is applied for the following simulations, unless explicitly mentioned. ILES is utilised for all transient simulations with the explicit time-marching, satisfying the numerical stability criterion ($CFL=0.9$, with the speed of sound in the bulk flow as the reference velocity).

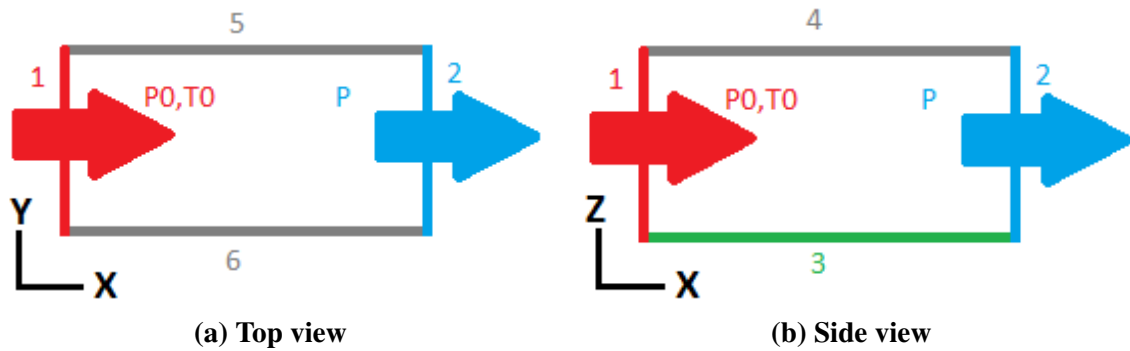


Figure 5.1 Computational domain

5.2 Multi-Block Test Cases

Initially, the ability of the present solver to accurately predict the boundary layer growth on a flat plate and around a cylinder is studied. The test cases are carried out assuming a laminar flow regime and using FLUENT 14.5 as the benchmark commercial software of choice. Mesh dependencies are studied and the presented results are mesh independent. The domain is distributed among multiple cores, when applicable, and various core distribution configurations of the domain are tested to ascertain the consistency of the results and thus proper core communication.

The first case is a 2D flat plate. The domain consists of a viscous wall with isothermal conditions and is preceded by an inviscid wall. The length of the viscous wall spans 80% of the length of the domain L . The height of the domain is $50 \times \delta_L$, where δ_L is the boundary

layer thickness at $x = L_{\text{viscous}}$. The flow is in the laminar regime with $Re_{L_{\text{viscous}}} = 13300$ and $Pr \approx 0.71$. Both velocity profiles (Figure 5.2a) and local heat transfer (Figure 5.2b) agree with the analytical and experimental results, respectively. For the heat transfer prediction, the relative error at the exit plane is less than 2%.

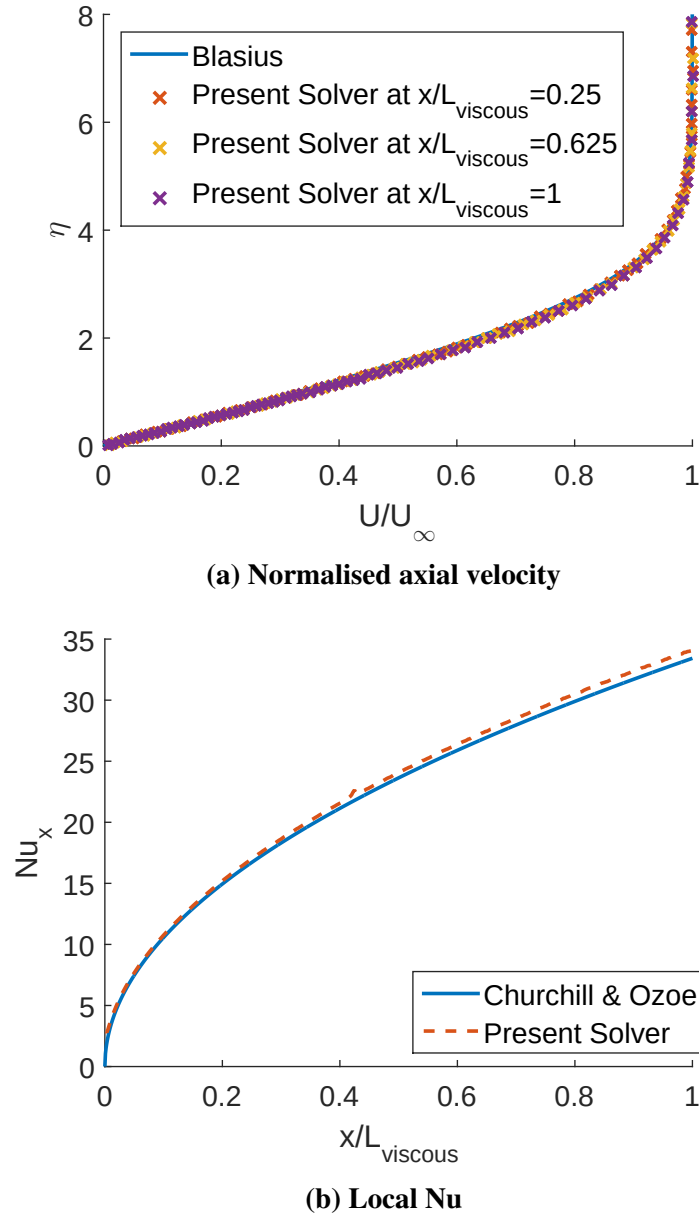


Figure 5.2 Flat plate: Normalised axial velocity and local Nu profiles compared to analytical and experimental data [79], respectively

The second case is a 2D cylinder in isolation ($Ma_{in} = 0.2$). An O-grid is fitted around the cylinder as shown in Figure 5.3. The span of the domain is equal to $6 \times D$, where D

is the diameter of the cylinder. Symmetry conditions are applied at the side boundaries. The heat transfer predictions for $Pr = 0.71$, shown in Figure 5.4, show good agreement with semi-analytical data and commercial software results. For $Re_D = 40$ the relative error e_a at the upstream stagnation point ($\theta = 0^\circ$) is equal to 2.4% and its maximum value of $\approx 6\%$ is located at $\theta \approx 140^\circ$, within the separation region. For $Re_D = 94$ the e_a at the upstream ($\theta = 0^\circ$) and downstream ($\theta = 180^\circ$) stagnation points are equal to 2.8% and 18%, respectively. Still, the large relative error in the separation region does not compromise the agreement of the predicted trends.

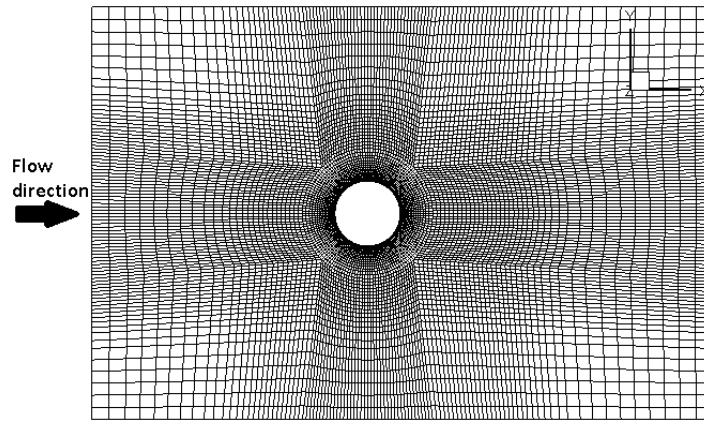


Figure 5.3 2D cylinder: Mesh

5.3 Dual-Mesh Test Case

In this section the dual-mesh configuration is compared to a multi-block configuration of equivalent mesh density. The case deals with four 3D cylindrical pedestals, configured as shown in Figure 5.5. Figures 5.6 and 5.7 show the multi-block configuration (a) which consists of O-grids around the cylindrical pedestals, contrary to the Cartesian mesh surrounding the pedestals in the dual-mesh (b) which consists of multiple embedded mesh blocks. For the dual-mesh setup, the mesh is deformed to outline the cylindrical solid domain. The coarse mesh and the levels of refinement by the embedded mesh are selected so that the

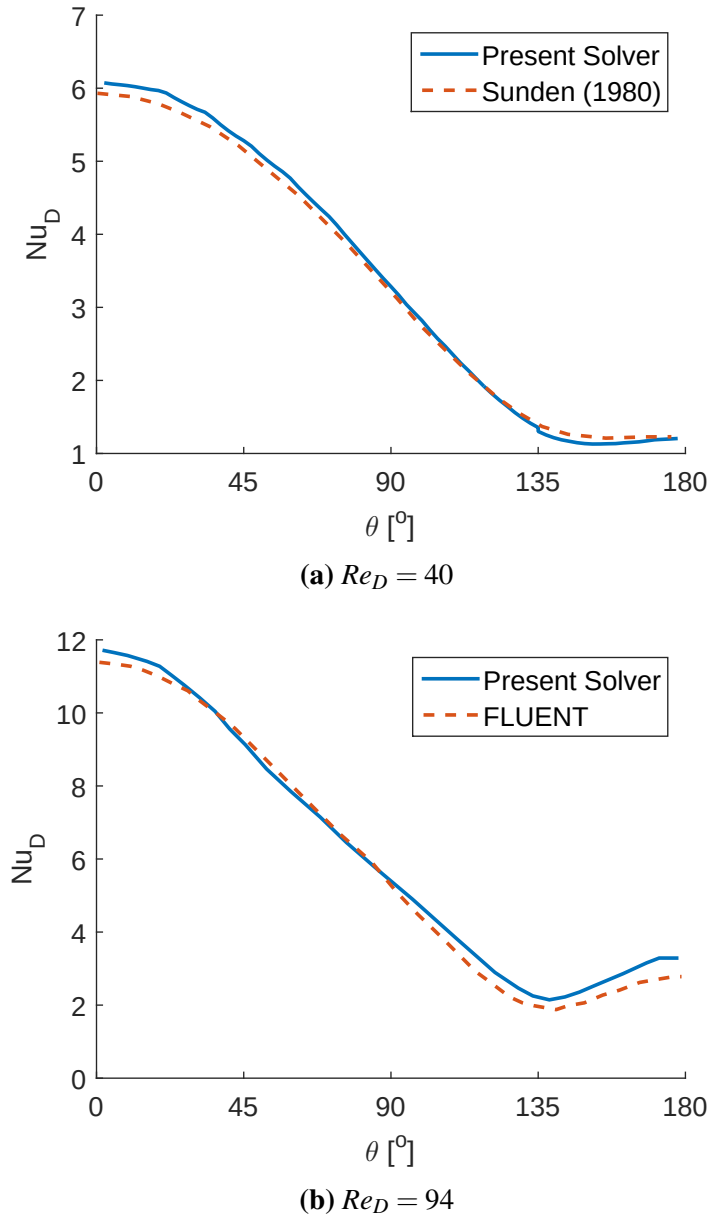


Figure 5.4 2D cylinder: Nu distribution around cylinder in isolation compared to semi-analytical data [80] (a) and commercial software (b). 0° and 180° denote upstream and downstream stagnation points, respectively

embedded mesh density is similar to the mesh density of the multi-block configuration. The embedded domain consists of six embedded blocks refining the base mesh 11 times in each direction. In Figure 5.8 the stagnation pressure distribution of the fine mesh is shown to be in good agreement with the multi-block solution. The location of the presented results is at the interface of the fine blocks, where the coupling effect would be most prominent. The

maximum relative errors in Figure 5.8 appear at $\text{span} = \pm D$ (wake regions) and $\text{span} = 0$ (region between the wakes). Still, the relative error in both areas is significantly less than 1%.

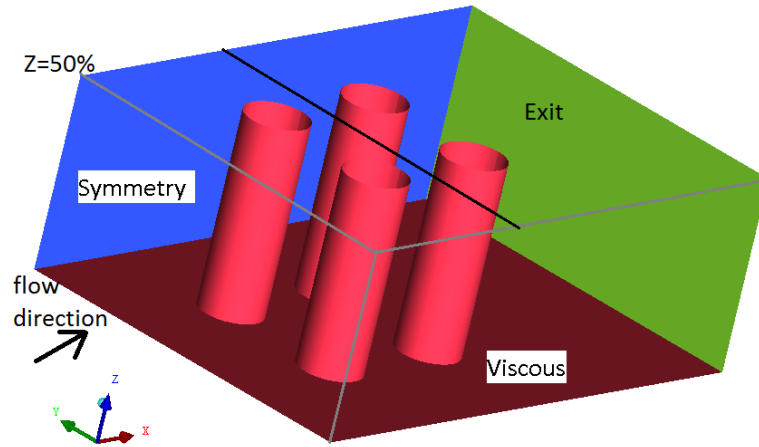


Figure 5.5 3D Pedestals: Cross section at a symmetry plane. Black line indicates plotting line in Figure 5.8 and it is located between the pedestals, equidistantly from the upstream and downstream pedestals

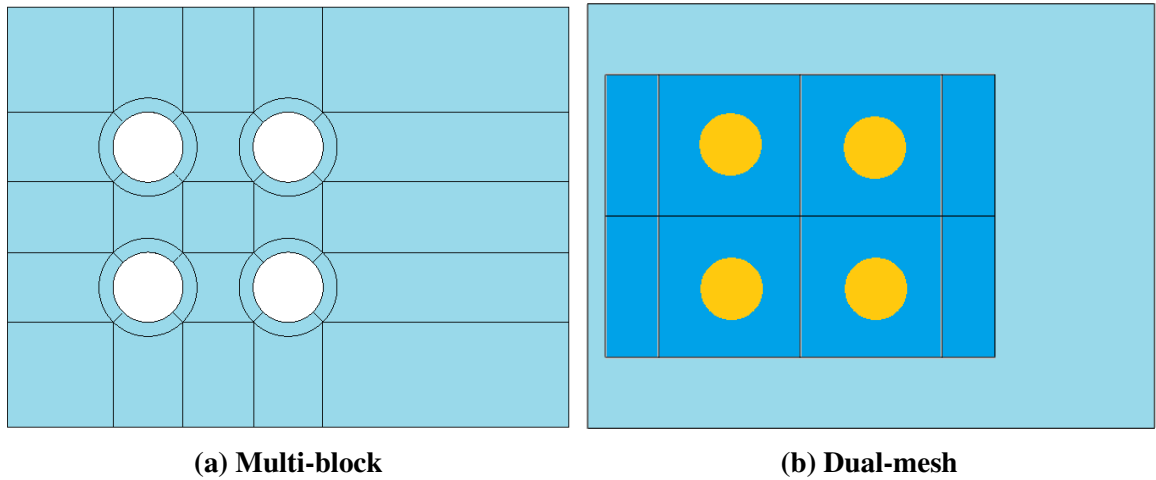


Figure 5.6 3D Pedestals: Top view of mesh blocks. Light-blue coloured regions denote the base mesh, dark-blue coloured regions are the embedded blocks. Yellow cylinders indicate embedded solid domains

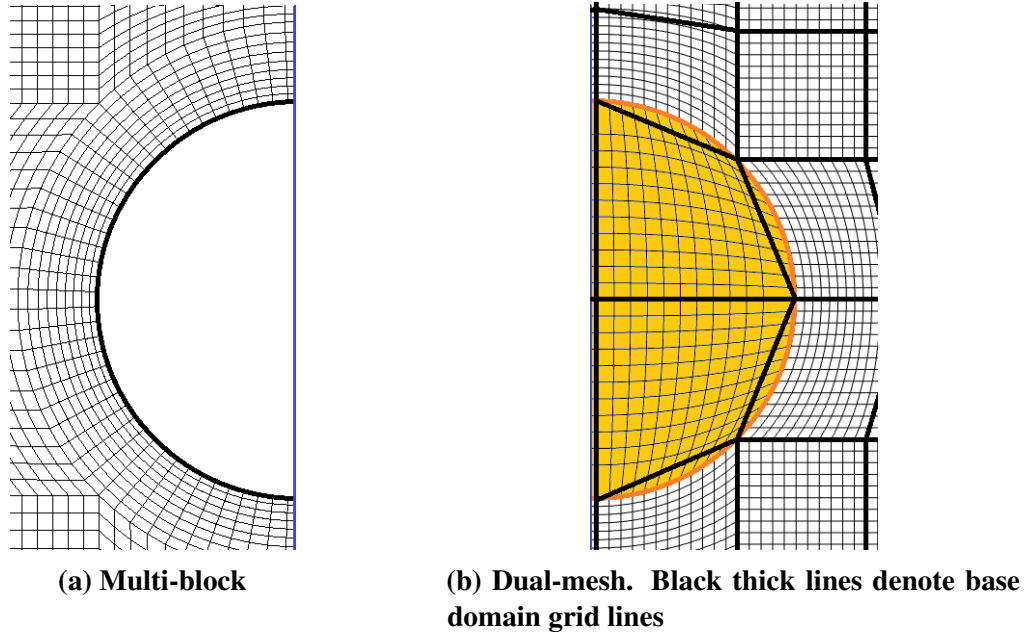


Figure 5.7 3D pedestals: Mesh close-up around the pedestals. Yellow regions denote embedded solid domains

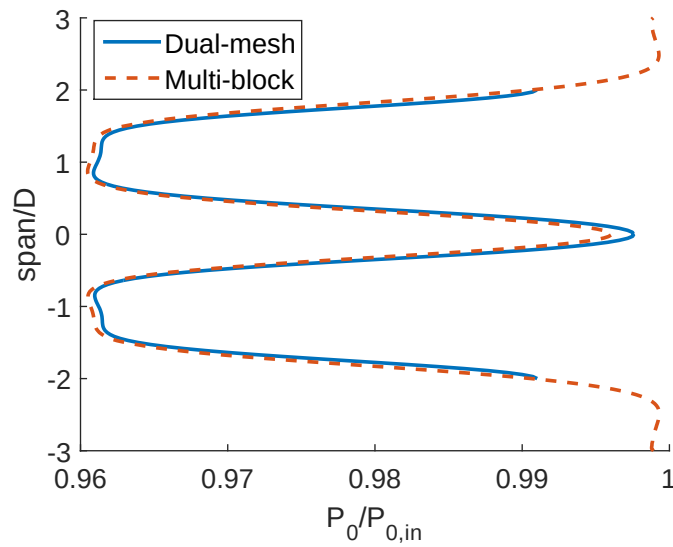


Figure 5.8 3D Pedestals: Comparison of stagnation pressure loss along the black line as indicated in Figure 5.5

5.4 Micro-Structures in Channel Flow

In this section, a further validation study of the present solver is carried out for a square channel flow with bisected hemispherical positive (pimples) and negative (dimples) micro-structures in an alternating pattern. Firstly, the experimental configuration is discussed,

followed by the computational configuration. The final part of the section consists of comparisons between the experimental and numerical results for three approaches of turbulence modelling, closing with an investigation of the near-wall flow features.

5.4.1 Experimental Configuration

The geometry and experimental results are obtained from the University of Central Florida/Siemens project team [81, 82]. The tested $Re_{D_h} = U_b D_h / \nu$ ranged from $\approx 150,000$ to $\approx 260,000$, where U_b and $D_h = 0.0508m$ are the bulk velocity and the hydraulic diameter of the channel, respectively. The examined flow is subsonic, with maximum $Ma_b \approx 0.27$ at $Re_{D_h} = 260,000$.

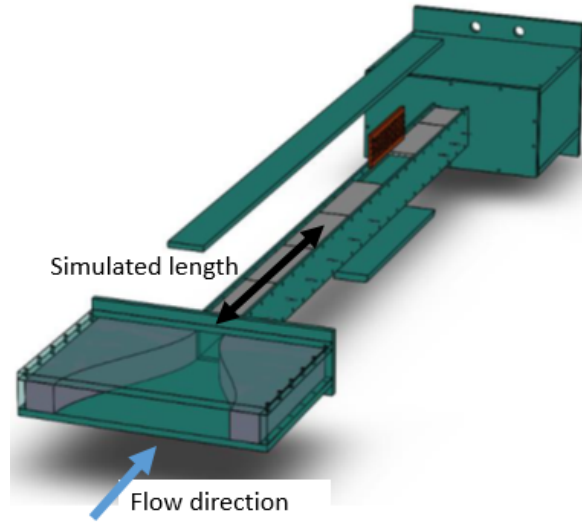
For the experimental studies two rig variations are employed. For the pressure drop along the flow direction, the experimental rig, illustrated in Figure 5.9a, is $\approx 22D_h$ in length. For the heat transfer measurements, the length is extended and fitted with smooth walls in the upstream direction, as shown in Figure 5.9b. Additionally, heaters are installed around the wall of the test section. The geometric properties are summed up in Table 5.1.

Table 5.1 Rig geometric configuration

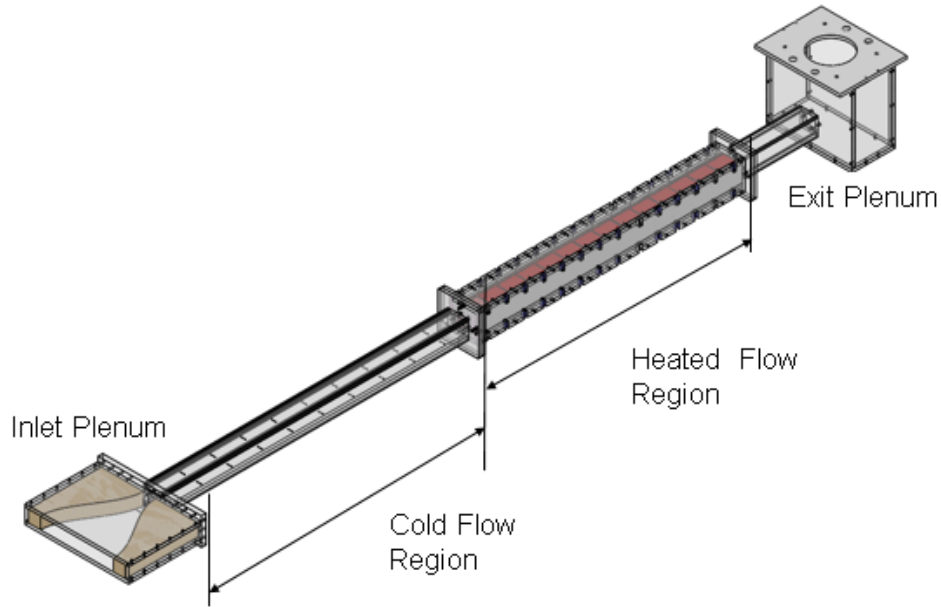
Cold length	Heated length	Cross-sectional area	Contraction ratio
$25D_h$	$\approx 21.9D_h$	$D_h \times D_h$	6.0

There are 14 axially distributed heaters per wall, and the conditions on the heated walls are found to be isothermal ($333 \pm 0.4K$). Since the measured power per heater is available, the area-averaged Nu is calculated as:

$$Nu = \frac{qD_h}{(T_{in} - T_{wall})k_{cond}} \quad (5.1)$$



(a) Loss measurements [82]



(b) Heat transfer measurements [81]

Figure 5.9 Rig configurations

where k_{cond} is the thermal conductivity of the air at the inlet and q is the area-averaged heat flux calculated as:

$$q = \frac{\text{power per heater}}{\text{projected area of heater on the plane of the wall}} \quad (5.2)$$

The micro-structures, which are 3D printed in 0.1mm resolution, are distributed in multiple rows of 7 and a pair of a pimple and a dimple is repeated periodically in both directions every

$\lambda = 0.246D_h$, with the diameter of the pimples being larger ($D_p > D_d$). The surface fitted with pimples and dimples is illustrated in Figure 5.10. The radius at the plane of bisection r_f and the height (depth) h of the micro-features are presented in Table 5.2.

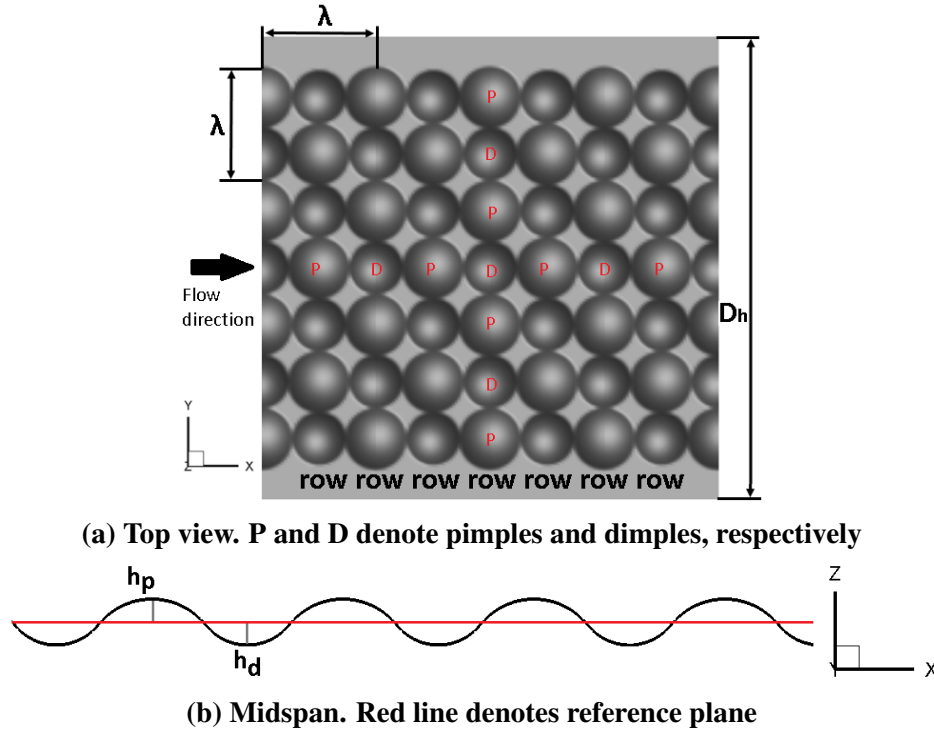


Figure 5.10 Surface fitted with micro-structures

Table 5.2 Properties of the micro-structures

	Pimple	Dimple
r_f/D_h	0.066	0.057
h/D_h	0.030	-0.030

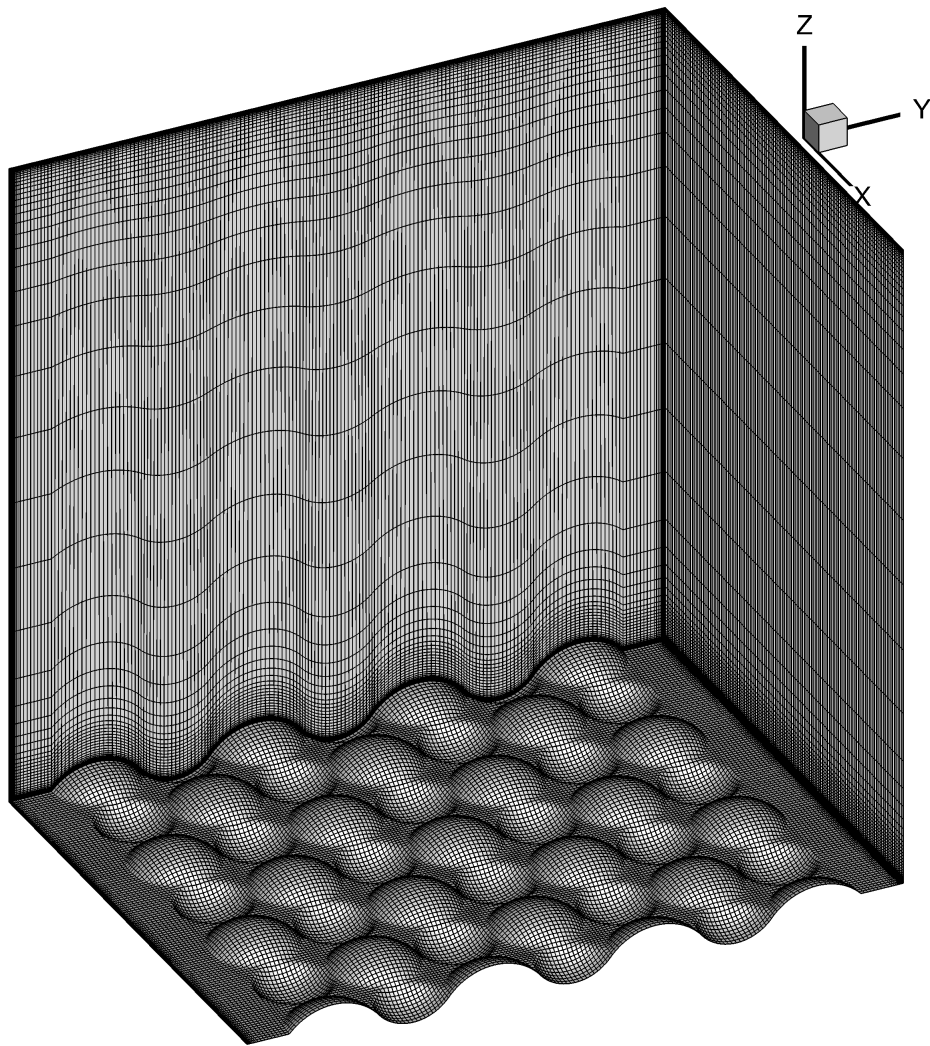
5.4.2 Computational Configuration

In Figure 5.11a an overview of the nominal multi-block mesh is presented. Each mesh block, as indicated by the highlighted red lines in Figure 5.11b, consists of $44 \times 210 \times 72$ mesh points in the X, Y and Z direction, respectively, with the surface fitted with the micro-structures located at the bottom. Due to the structured nature of the mesh the size of the domain increases significantly as the length of the simulated channel increases. Hence, the

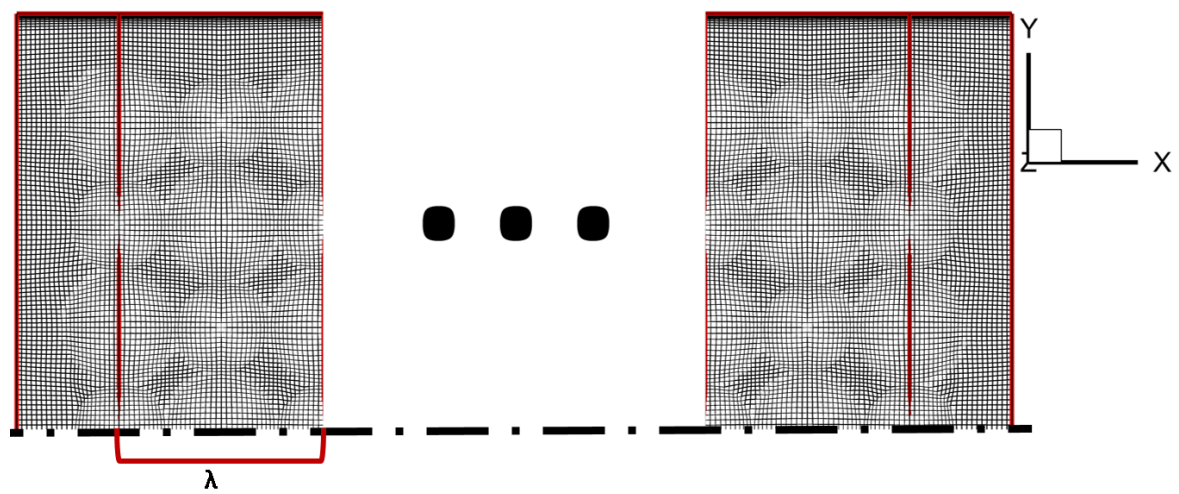
length of the computational domain considered here is 40% of the length of the experimental domain shown in Figure 5.9a. For the simulations with the nominal mesh, 77 rows of micro-features are considered, reaching a total mesh size of $\approx 28M$ cells. All the presented results are for $Re_{D_h} = 230,000$. From the geometric properties of the micro-structures it is derived that $Re_{h_p} \approx 6900$, thus the flow near the surface fitted with micro-structures is anticipated to transition into unsteadiness.

The boundary conditions are as previously outlined in section 7.1. Here, no slip and isothermal conditions are applied to all four walls. For setting up the inlet and outlet boundary conditions, data generated by the experiments in [81] are utilised. Atmospheric conditions are taken as inlet stagnation properties. The static pressure at the exit is derived by interpolating the experimental data points of [81]. Regarding the inlet stagnation pressure profile, there is a lack of information about the inlet profiles of the test section. Hence, the simplest applicable distribution is assumed, that of a uniform profile.

For treating turbulence, employing RANS is the initial approach. In this case, a fully turbulent regime is assumed, adopting the one-equation SA model with inlet $Tu_{in} = 4\%$ (based on the PIV measurements in the bulk flow). However, in order to accurately capture the wake interactions a substantial portion of the turbulence spectrum should be directly resolved. Hence, the primary analysis is performed for the implicit LES results. The coarse mesh ILES introduces a laminar boundary layer at the flat surfaces. To resolve this, a hybrid ILES configuration ($Tu_{in} = 0\%$) is also performed where the top wall (opposite of the surface fitted with micro-structures) is tripped near the inlet according to the SA tripping function [74]. This modification of the SA model introduces a trip term in the standard SA transport equation. Since the Tu at the inlet is zero, the SA turbulence field variable $\tilde{v}$ is zero as well. This term is a function of the distance from the wall, the vorticity at the wall, and the



(a) 3D view of the mesh distribution on surface of interest, side wall (X-Z plane) and bulk domain (Y-Z plane)



(b) Top view of surface of interest. Red lines denote the mesh block edges

Figure 5.11 Nominal mesh: First cell height is $4.9 \times 10^{-5} D_h$

difference of the velocities at the cell centres between the local cell and the cell adjacent to the wall. To apply the trip term, a region adjacent to the viscous wall is chosen near the inlet. In this way, the source term of the $\tilde{\nu}$ transport equation acquires a non-zero value at the tripping region, as well as downstream as the $\tilde{\nu}$ is convected through the transport equation. The side walls are not tripped in order to avoid contaminating the near-wall region of the regular roughness with turbulent viscosity. Due to the stratified nature of the flow at the top flat surface, the $\tilde{\nu}$ does not diffuse far from the top surface towards the bulk region for the given domain length. Thus, no blending function is utilised, as the SA transport equation is passive at the rest of the domain.

For the mesh dependency study an even more truncated domain, $1.23D_h$ long, is utilised. The plotting lines are illustrated in Figure 5.12. The metrics chosen for this study are:

- The time-averaged axial velocity (Figure 5.13a) along a Z-direction plotting line perpendicular to the second last row of micro-structures in the simulation.
- The time-averaged local Nu (Figure 5.13b) in the Y-direction across the central line of the second last row of micro-structures

The study is performed only for the ILES, as this configuration is highly sensitive to mesh refinement. Both metrics are mesh independent for the nominal mesh size. The results indicate that the utilised nominal mesh size is adequate and produces reasonably mesh-independent solutions for the coarse mesh ILES, as reported in Table 5.3. In addition, interesting trends for the heat transfer at the centre of the micro-structures are shown. These trends will be further discussed after the identification of the near-wall flow mechanisms.

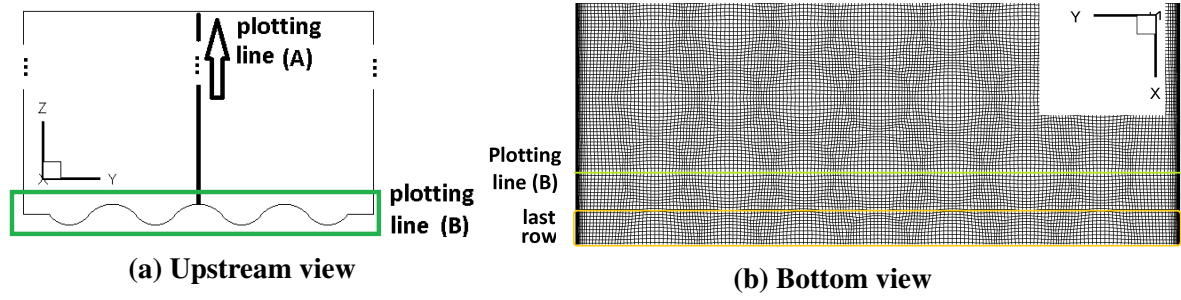


Figure 5.12 Plotting lines investigated in Figure 5.13

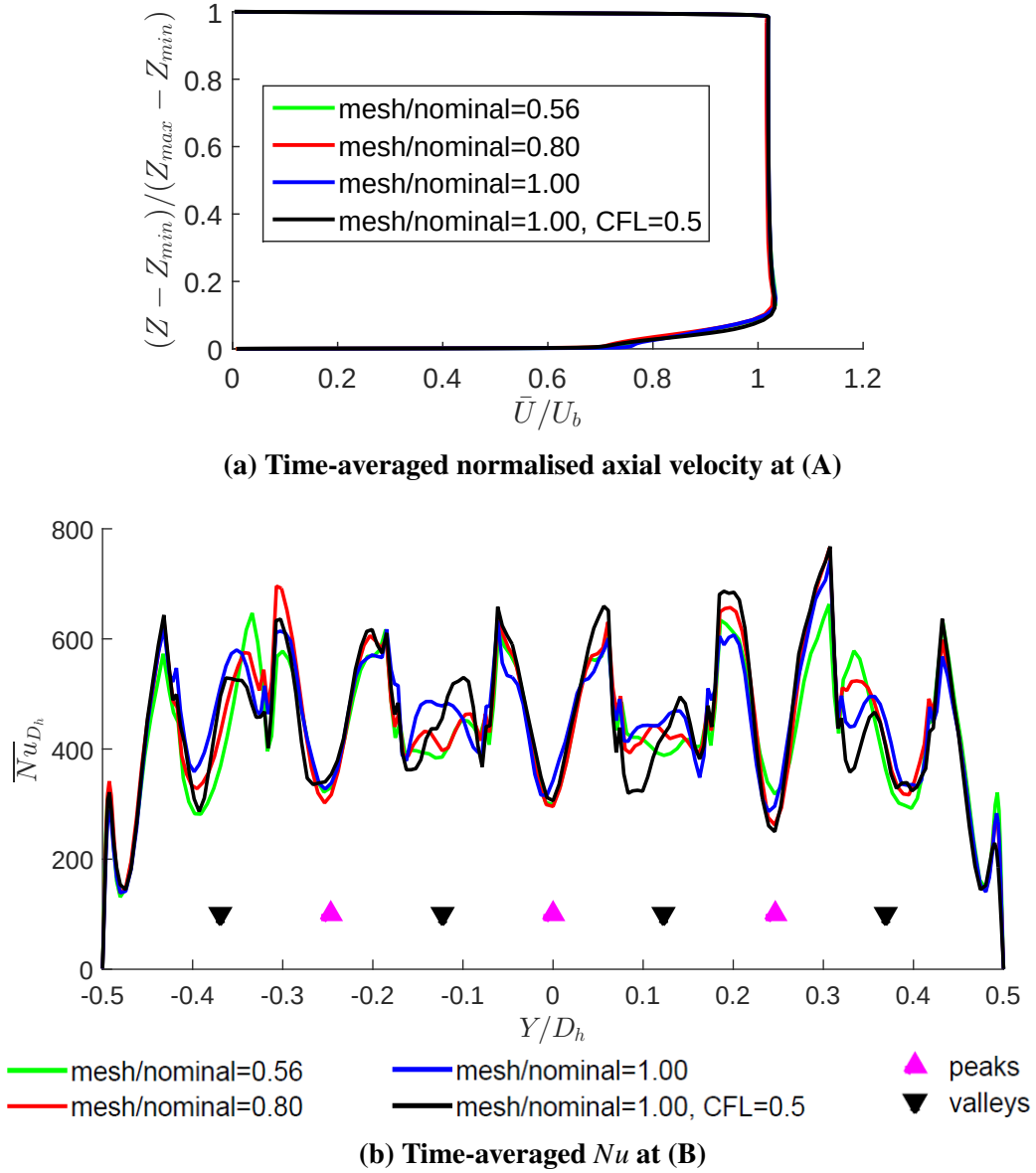


Figure 5.13 Mesh dependency study. Plotting lines are illustrated in Figure 5.12

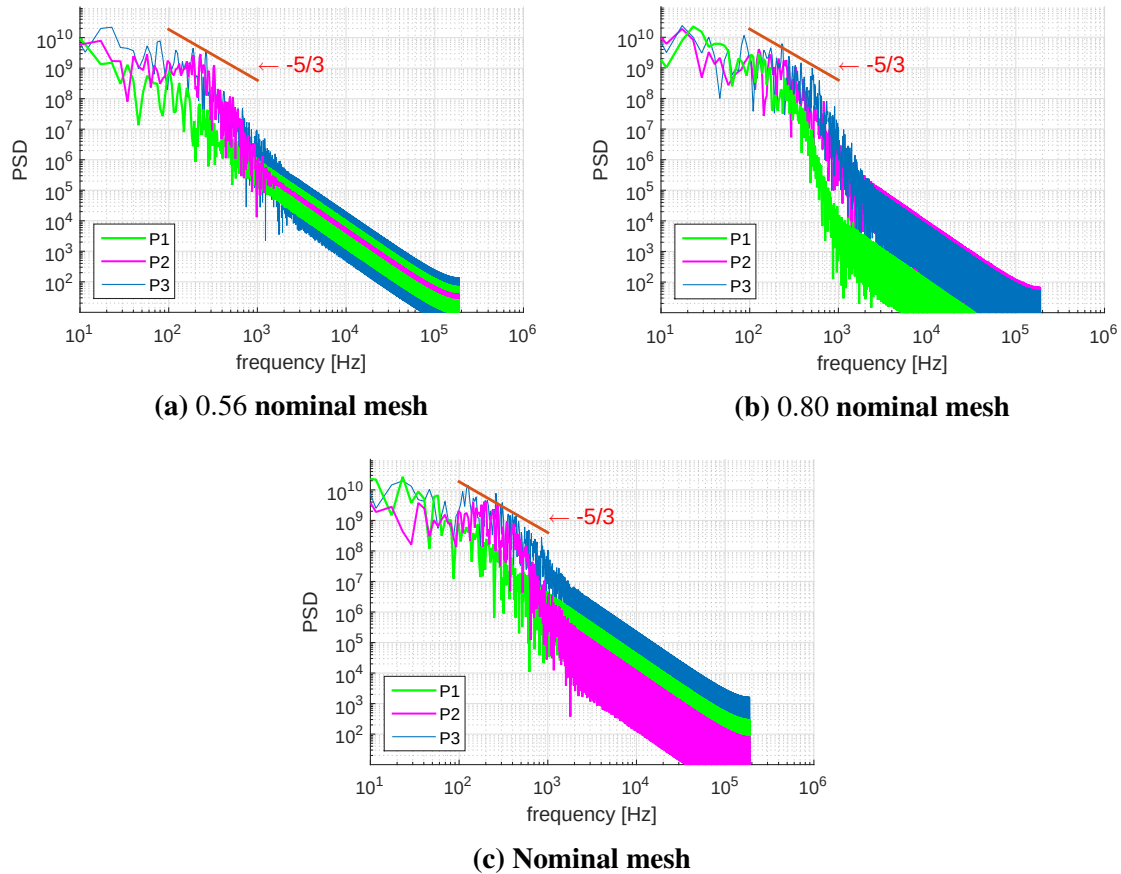
Table 5.3 Relative error e_a of the area-averaged $\overline{Nu}$

mesh/nominal	0.56	0.80	1	1, $CFL = 0.5$
$e_a[\%]$	3.6	0.4	0.0	1.2

Additionally, in Figure 5.14 the power spectra downstream of a pimple are presented. The Z positions of the plotting points are listed in Table 5.4. The captured inertial turbulence scales are located between $100Hz - 800Hz$ in the frequency regime.

Table 5.4 Z coordinate of the plotting points of Figure 5.14. The plane of the smooth surface is the reference plane

point	P1	P2	P3
Z/D_h	-0.028	-0.015	0.033

**Figure 5.14** Power spectra density of axial velocity in the centre of the last central dimple

5.4.3 Results

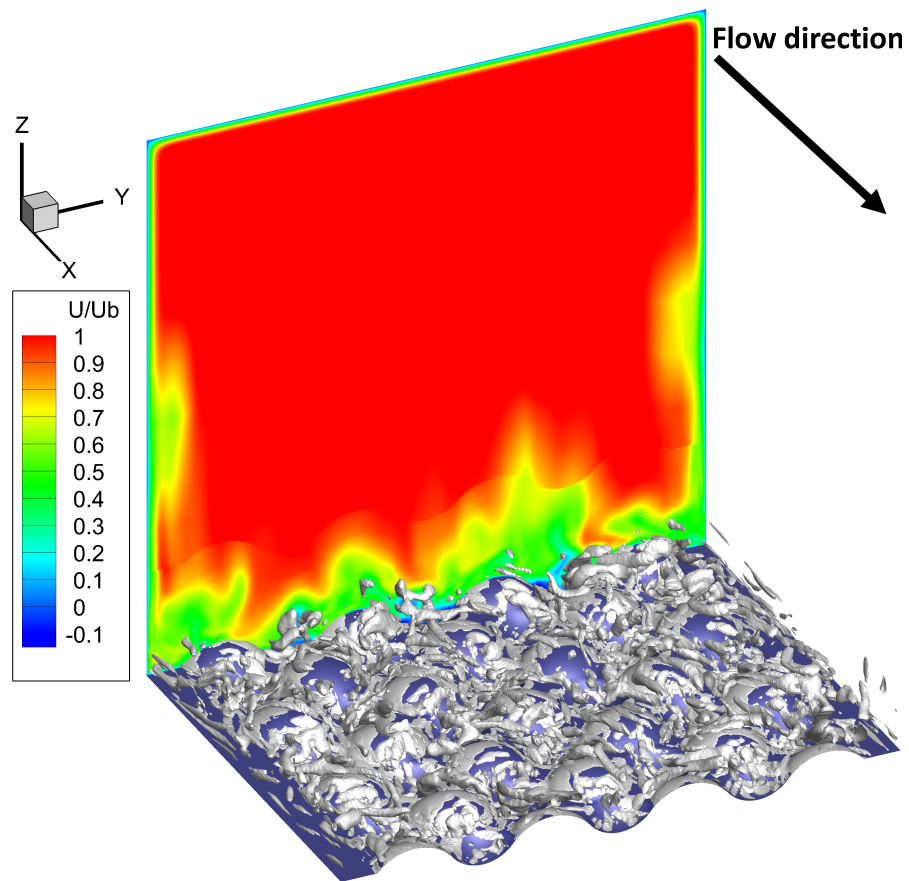
Outline of Predicted Flow Fields

Firstly, the flow characteristics resolved by each turbulence modelling approach are compared. This discussion is based on Figures 5.15, 5.16 and 5.17. For these figures the following configuration is applied: the range of the presented pimple-dimple rows is from row 71 to row 77, located near the exit plane. The illustrated cross-sectional plane is located in the centre of row 71 ($0.861D_h$ upstream of the exit plane). For visualising the iso-surfaces considering the Q criterion, the selected magnitude for the transient results of the ILES and hybrid ILES is significantly higher than the selected magnitudes for the SA results and the time-averaged results of the ILES and hybrid ILES.

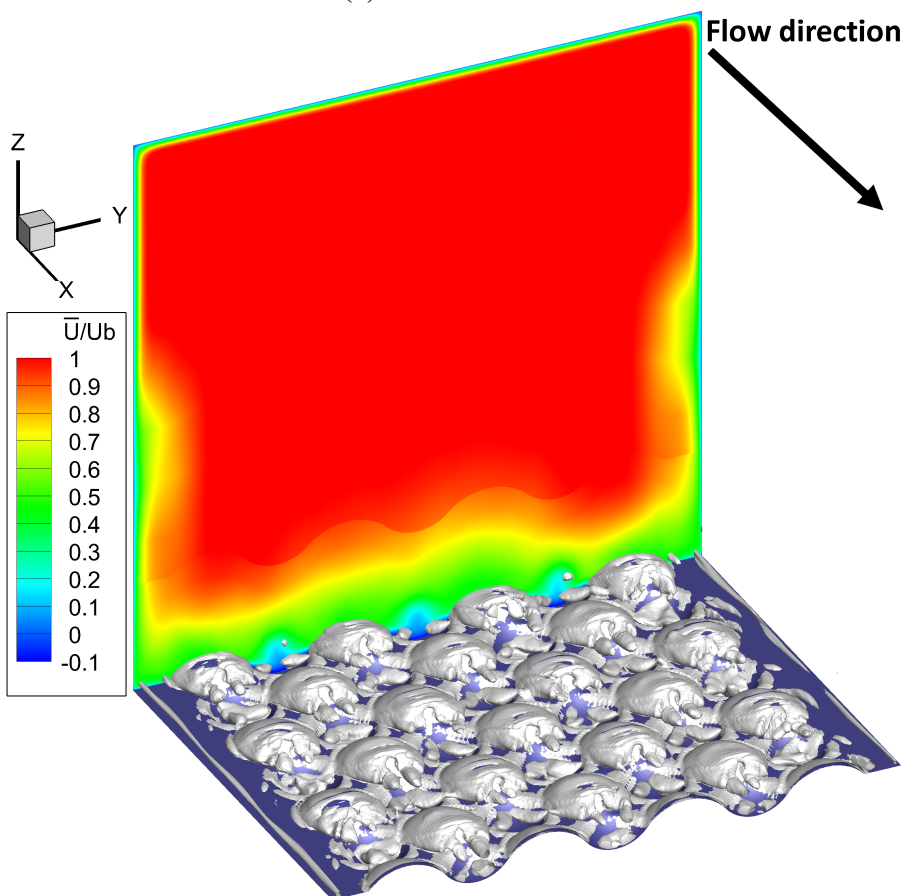
As expected, the flow around the surface fitted with micro-structures is highly unsteady (Figures 5.15a and 5.17a). The upstream laminar flow rapidly transitions into an unsteady flow in the presence of the micro-structures. However, this is not the case when the SA turbulence modelling is applied. The generated flow field is largely steady, as the turbulent viscosity dissipates most of the flow perturbations in the near-wall region (Figure 5.16).

The boundary layer of the top wall remains steady for all the turbulence models considered. For the ILES, the mesh is not sufficiently fine to allow the flow over the flat surface to transition. Introducing a fully turbulent regime or a trip function causes the thickness of the boundary layer to increase (Figures 5.16a and 5.17a). In the case of the hybrid ILES, the propagation of the turbulent viscosity in the vicinity of the top flat wall is shown in Figure 5.18.

Also, there are clear secondary flow patterns near the side walls (Figures 5.15a and 5.17a) where axial velocity deficits are observed at the sides of the domain. These deficits are generated by the interaction between the wakes shed by the micro-structure and the boundary

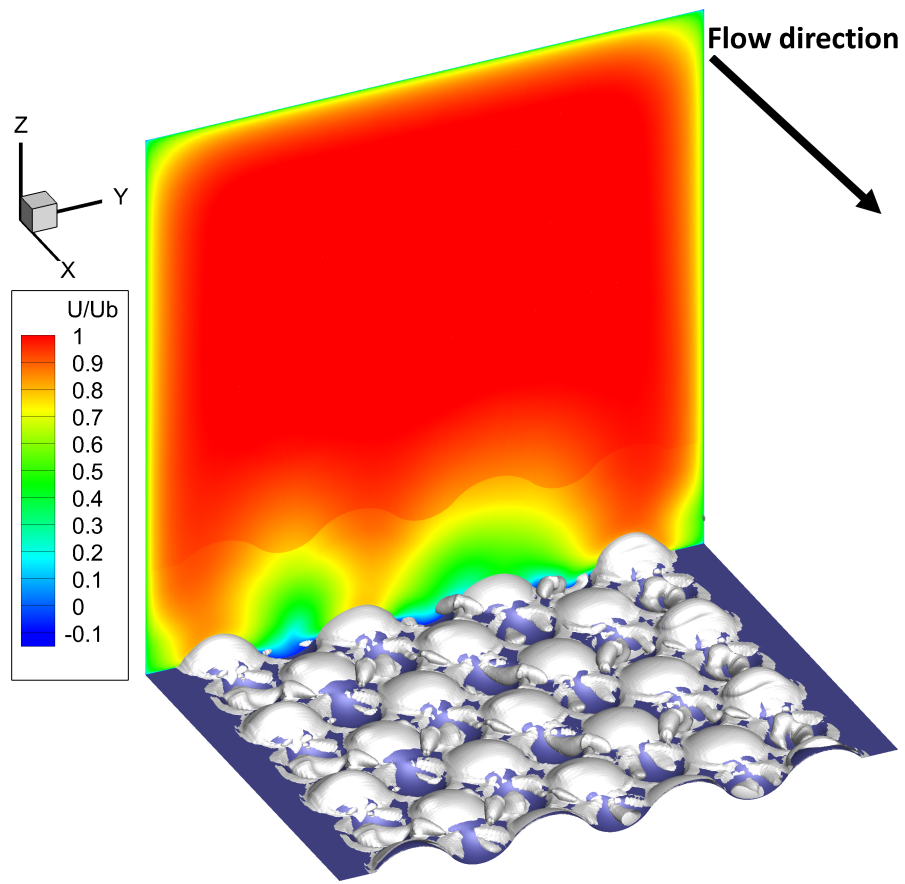


(a) Instantaneous

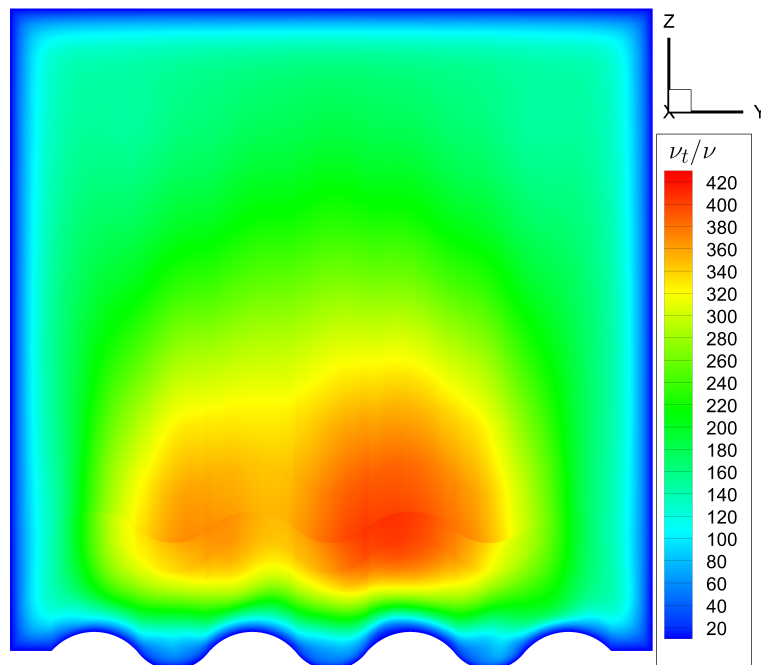


(b) Time-averaged

Figure 5.15 ILES: Q criterion iso-surfaces and normalised axial velocity contours

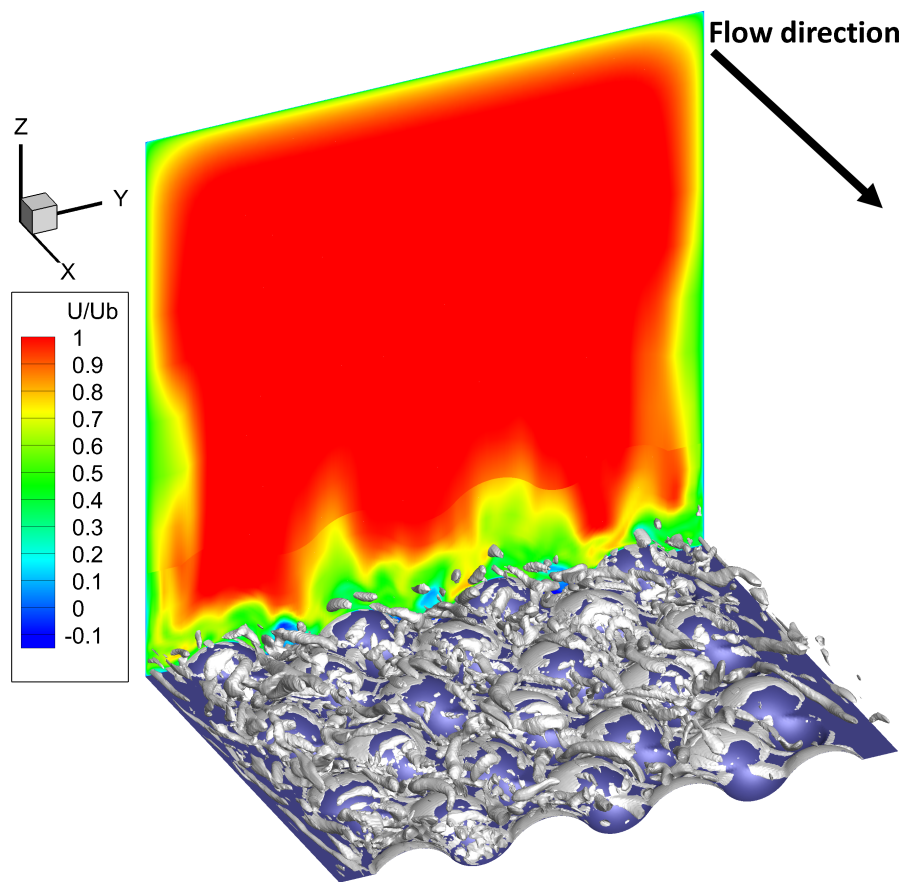


(a) Instantaneous

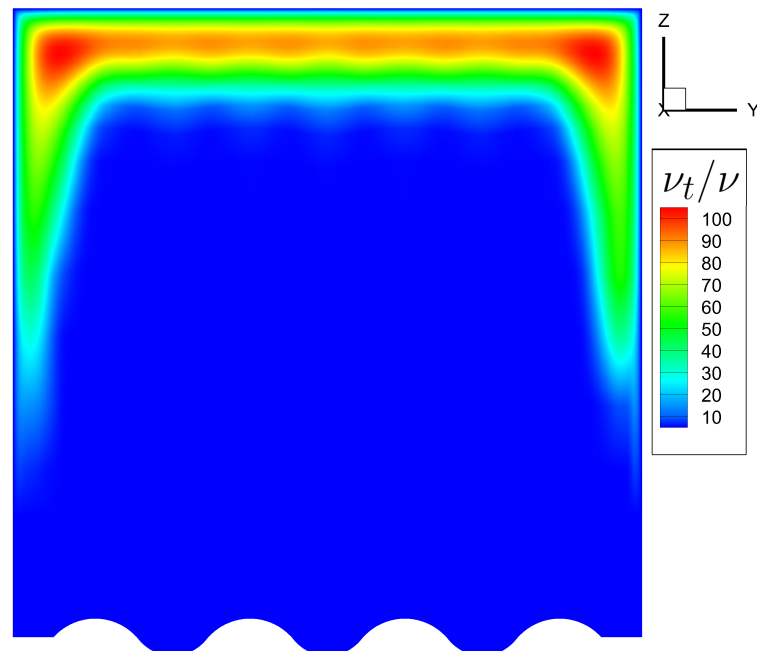


(b) Normalised instantaneous turbulent viscosity

Figure 5.16 SA: Q criterion iso-surfaces, normalised axial velocity contours and normalised turbulent viscosity contours



(a) Instantaneous



(b) Normalised instantaneous turbulent viscosity

Figure 5.17 Hybrid ILES: Q criterion iso-surfaces, normalised axial velocity contours and normalised turbulent viscosity contours

layers of the side walls. This mechanism is quite prominent for the ILES and hybrid ILES configurations, and its evolution is illustrated in Figure 5.19. Such region occupied by a secondary vortex does not appear in the SA solution.

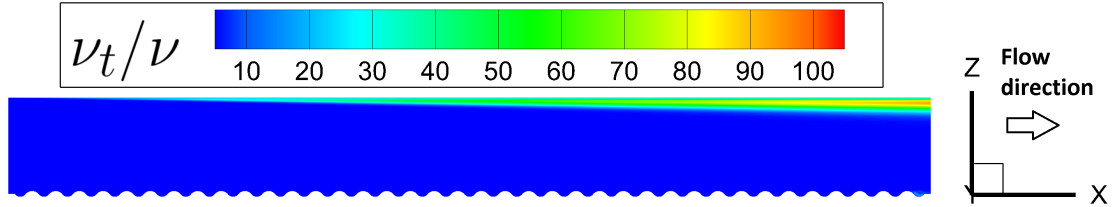


Figure 5.18 Hybrid ILES: Normalised turbulent viscosity contour at midspan

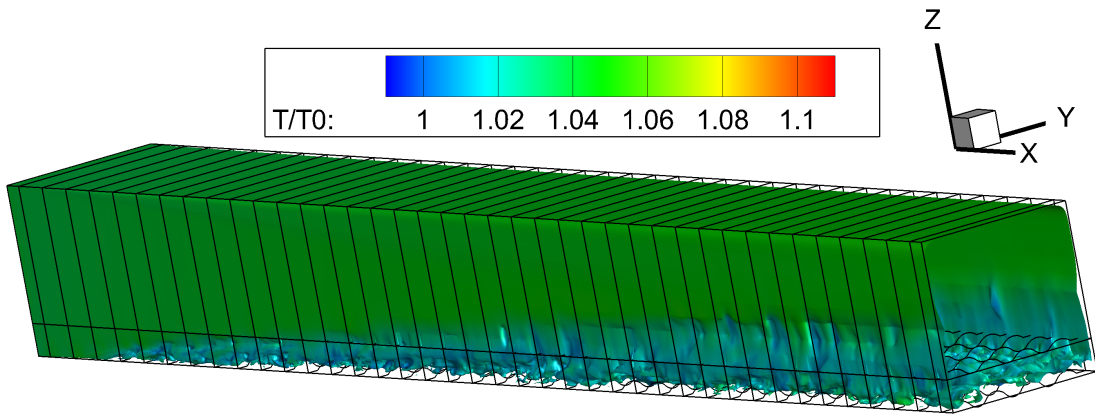


Figure 5.19 ILES: Instantaneous normalised axial velocity iso-surfaces ($U/U_b = 0.65$), contoured by normalised temperature for visualisation purposes

An additional difference in the vortical structures is observed between the ILES and the SA solutions. Firstly, the instantaneous ILES solution (Figure 5.15a) shows a chaotic flow field, as well as strong vortical structures being shed by both the pimples and the dimples. Similarly, the time-averaged ILES solution (Figure 5.15b) shows the shed vortices, as well as the vortical structures generated by the dimples. These flow structures are weaker in the SA solution (Figure 5.16a), as they are mostly wrapped around the pimples, with a less pronounced vortex shedding. The vortical structures generated by the ILES will be further discussed later on.

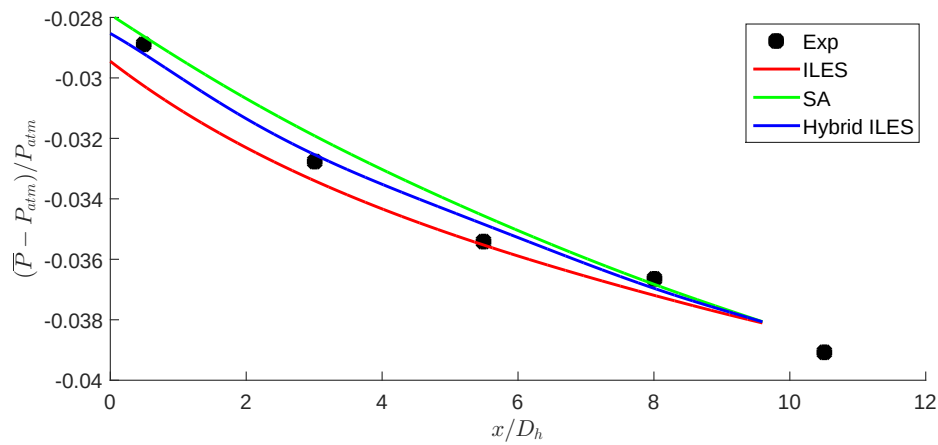
Comparison Against Experimental Data

Having compared the flow fields generated by the three turbulence modelling techniques, the validity of the numerical results is then investigated. Comparisons between the experimental and the numerical results are shown in Figure 5.20, generally showing good agreement in terms of the pressure drop and heat transfer. For the heat transfer results, the surface fitted with micro-structures and the opposite flat surface are considered.

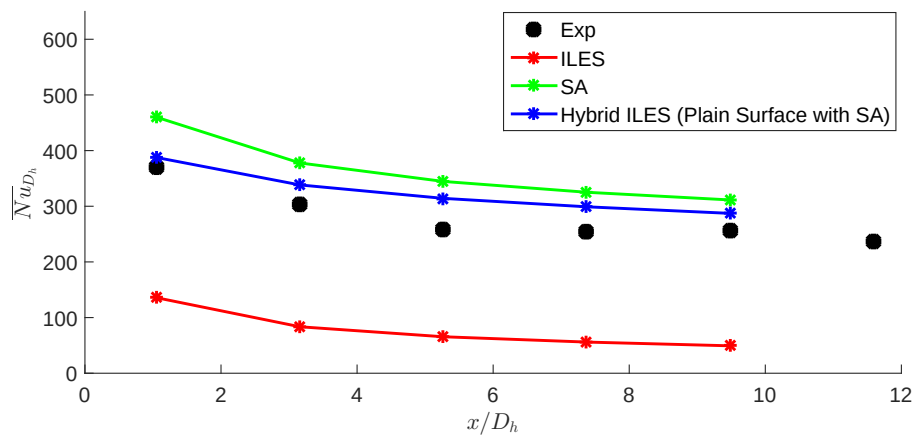
The ILES solution generates the pressure drop with the highest curvature, an indication that the flow is yet not fully developed as in the experiments. This is also indicated by the initial high pressure drop near the inlet. Still, the error is relatively small ($e_{a,max} \approx 5.6\%$). The flat surface heat transfer was significantly under-predicted by the numerical prediction (by ≈ 4.5 times), since the boundary layer remains laminar due to the coarse mesh ILES. For the surface fitted with micro-structures the numerical heat transfer compares well with the experimental results, especially after the first datum point ($e_{a,max} \approx 17\%$), which is subject to the transitioning phase. Considering the assumed inlet conditions for the numerical solutions, the predicted Nu of the pimple-dimple surface also demonstrates good agreement with the experimental measurements.

The SA solution demonstrates the lowest pressure loss ($e_{a,max} \approx 3\%$), as the flow is assumed to be fully turbulent without a transitioning phase. The heat transfer of the flat surface is closer to the experimental results than the ILES results ($e_{a,max} \approx 28\%$). However, the trend of the heat transfer of the roughened surface is poorly predicted when RANS turbulence modelling is applied, an indication of the problematic prediction of the bluff body separation by RANS, as discussed in Chapters 2 and 3.

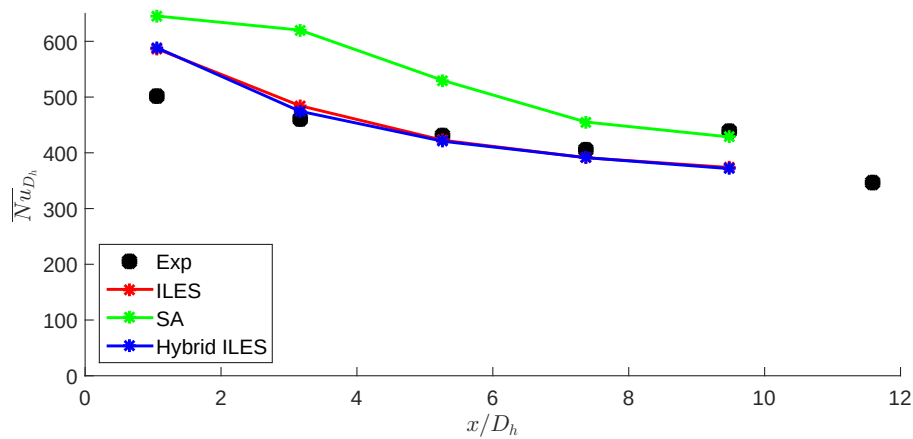
The hybrid ILES shows the most accurate pressure drop prediction, indicating in turn the most accurate prediction of the wall shear stress, with the trend being positioned between



(a) Pressure drop



(b) Heat transfer: Plain wall (flat surface positioned opposite of the surface fitted with micro-structures)



(c) Heat transfer: Surface fitted with micro-structures

Figure 5.20 Comparison of time-averaged properties between experimental data (Exp) and the three CFD configurations (ILES, SA and hybrid ILES). For the heat transfer results, the points are positioned at the centres of each heater

those of the ILES and SA configurations ($e_{a,max} \approx 2.6\%$). The heat transfer of the flat surface is also best predicted by this configuration ($e_{a,max} \approx 15\%$), while the prediction for the bottom surface is identical to the ILES result.

Flow Mechanisms

The unsteady simulation provides an opportunity to look at the transient and the time-mean flow properties around the surface with the positive and negative micro-structures. The heat transfer results (Figure 5.21) provide indications of the existence of interesting flow structures. The pimples demonstrate higher heat transfer at two upstream regions (Figure 5.21-(A)). This

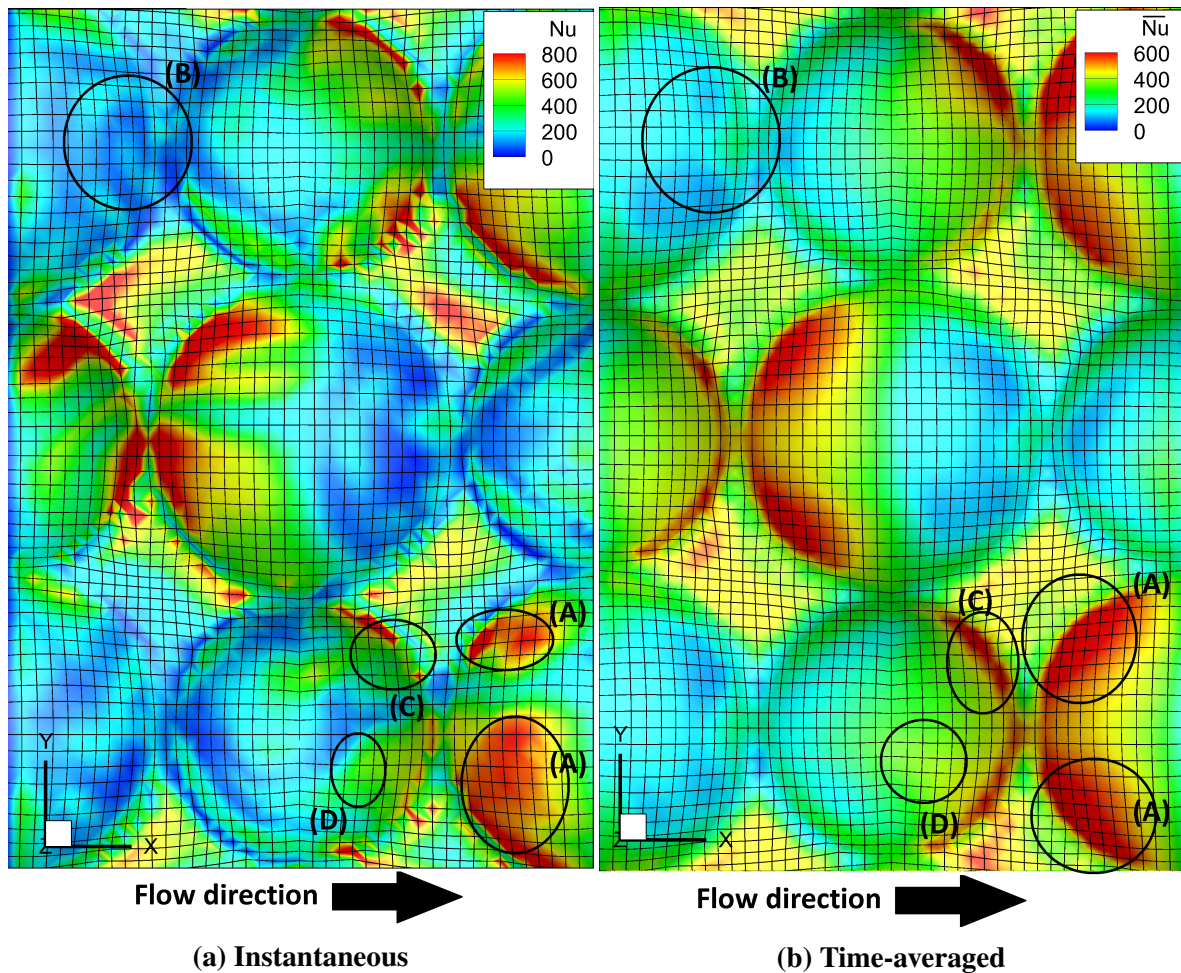


Figure 5.21 ILES: Top view of local Nu contours. Value limits do not reflect actual limits across the domain

behaviour is a combined effect of the flow acceleration between the pimples and the flow

impingement upon the upstream face (Figures 5.22 and 5.24a-(A)). The downstream faces are subject to flow separation (B), hence the low values of the Nu .

Additionally, the dimples demonstrate heat transfer at the downstream faces and the flat regions downstream of their edges (C). This is due to the flow impinging on the downstream face of the dimples and the flow accelerating over the rims. Also, the dimples demonstrate heat transfer contributions at their valleys (D). The probable underlying mechanism is the reattachment of the separated flow, caused by the pimples (Figure 5.24-(E)), as well as the vortical structures (Figure 5.25-(F)) that are formed inside the dimples.

In Figure 5.26 the vortical structures generated near the wall fitted with micro-structures are shown. As mentioned earlier, there are vortices being shed by the top regions of the pimples (A), similarly to the vortices observed in Chapter 3, while vortical structures near the surface of the cavity also seem to form (B). The vortices within the cavities seem to be formed due to the surface topology, which forces the flow entering the cavity to roll into a recirculating structure.

In Figures 5.27 and 5.28 the migration of the flow around the pimples and inside the dimples, respectively, is visualised. Pairs of counter-rotating vortices are selected and followed through each time-step, denoted in Latin numerals. In Figure 5.27, the selected vortices are being shed from the pimples, impinging to the downstream pimples and rolling over to be shed again. Figure 5.28 is focused on the formation of the vortices within the dimple. Counter-rotating structures seem to be continuously forming, not necessarily both at the same instant, and then propagated downstream outside of the dimples. These structures alongside with the flow acceleration between the pimples result in the surface topology surrounding the peaks and valleys to be significant contributors to local heat transfer enhancement, with distinctive signatures in surface heat transfer, as shown in Figure 5.29.

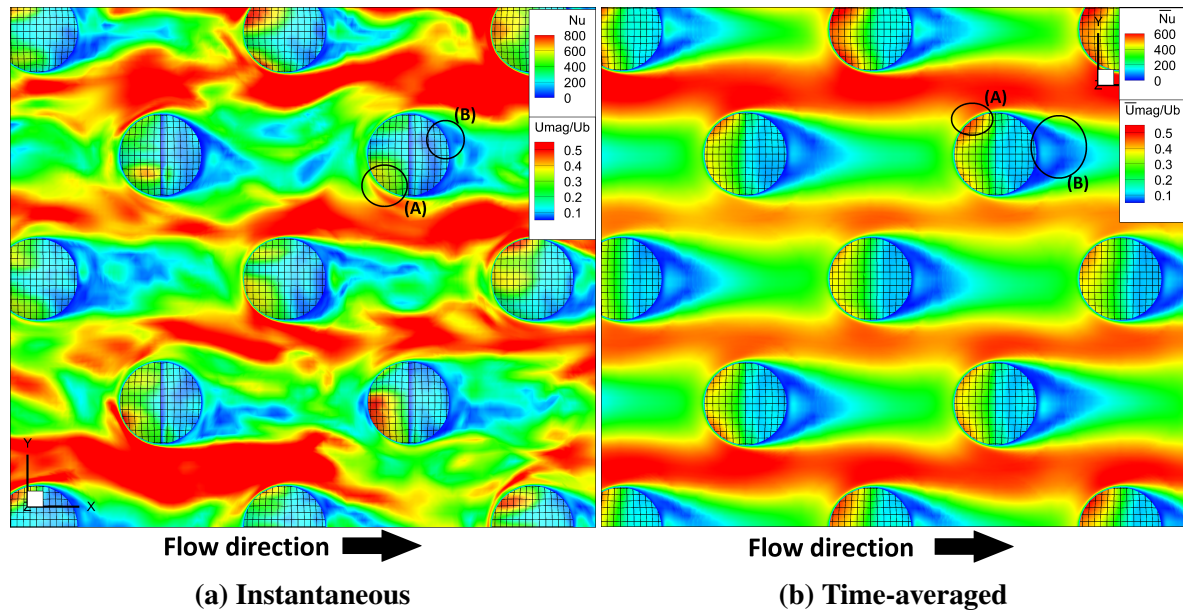


Figure 5.22 ILES: Normalised velocity magnitude contours at X-Y plane located at $z = 0.01969D_h$ above reference plane. Grid regions denote the protruding top of the pimples. Nu contours are also shown on the pimples

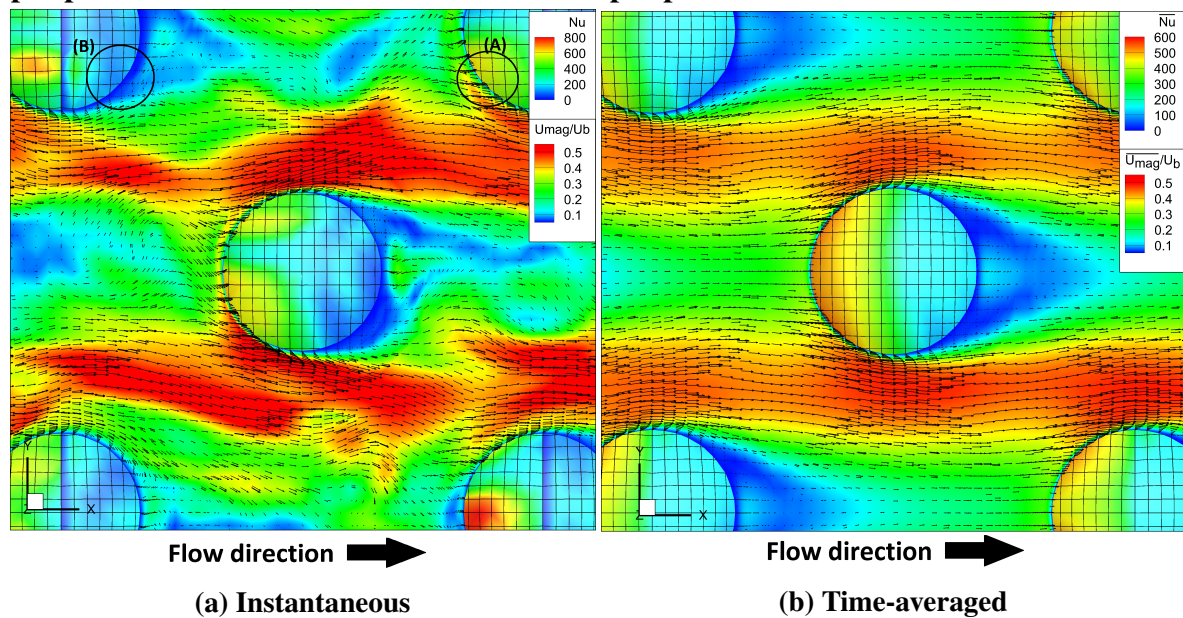


Figure 5.23 ILES: Close-up of Figure 5.22 with vectors which are tangent to the studied plane

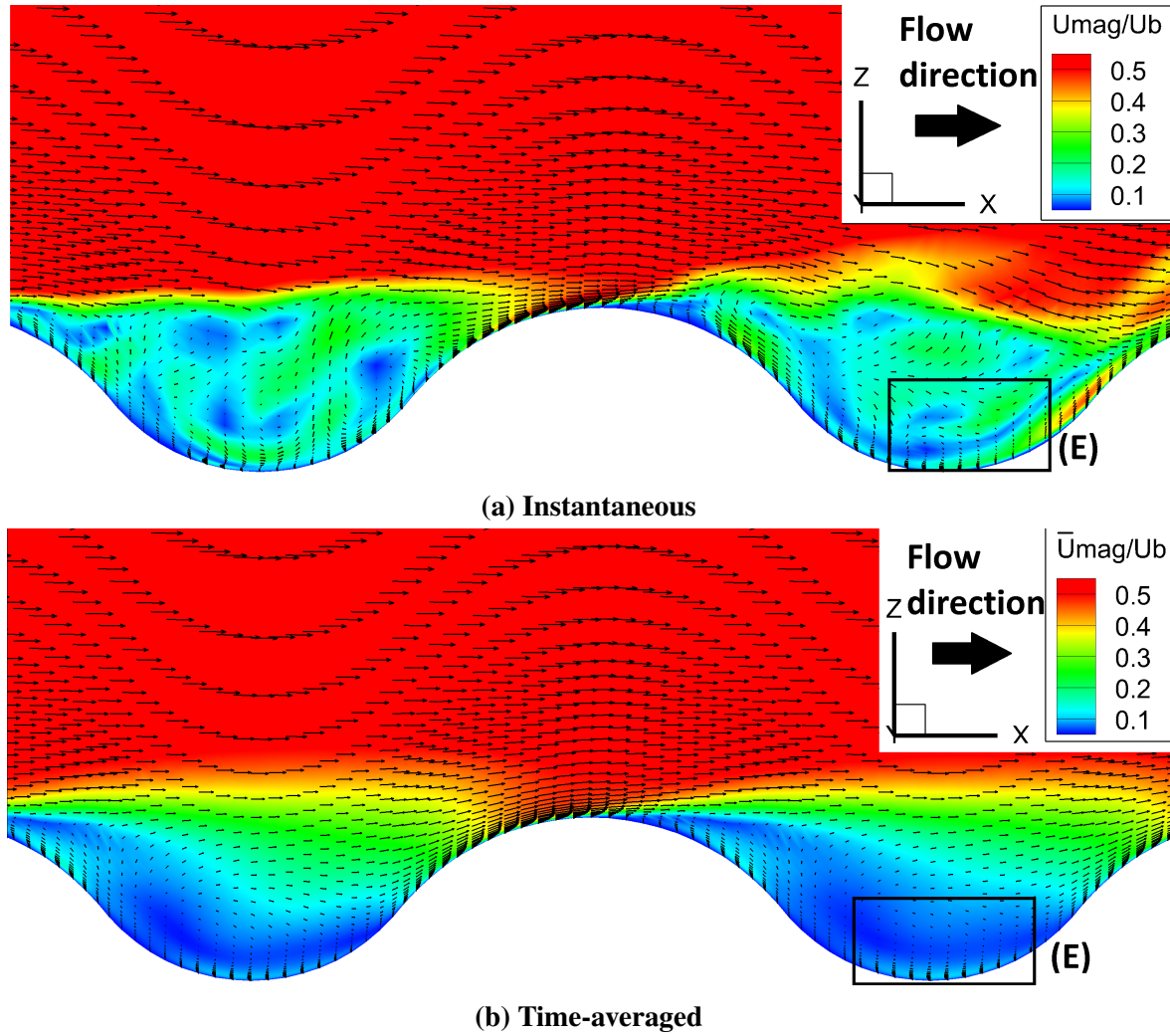


Figure 5.24 ILES: Normalised velocity magnitude contours at X-Z plane located at the midspan. Vectors are tangent to the studied plane

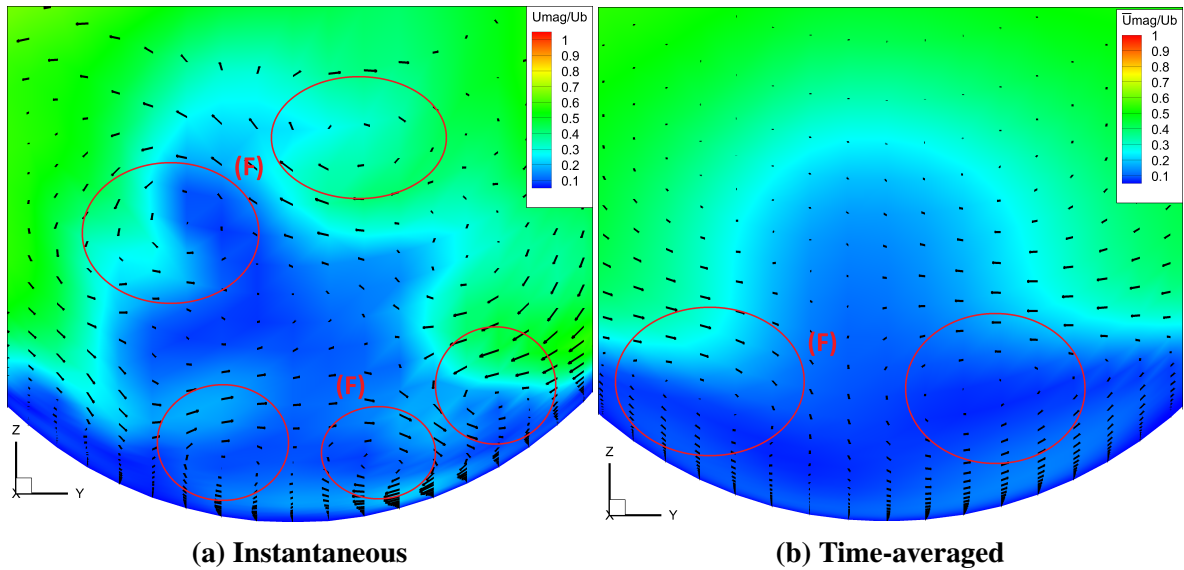


Figure 5.25 ILES: Normalised velocity magnitude contours at Y-Z plane located at the centre of the cavity. Vectors are tangent to the shown plane. Flow direction is perpendicular to the plane

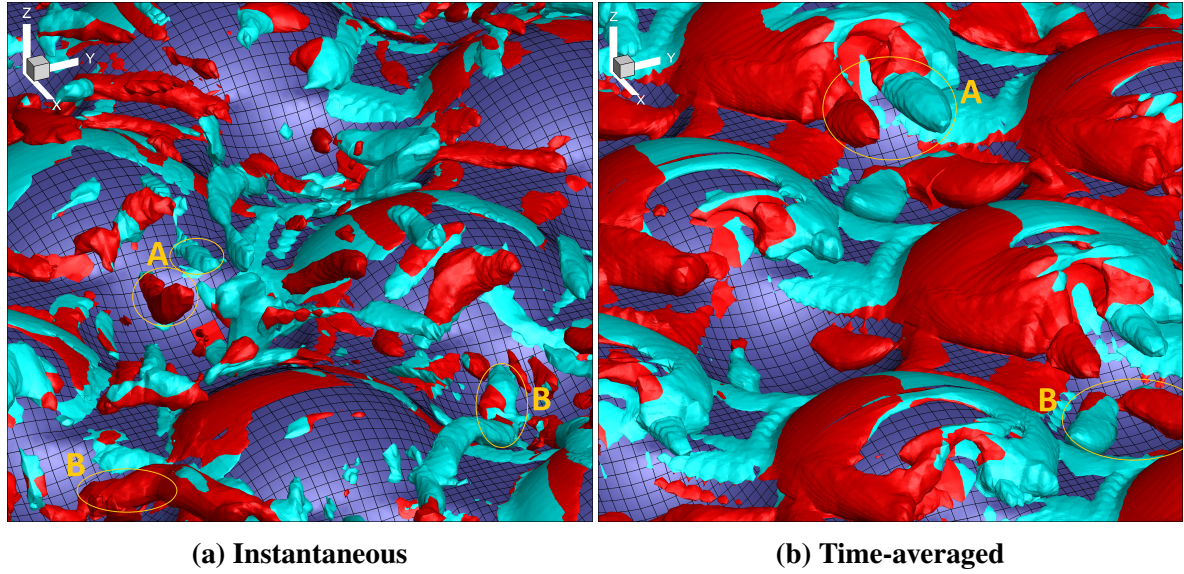


Figure 5.26 ILES: Close-up of the iso-surfaces of Q criterion. Flow structures are contoured red and blue for positive and negative helicity, respectively. Flow direction points in the X direction

The above analysis can also explain Figure 5.13b from the previous subsection. The figure is located at the central line of the row, thus the peaks and valleys are taken into consideration. The sides of the pimples show the highest heat transfer because of the accelerating flow, while the heat transfer at the peak of the pimple is low due to flow separation. At the location of the figure, where the wakes introduced by the pimples are close to reattachment within the dimples, the dimples begin to show non-negligible heat transfer enhancement. There, the rotating incoming flow promotes heat transfer that is comparable to the heat transfer on the pimples.

These observed mechanisms are in line with the flow features observed by Elyyan and Tafti [83]. In their study, a LES with the Smagorinsky model [76] is performed. The finite-volume second order Central Difference solver is applied to an incompressible channel flow of height H . The identical micro-structures are periodically repeated every $3.24H$ in both directions, as shown in Figure 5.30. The foot diameter (D_f) of the micro-structures is equal to $2H$ and the height/depth of the protrusions is equal to $0.4H$. No slip and isothermal conditions are applied

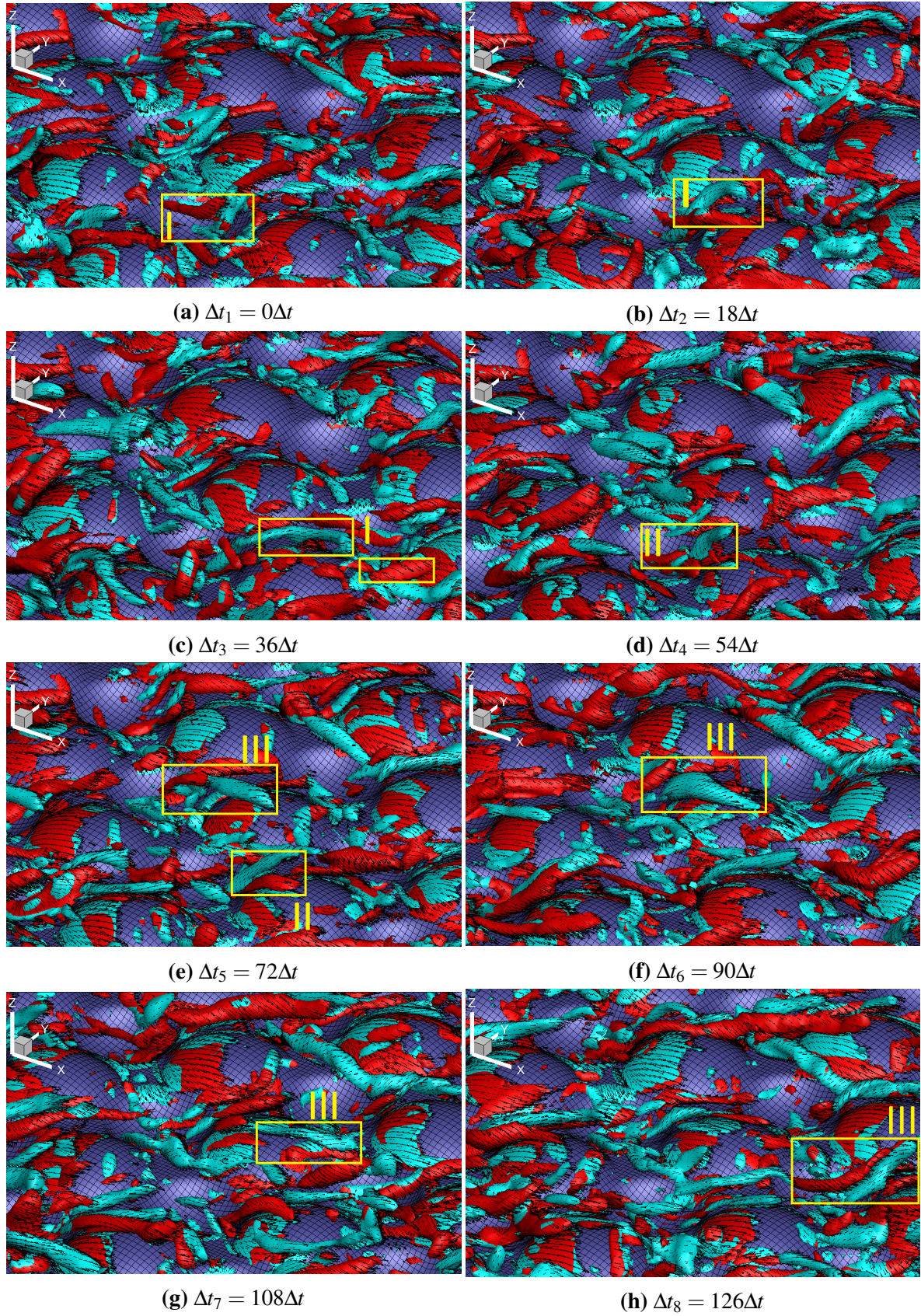


Figure 5.27 ILES: Instantaneous Q criterion iso-surfaces, fitted with vectors visualising the vortical structures which interact with the pimples ($\Delta t = 2.64 \times 10^{-6}s$). Red and blue denote positive and negative helicity, respectively. Flow direction points in the X direction

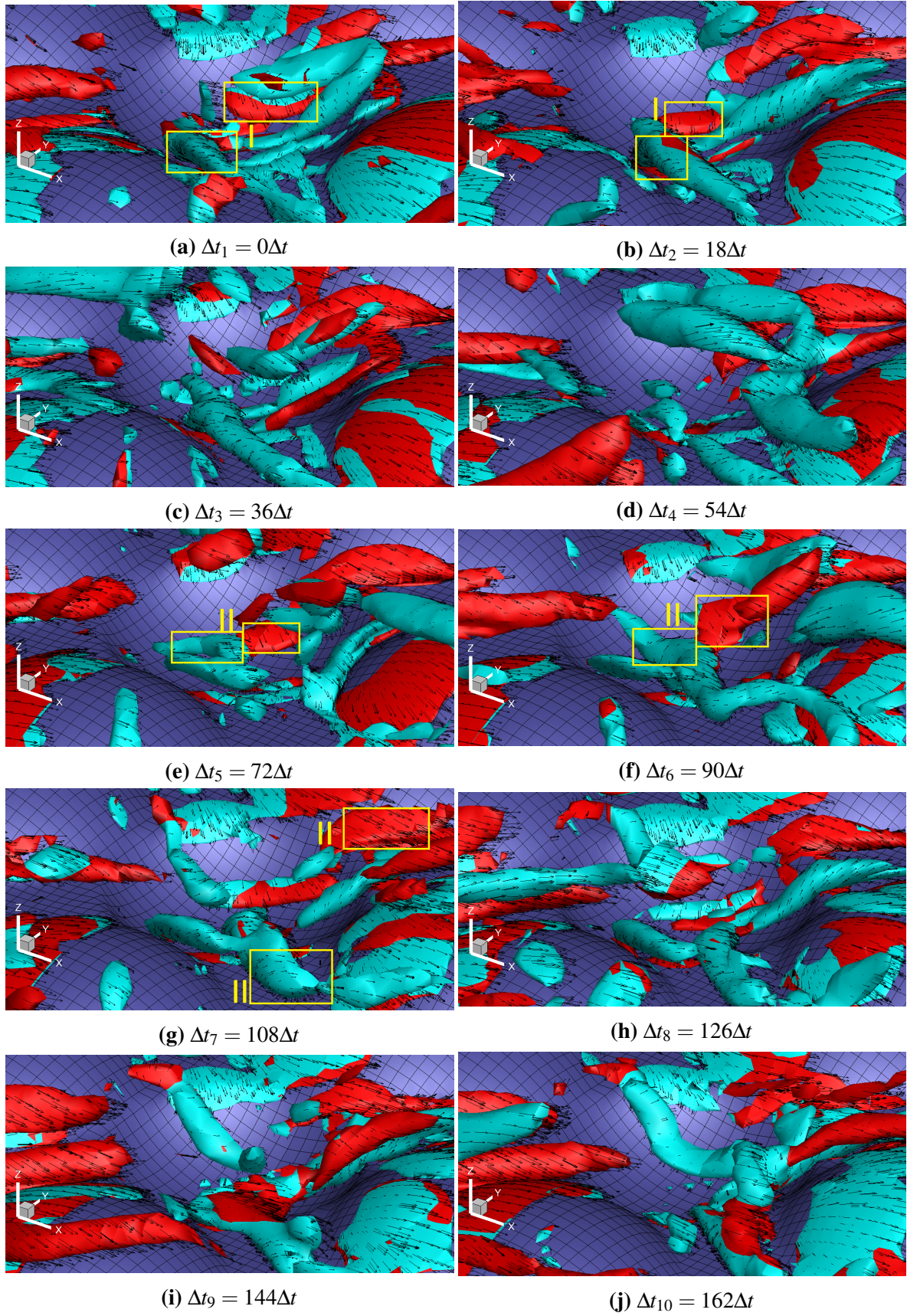


Figure 5.28 ILES: Instantaneous Q criterion iso-surfaces, fitted with vectors visualising the vortical structures within the dimples ($\Delta t = 2.64 \times 10^{-6}s$). Red and blue denote positive and negative helicity, respectively. Flow direction points in the X direction

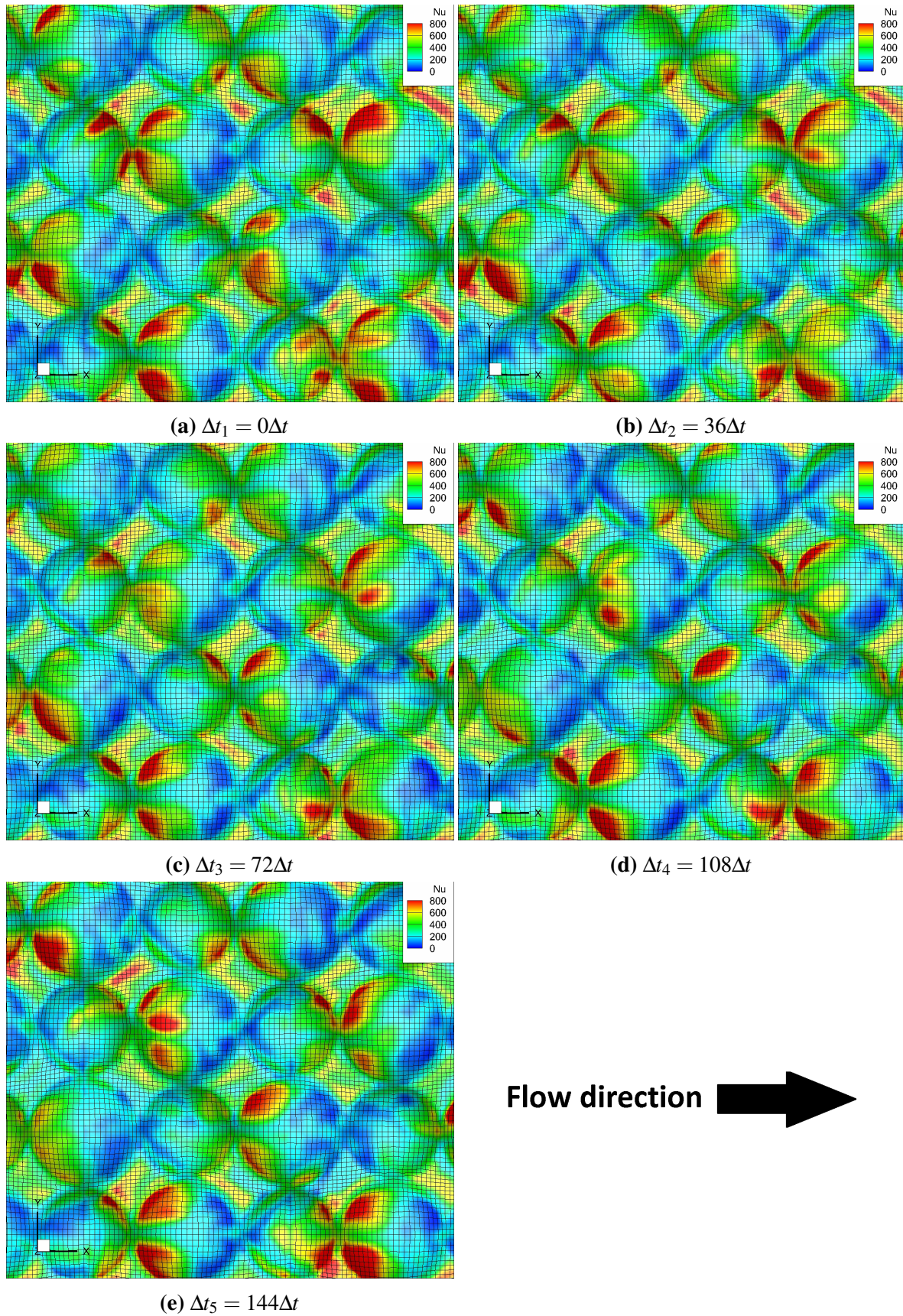


Figure 5.29 ILES: Top view of instantaneous contours of local normalised heat transfer ($\Delta t = 2.64 \times 10^{-6}s$)

to the viscous walls and periodicity is assumed in the spanwise and streamwise directions. The computational domain is restricted within a volume of $H \times 3.24H \times 3.24H/2$, including a single pimple and a single dimple. It is observed that heat transfer is affected by the flow acceleration between the pimples, as well as by the impinging flow on the downstream face of the dimples. Moreover, the turbulence created by the edges of the dimples increases the heat transfer at the smooth surface downstream of the micro-features.

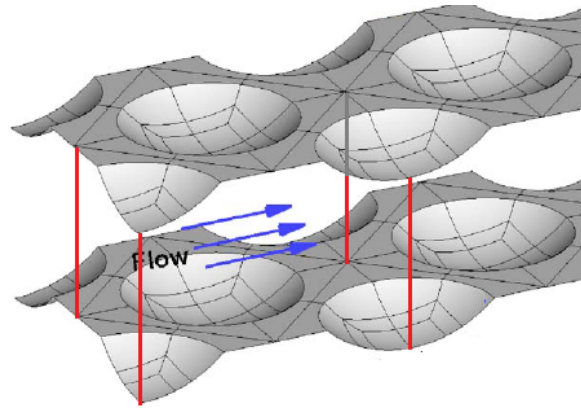
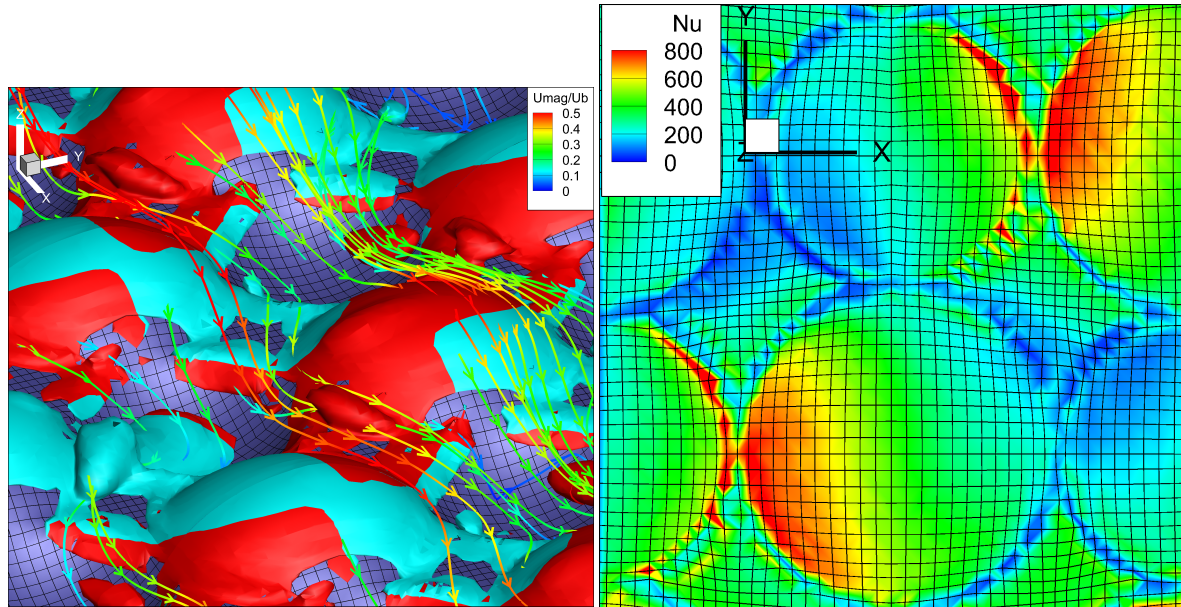


Figure 5.30 Geometry studied in [83]. Red edges outline the volume of the computational domain

The highly unsteady behaviour captured by the ILES is contrasted to the largely steady flow field generated by the SA turbulence model. In Figure 5.31a the ordered flow is depicted by both the lack of pronounced vortical structures and the streamlines, which in turn results in a largely time-invariant heat flux distribution (Figure 5.31b). In addition, the contours of heat transfer show qualitatively different patterns when compared against both the instantaneous and the time-averaged results of the ILES. The most striking difference is observed at the upstream stagnation region of the pimples, where the SA modelling predicts a single front of high heat transfer instead of two distinctive regions. This disagreement can be linked to the area-averaged heat transfer trends observed earlier, which failed to match the trends of the heat transfer measurements.



(a) Q criterion iso-surfaces. Red and blue denote positive and negative helicity, respectively. Streamtraces are coloured by the normalised velocity magnitude

(b) Normalised heat transfer

Figure 5.31 SA: Instantaneous vortices and normalised heat transfer. Flow direction points in the X direction

5.5 Remarks

In this chapter the validity of the baseline solver was investigated. Initially, the solver accuracy was tested for a 2D flat plate and a 2D cylinder test cases. The solver was verified through comparisons to well-documented analytical, semi-empirical and experimental data from the open literature, as well as benchmark results from a commercial solver. Also, a dual-mesh configuration of 3D pedestals was compared to an equivalent multi-block setup, indicating satisfactory base-fine domain interface treatment as well as proper wall treatment for the embedded walls.

The main test case presented is a surface with pimples and dimples micro-features subjected to a channel flow. The computational domain resolved a part of the channel and three turbulence modelling approaches were applied. Coarse mesh ILES results, particularly when the boundaries layer of the top plain surface is modelled by the SA model, matched

well with the experimental measurements. RANS with an one-equation turbulence model was shown to perform poorly when resolving the flow structures around the DREs, generating less accurate solutions. The transient simulations provided the opportunity to identify key flow mechanisms that affect the heat transfer on the pimples and dimples in different ways.

The results of this case underline the two premises of the Block Spectral approach [63–66]: a) the need for local fine mesh resolution, and b) the similarity in flow patterns among micro-structures/elements, thus the ability of propagating the flow patterns from one micro-structure to another. It is of particular interest to contrast the instantaneous unsteady flow patterns with their time-averaged counterparts. The instantaneous flow patterns are clearly not similar, and thus they cannot be propagated among the micro-structures. The time-averaged patterns, on the contrary, appear to be quite similar and therefore they can be propagated, thus confirming the basic assumption made for the methodology [66].

Chapter 6

Validation and Demonstration of Block

Spectral Method

This chapter includes a first of the kind verification and demonstration of the Block Spectral Method when applied to wall surfaces fitted with a large number of regular roughness elements. Hence, the validation study covers several cases with surface micro-structures, where the 1D and 2D mapping of the TST (as discussed in Chapter 4) are studied. Finally, the computational gain of the developed method is demonstrated through its application to one large domain case with more than 1000 surface micro-structures. A significant portion of the present chapter has been included in Kapsis et al. [84].

6.1 Unsteady 3D Flow Subject to Regular Roughness

The capability verification of the BSM is carried out in three parts depending on the direction of the source term propagation (TST mapping) and is dealt with in the following order: a) freestream direction, b) lateral direction and c) both directions.

Four mesh configuration are considered:

1. Only base: no embedded domain is applied.
2. Fully embedded: all micro-structures and their vicinity are refined by the embedded domain.
3. Partially embedded without source terms: some DREs and their vicinity are refined by the embedded domain.
4. Partially embedded with BS source terms: some DREs and their vicinity are refined by the embedded domain, while TST are introduced to the unrefined regions.

Since the Block Spectral Method aims to correct the time-averaged flow of the base domain (coarse mesh), all the presented results refer to the coarse mesh, unless noted otherwise. Also, the relative error e_a of the predicted property Φ between the fully embedded and the partially embedded with BS source terms configurations is reported, calculated as:

$$e_a = \left| \frac{\overline{\Phi}_{\text{fully embedded}} - \overline{\Phi}_{\text{partially embedded with BS source terms}}}{\overline{\Phi}_{\text{fully embedded}}} \right| \quad (6.1)$$

For this section rectangular micro-structures are considered. The micro-structures are subject to $Re_\ell \approx 2600$ ($U_b \approx 57m/s$), based on the characteristic length $\ell = 0.5mm$, and $Ma_b \approx 0.17$. The computational configuration is identical to the description given in section 5.1.

6.1.1 Block Spectral Mapping Applied in the Freestream (X) Direction

Initially, the proof of concept of the BSM application to 3D regular roughness is performed considering mapping in the freestream direction. The height of the micro-structures is $h = \ell$ and their bases are $2\ell \times 2\ell$, as shown in Figure 6.1. Each micro-structure is 5 coarse cells high (in the Z direction) and 3 coarse cells long/wide (in the X/Y directions). Both spanwise and

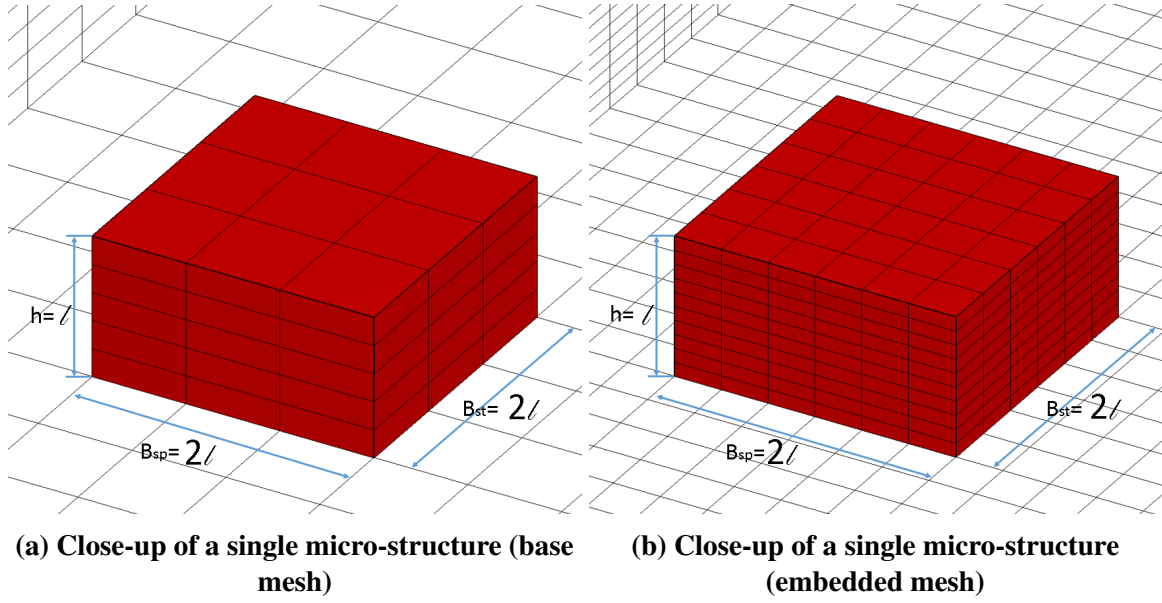


Figure 6.1 Mapping in a single direction: Mesh distribution on and around the micro-structures (solid domain is outlined by the red surfaces)

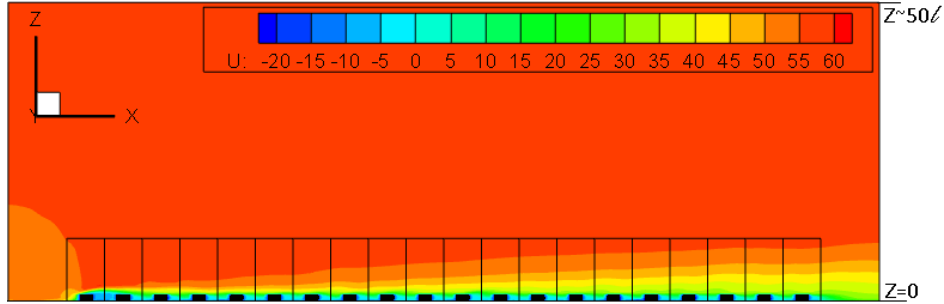
streamwise spacing are set to be 6ℓ . The computational domain consists of 20 micro-structures equidistantly distributed in the freestream direction (Figure 6.2a). Table 6.1 sums up the physical properties of the micro-structures.

Table 6.1 Mapping in a single direction: Geometric and pattern properties of the micro-structures

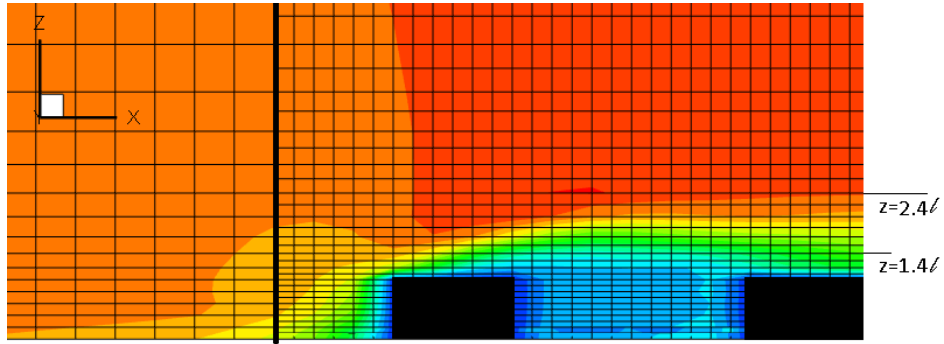
B_{st}/h	B_{sp}/h	λ_{st}/h	λ_{sp}/h
2	2	6	6

The embedded domain consists of a single block per micro-structure and it is created by refining the coarse domain $2 \times 2 \times 2$ times, as shown in Figure 6.2b. Each embedded block covers a volume of $9 \times 9 \times 18$ (in X, Y and Z directions respectively) coarse domain cells. The far-field top boundary is $\approx 50\ell$ away from the bottom wall fitted with the micro-structures. The flow field is non-periodic, hence mirrored source regions are used (as described in Appendix F). This means that the number of defined source regions in the domain is equal to the number of harmonics plus one.

For this subsection, a plotting line that is located at the midspan, 2 coarse cells above the micro-structures ($Z = 1.4\ell$) is selected for comparing results of different solution methods. In the end of this subsection, a comparison is also presented for a plotting line located at the midspan 5 coarse cells above the micro-structures ($Z = 2.4\ell$).

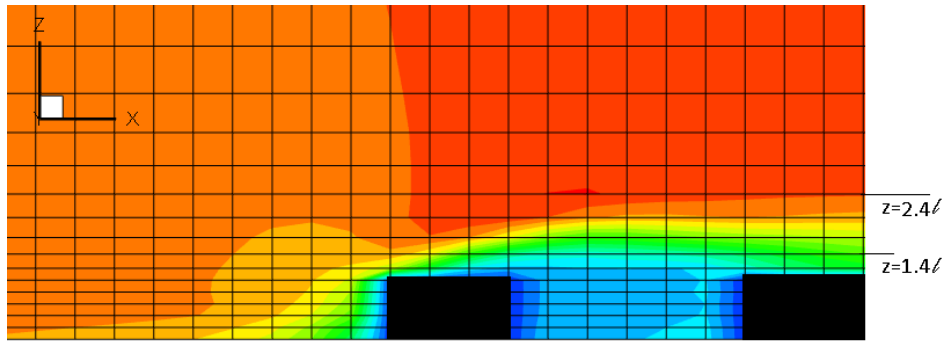


(a) Overview of base (one block to cover the whole domain) and embedded (20 blocks) domains



base-embedded mesh interface

(b) Close-up of first micro-structure (dual-mesh)



(c) Close-up of first micro-structure (base mesh only)

Figure 6.2 Instantaneous freestream velocity at midspan for the fully embedded setup. Blackened areas denote embedded solid domains

The 3D nature of the flow, being resolved by the embedded mesh, can be seen in Figure 6.3. The instantaneous freestream velocity of the well-developed flow is non-symmetric with

respect to the midspan plane. The oscillatory nature of the flow can be clearly seen from the seventh, counting from upstream, micro-structure and downstream.

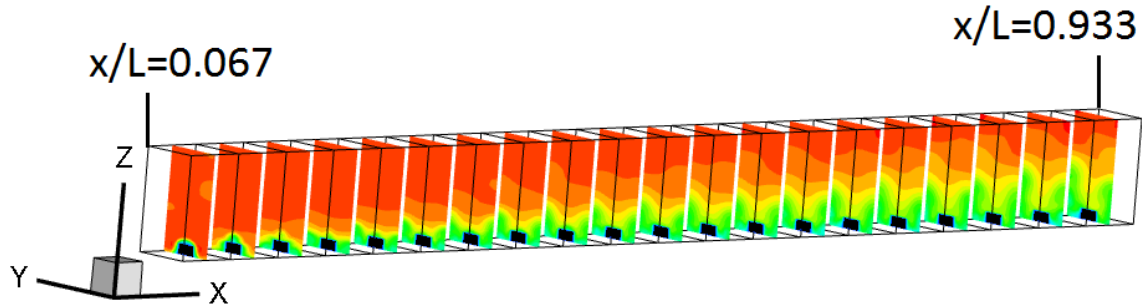


Figure 6.3 Embedded domain contours of instantaneous freestream velocity slicing through the centres of each micro-structure

The following study:

- demonstrates a 3D case that requires local refinement in order to improve the capturing of transient flow phenomena;
- demonstrates that the BSM can produce time-averaged results similar to those of a fully embedded mesh.

The first above-mentioned point is satisfied straight away from Figure 6.4 where the embedded fine mesh is shown to markedly affect the time-averaged results.

One Harmonic

In the second stage of the analysis most of the embedded blocks are removed (Figure 6.5). Since one harmonic is utilised, two source regions need to be defined for the calculation of the spectral representation. The source regions are always surrounded by embedded regions in order to ensure sufficient resolution of the upstream and downstream flow effects on the source region. The amount of additional embedded mesh blocks, particularly upstream, could affect the flow properties in the source region. For now it is assumed that one additional embedded mesh block upstream and another downstream of the source region are sufficient.

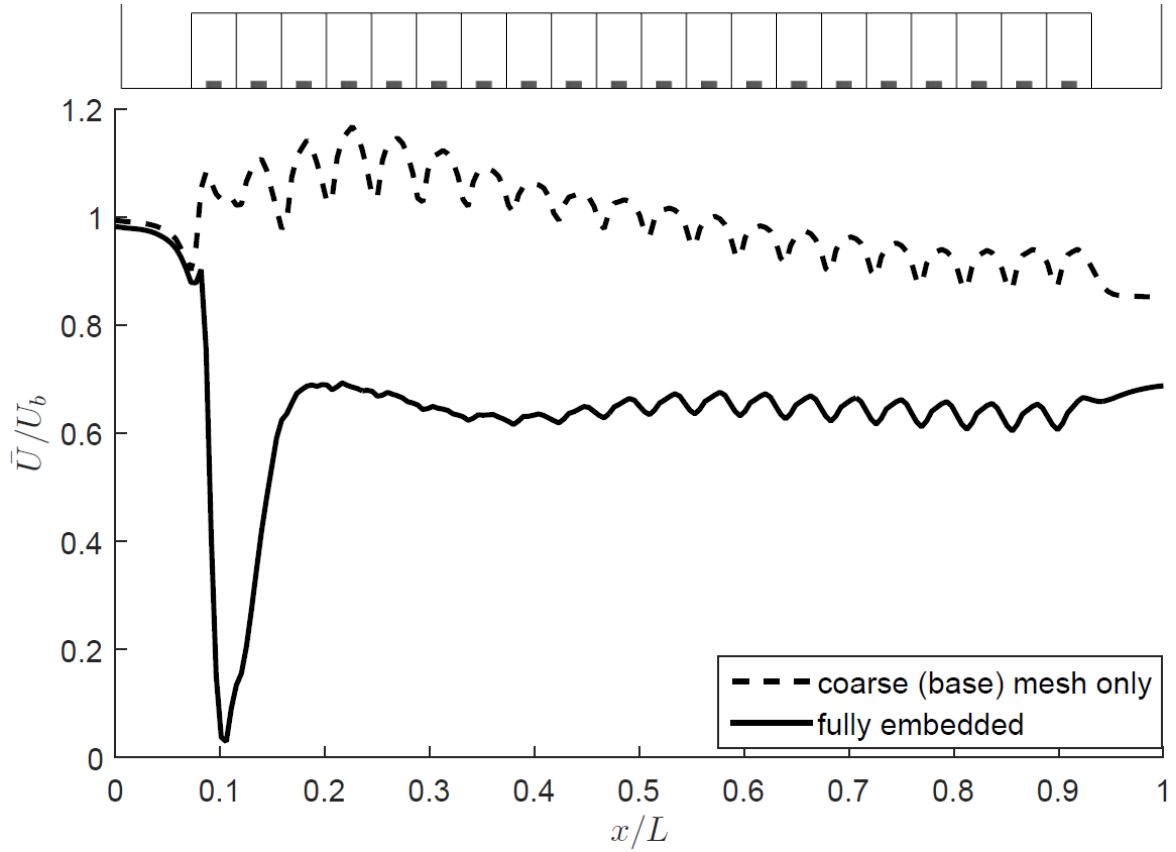


Figure 6.4 Normalised time-averaged freestream velocity along the plotting line $Z = 1.4\ell$ for only coarse mesh and fully embedded configurations

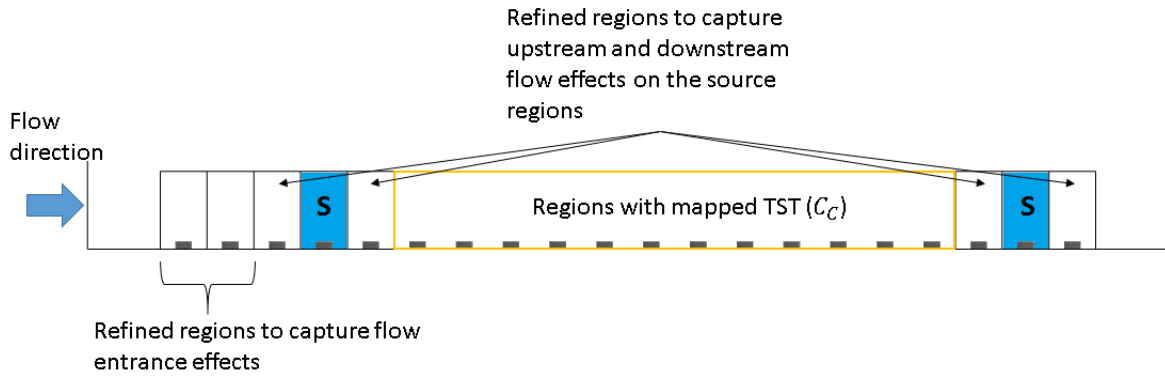


Figure 6.5 Embedded mesh block distribution at midspan. S denotes the source regions

Additionally, as the flow is still being developed and severely separated after the first upstream micro-structure, the first two embedded regions are additionally included to allow for the entrance effects to settle down.

In Figures 6.6 and 6.7 it is shown how the partially embedded without source terms configuration fails to match the fully embedded trends. This can be justified by the fact that the upstream history effect is not properly captured due to the lack of full mesh embedding.

This issue is resolved once the TST of the Block Spectral Method are introduced. When partial embedding is considered, the BSM is needed, as the refined regions do not match the results of the fully embedded setup. The Block Spectral solution does not provide a perfect match with the fully embedded case, it is, however, worth-mentioning that just one harmonic is utilised. Higher numbers of harmonics should provide a better match.

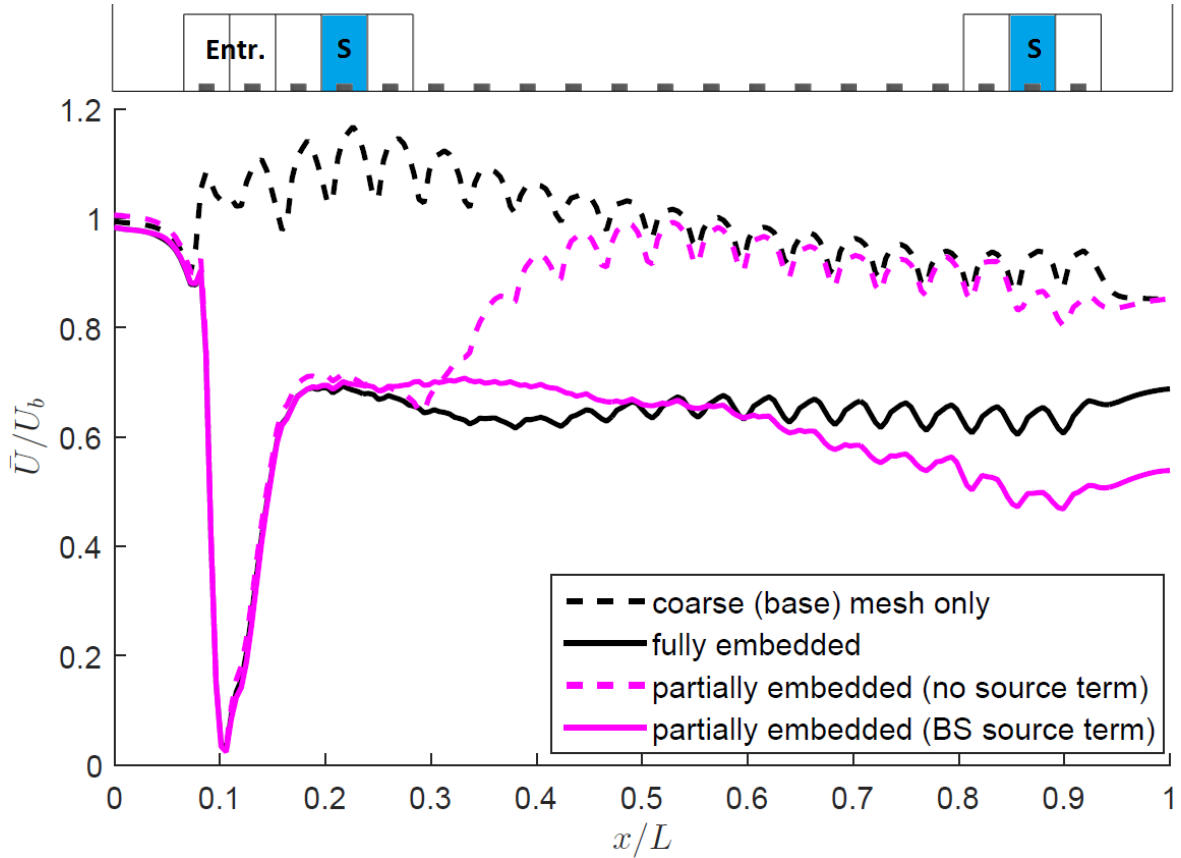


Figure 6.6 Normalised time-averaged freestream velocity at the plotting line $Z = 1.4\ell$ for 1 harmonic. S denotes the source regions

Two Harmonics

Considering the above partially embedded configuration, the first source region does not seem to be subject to a fully developed boundary layer. This is reflected in the predicted by the BSM velocity trends. In the next stage, an additional embedded source region is added as shown in the top of Figures 6.8 and 6.9, thus 2 harmonics are considered for the BS source term propagation. Comparisons against the fully and partially embedded without

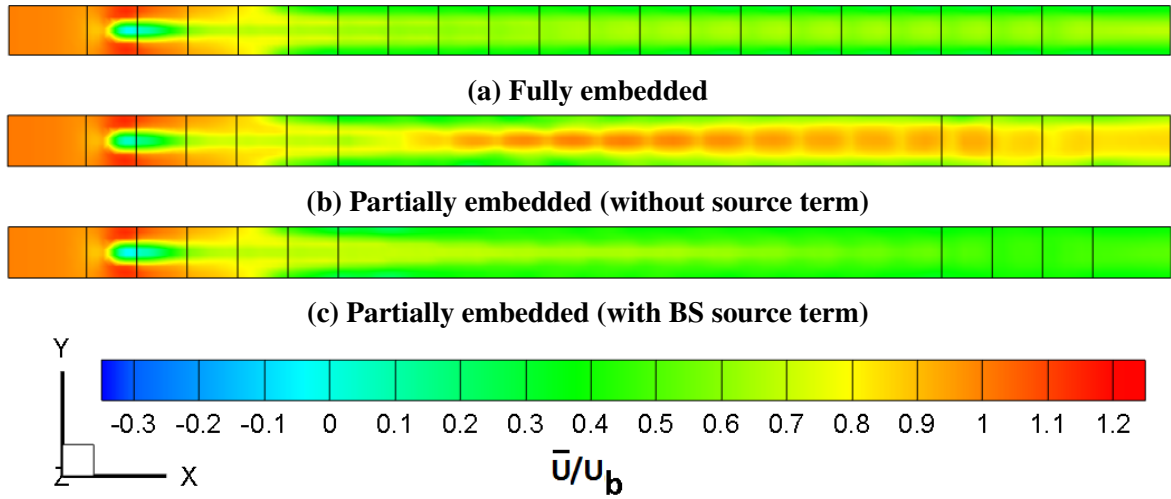


Figure 6.7 Contours of normalised time-averaged freestream velocity at a X-Y plane ($Z = 1.4\ell$) for 1 harmonic

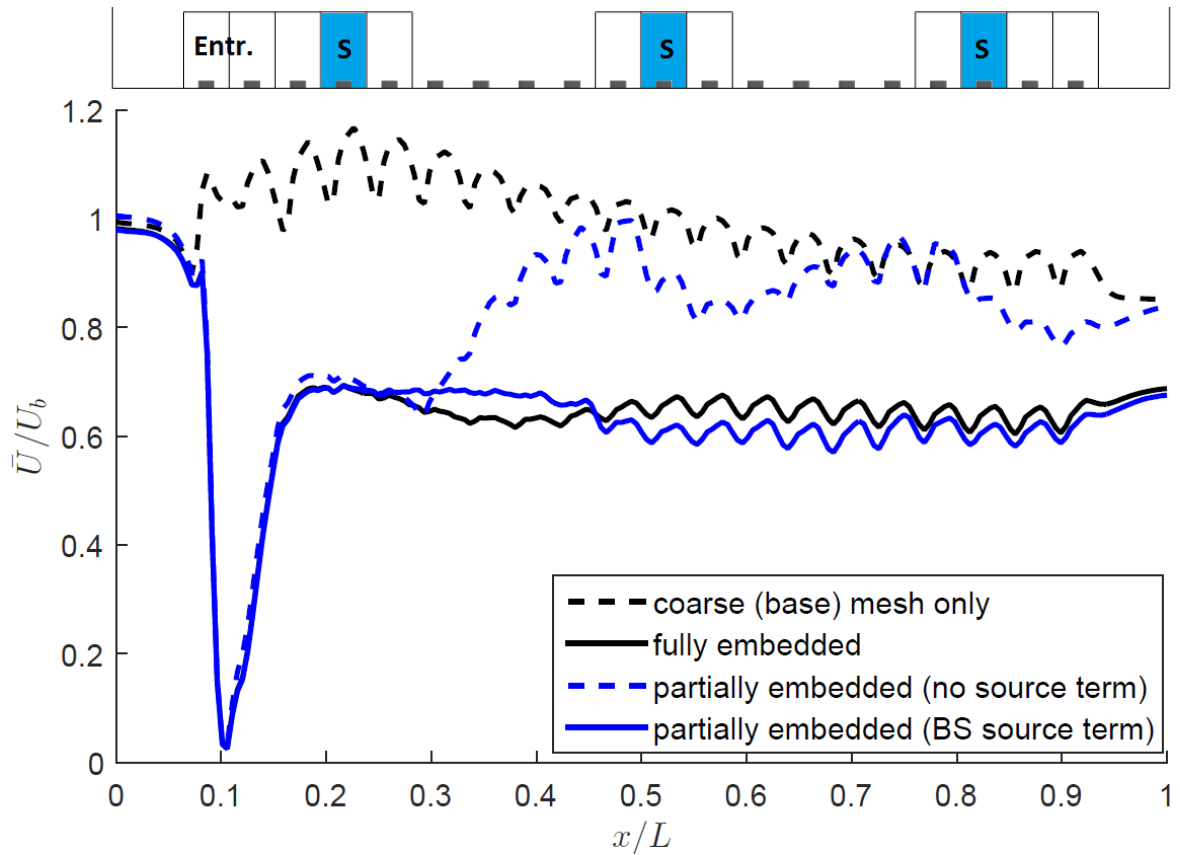


Figure 6.8 Normalised time-averaged freestream velocity at the plotting line $Z = 1.4\ell$ for 2 harmonics. S denotes the source regions

source terms configurations highlight the improved agreement of the BS solution with that of the fully embedded setup. Similar to the results of the BS with 1 harmonic, the present multi-scale method reduces the elevated time-averaged axial velocity values at the coarse

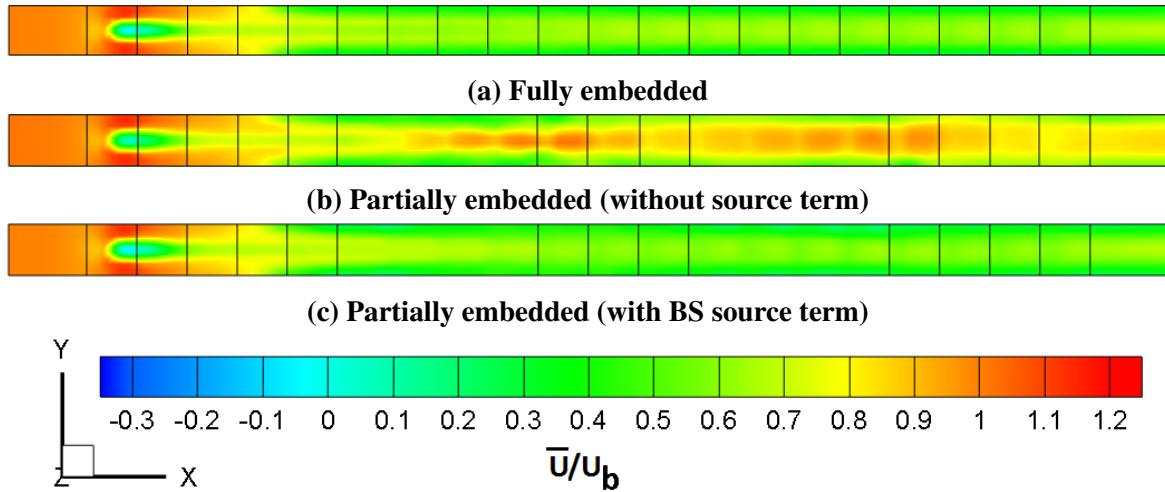


Figure 6.9 Contours of normalised time-averaged freestream velocity at a X-Y plane ($Z = 1.4\ell$) for 2 harmonics

(base) mesh of the partially embedded (without source terms) solution, providing a good match with the velocity of the fully embedded setup in terms of both trends and magnitudes.

Two Groups of Spectra

A similar predictive accuracy of the two harmonics configuration can be achieved with two spectral representations of one harmonic each. In this way, the domain is split into two regions with their respective BS group. The last case study in this subsection considers an embedded mesh block configuration similar to the previous test case with two harmonics, though the source mapping is performed by two groups of BS of 1 harmonic each. The spectra groups do not share any information, i.e. their spectra representations are built based only on the source regions selected for each BS group. The results are shown in Figure 6.10, demonstrating similar accuracy as in the above case. This serves as a demonstration of the capability to have separate source/mapped regions.

Finally, in Figure 6.11 the plotting line at $Z = 2.4\ell$ is considered. Similarly to the above analysis, agreement between fully embedded and BS solutions is observed, especially when 2 harmonics or 2 BS groups are employed. The relative error of the time-averaged axial velocity is presented on the plotting lines and contour plane considered above in Figures 6.12

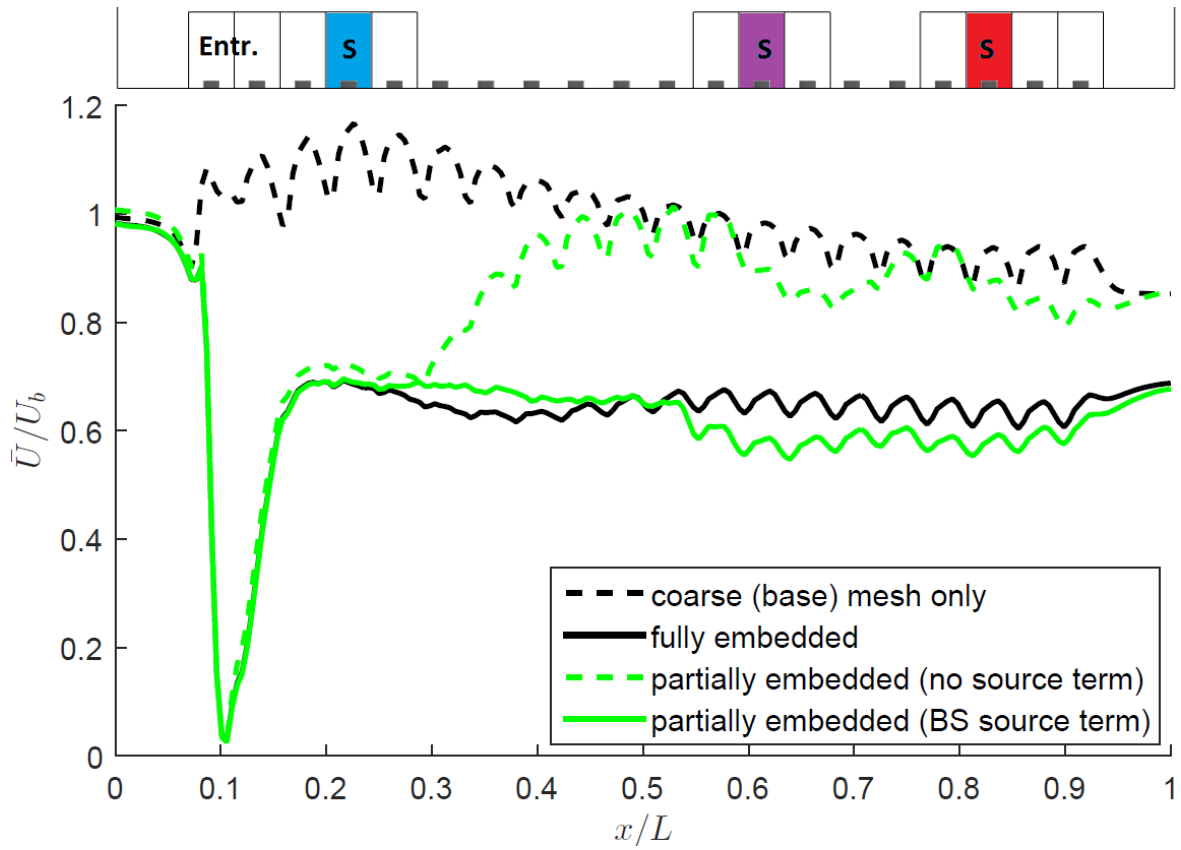


Figure 6.10 Normalised time-averaged freestream velocity at the plotting line ($Z = 1.4\ell$) for 2 BS groups. S denotes the source regions. Blue/Purple and Purple/Red constitute the source regions of the first and the second groups, respectively

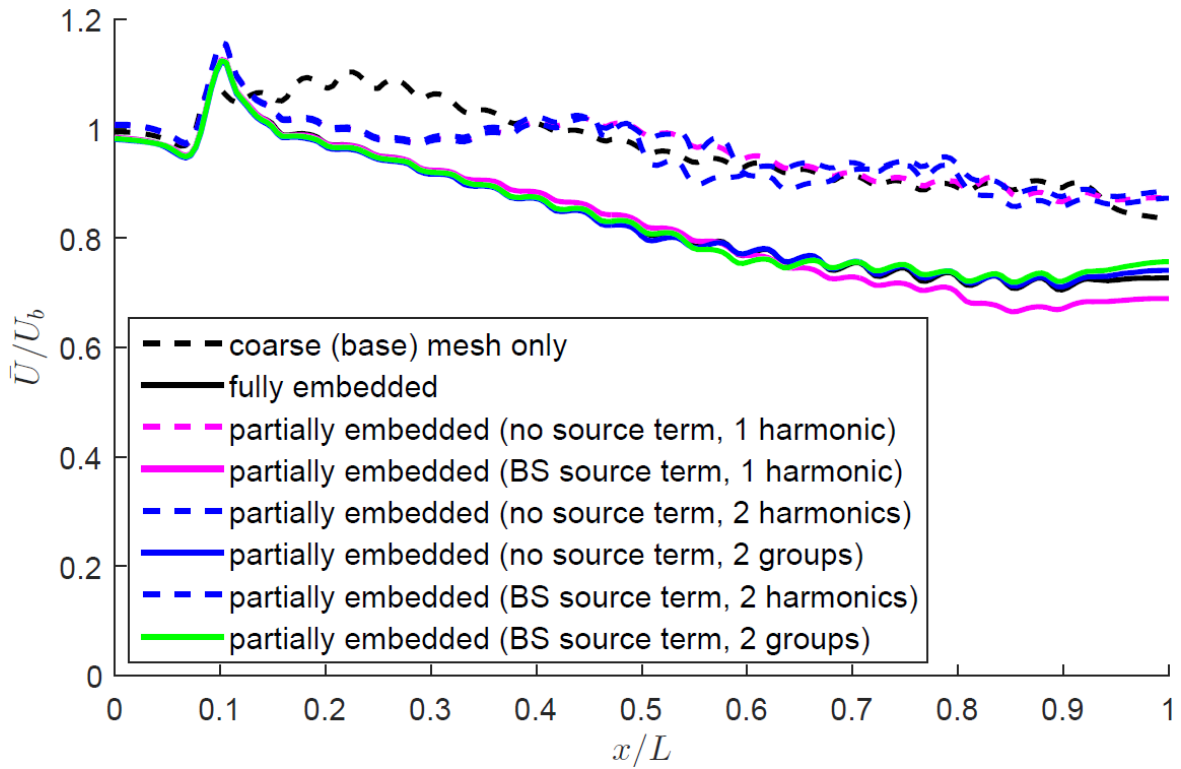


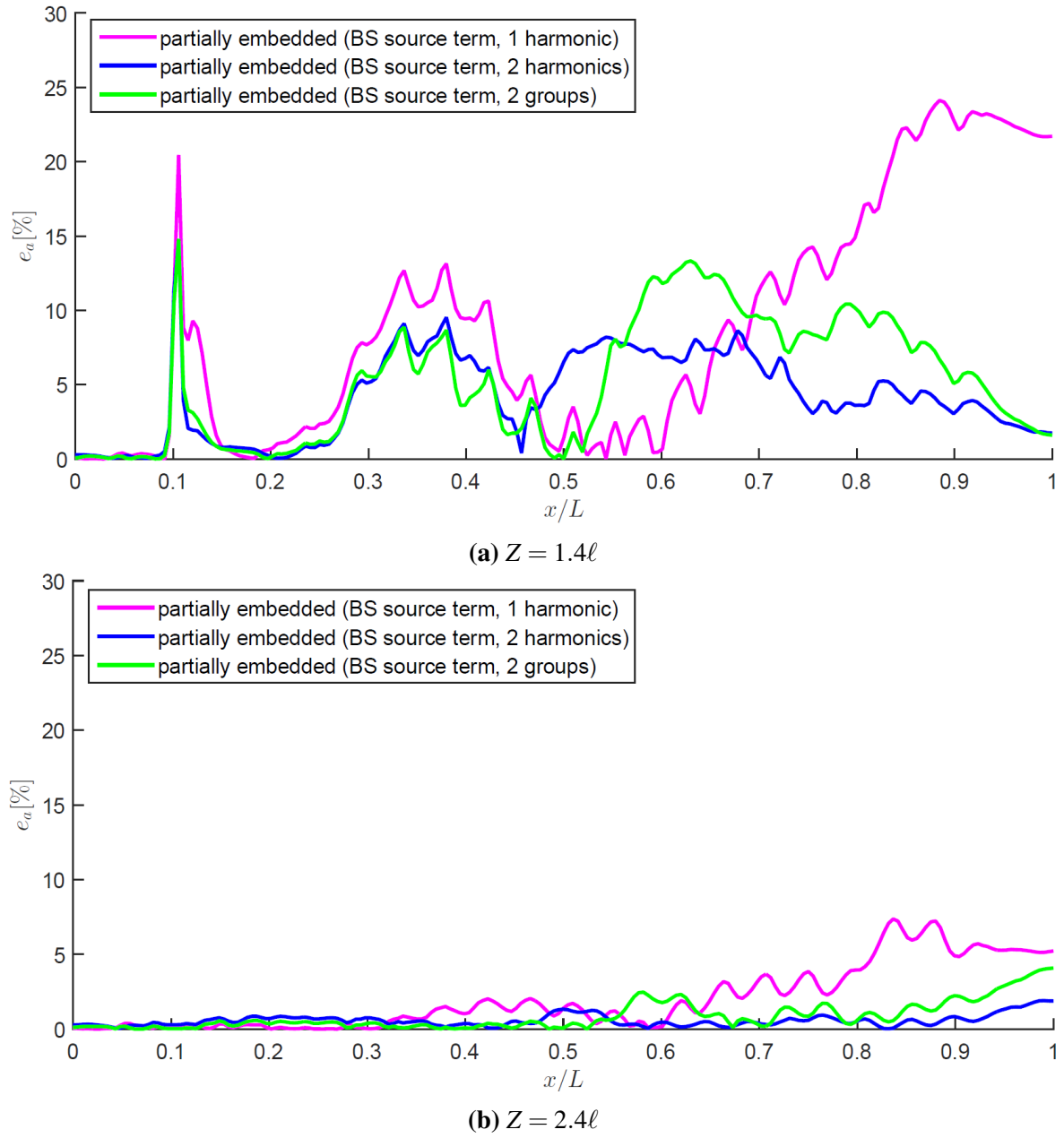
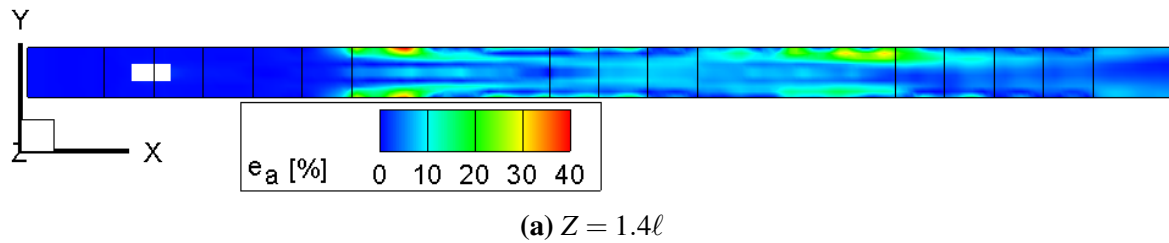
Figure 6.11 Normalised time-averaged freestream velocity at the plotting line $Z = 2.4\ell$

and 6.13, respectively. For the plotting line at $Z = 1.4\ell$ the area-averaged e_a is 8.8%, 4.2%, and 5.2% for the test cases with one harmonic, two harmonics, and two groups of spectra, respectively. For the plotting line at $Z = 2.4\ell$ the area-averaged values are dropped down to 2.0%, 0.6%, and 0.9% for the test cases with one harmonic, two harmonics, and two groups of spectra, respectively. For the test cases with two harmonics, area-averaged e_a of the X-Y plane at $Z = 1.4\ell$ is 5.6%. Any spikes in the e_a values that appear at small regions can be attributed to discrepancies introduced by the finite number of samples for the time-averaging.

Regarding the computational resources utilised, performance metrics are presented in Table 6.2. C_{map} is 1.08, 1.03, and 1.09 for the one harmonic, two harmonics, and two groups of spectra, respectively. The computational gain c_e is 1.13, 1.10, and 1.04 for the one harmonic, two harmonics, and two groups of spectra, respectively. Although the BSM requires a large number of micro-structures to demonstrate its capabilities, the prospect of the computational gain can already be observed.

Table 6.2 Performance metrics for the BS mapping in the freestream direction (1 core utilised). Metrics of the fully embedded configuration are used as reference values

ratio	Domain size	Time per iteration
fully embedded	1	1
partially embedded (no source term) one harmonic	0.82	0.62
partially embedded (with BS source term) one harmonic	0.82	0.67
partially embedded (no source term) two harmonics	0.88	0.78
partially embedded (with BS source term) two harmonics	0.88	0.80
partially embedded (with BS source term) two spectra groups	0.88	0.85

Figure 6.12 Relative error (e_a) at two plotting linesFigure 6.13 Relative error (e_a) at a X - Y plane ($Z = 1.4\ell$) for 2 harmonics. Blank regions correspond to $\bar{U} \approx 0$

6.1.2 Block Spectral Mapping Applied in the Lateral (Y) Direction

Having tested the Block Spectral Method in the freestream direction and proven its predictive capabilities when applied to 3D micro-structures that assist the flow into becoming unsteady, the validation study proceeds to investigate the BS mapping in the lateral direction. The micro-structures considered are of the same shape, size and spacing as before. The primary difference lies in the pattern: the domain consists of 2 columns of 18 micro-structures each as shown in Figure 6.14. The first column of micro-structures is fully embedded to capture the entrance effects.

In Figure 6.14 both the domain (a) and the BS setup (b) are shown. The embedded blocks are the same as in the previous study except for the embedded blocks of the second column which are extended in the freestream direction to capture the separated flow region, covering 20 coarse mesh cells in the streamwise direction. Due to the boundary conditions and the regular geometric pattern, the time-averaged flow field is expected to be fairly repetitive in the lateral direction. However, the lack of symmetry in the distribution of the embedded mesh blocks in the lateral direction such flow field is not generated by the partially embedded without source terms configuration.

Since the flow in the physical domain would be expected to be largely periodic, in the BS implementation the source regions are not mirrored, hence the number of defined source regions is 3, the exact amount needed for a single harmonic. In Figures 6.15 and 6.16 time-averaged axial velocity contour planes parallel to the viscous wall are presented.

The Block Spectral Method produces velocity patterns that are significantly closer to the laterally periodic velocity distributions of the fully embedded solution than those produced by the partially embedded without source terms configuration. The area-averaged relative error e_a of the time-averaged axial velocity is 9.7% and 7.3% for the planes at $Z = 3/5\ell$ and

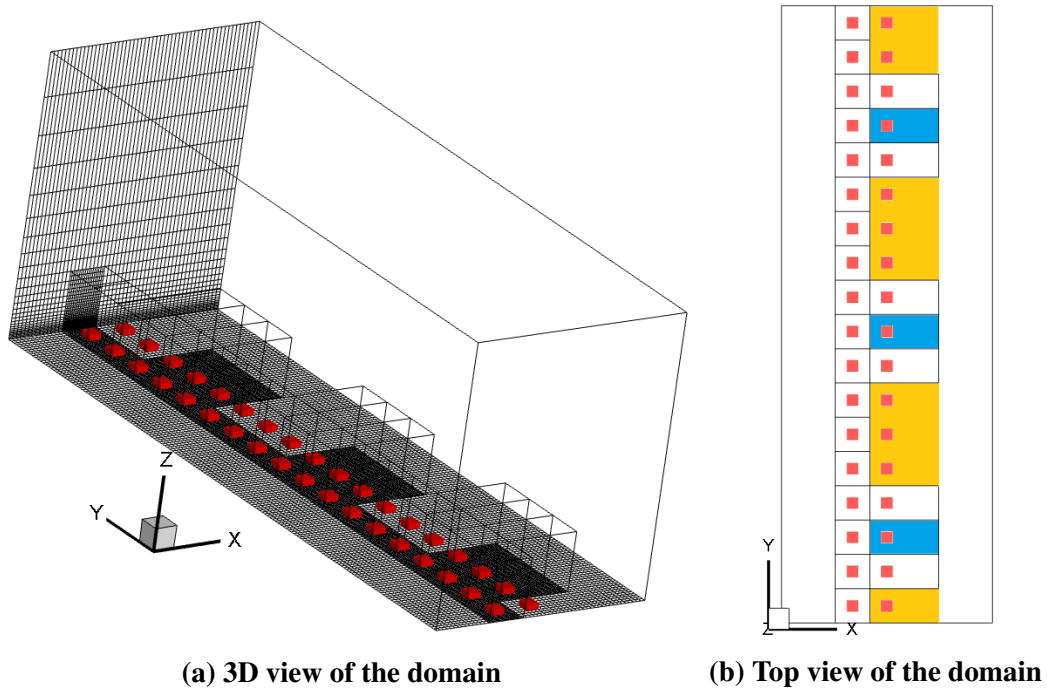


Figure 6.14 Embedded solid domain highlighted with red. Blue and yellow regions denote the BS source and mapped regions, respectively

$Z = 6/5\ell$, respectively. The local e_a at these two planes is also presented in Figure 6.17.

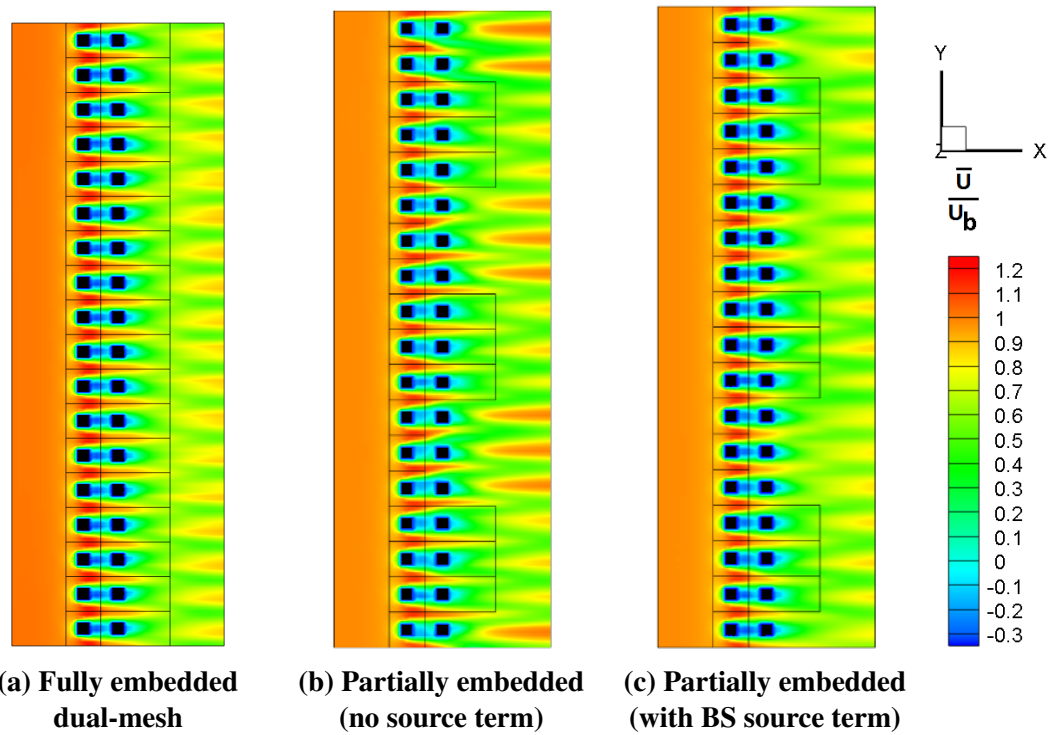
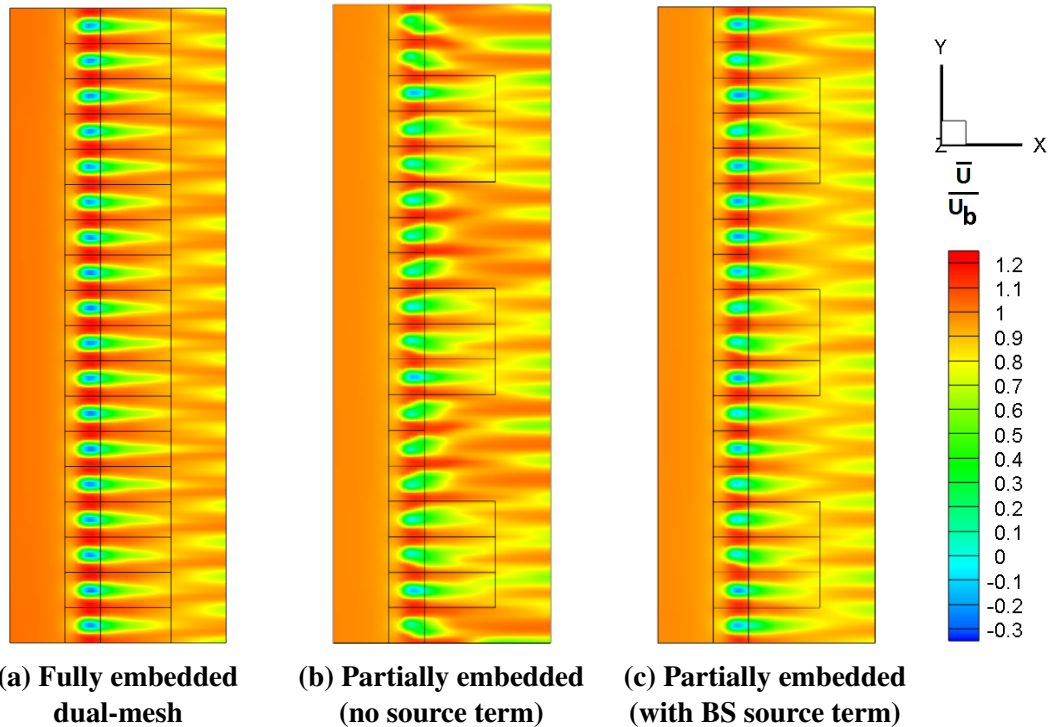
Again, local high values are caused by the lack of an infinite series of time sampling.

Regarding the computational resources utilised, performance metrics are presented in

Table 6.3. C_{map} is 1.02 and the computational gain c_e is 1.30.

Table 6.3 Performance metrics for the BS mapping in the lateral direction (1 core utilised). Metrics of the fully embedded configuration are used as reference values

ratio	Domain size	Time per iteration
fully embedded	1	1
partially embedded (no source term)	0.76	0.93
partially embedded (with BS source term)	0.76	0.95

Figure 6.15 Contours of normalised time-averaged freestream velocity at $Z = 3/5\ell$ Figure 6.16 Contours of normalised time-averaged freestream velocity at $Z = 6/5\ell$

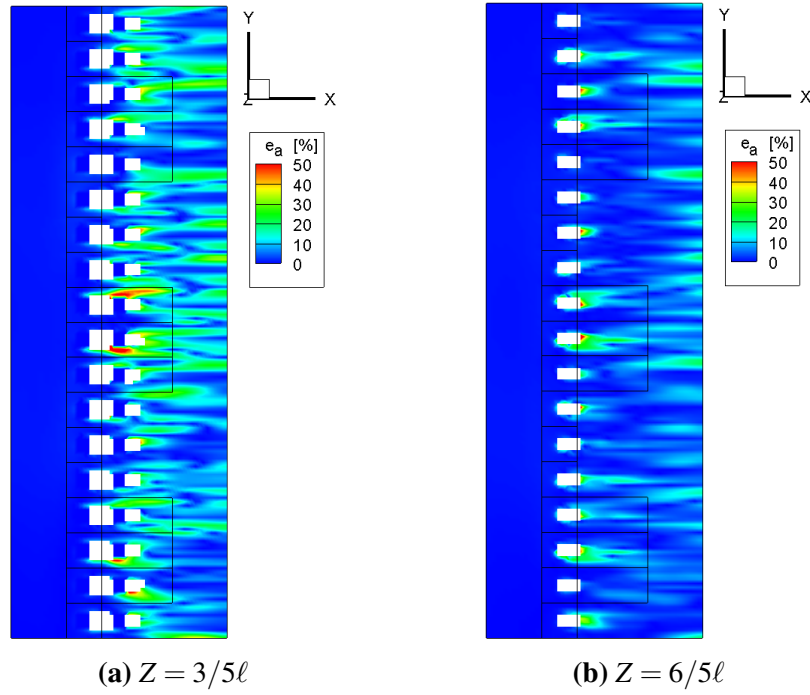


Figure 6.17 Relative error (e_a) at two X-Y planes when BS mapping in the lateral direction is applied. Blank regions correspond to $\bar{U} \approx 0$

6.1.3 Block Spectral Mapping Applied in Both (X-Y) Directions

In the present subsection the micro-structures considered are shorter in height ($h = 0.6\ell$), while the absolute magnitudes of the remaining geometric and pattern properties are unchanged. Table 6.4 summarises the relative physical properties of the micro-structures and their pattern, which are illustrated in Figure 6.18.

Table 6.4 Mapping in both directions: Geometric and pattern properties of the micro-structures

B_{st}/h	B_{sp}/h	λ_{st}/h	λ_{sp}/h
3.3	3.3	10	10

In Figure 6.19 the distribution of the micro-structures and the embedded mesh blocks configuration for the partially embedded without source terms and the partially embedded with BS source terms setup are illustrated. The distribution of the micro-structures is set to be 21 X-rows and 19 Y-columns (399 in total), with the spacing between them in both directions

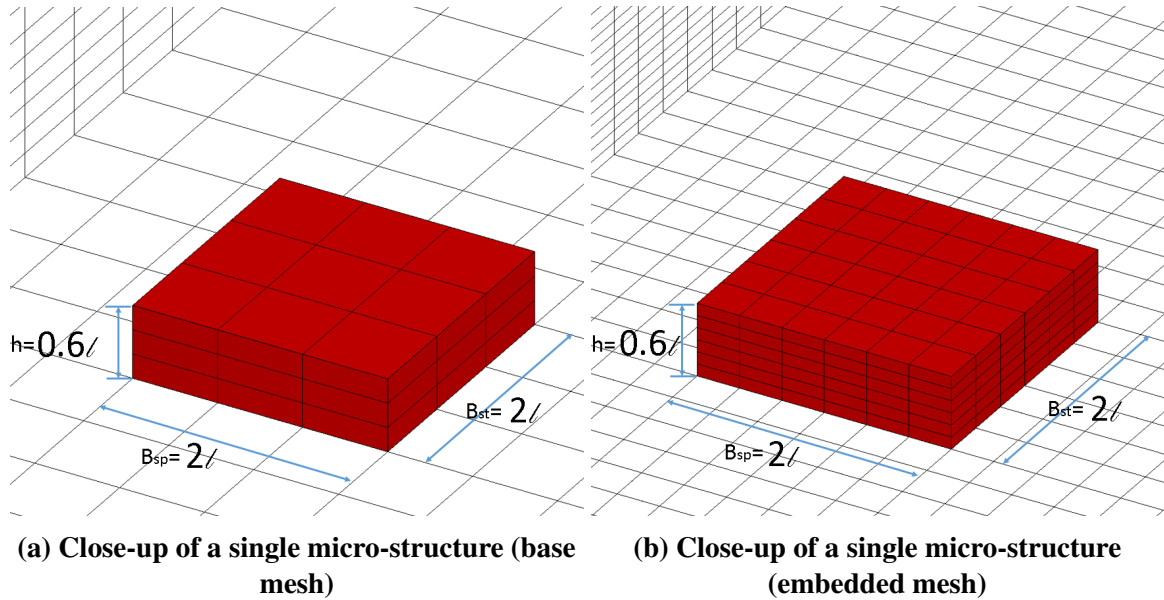


Figure 6.18 Mapping in both directions: Mesh distribution on and around the micro-structures

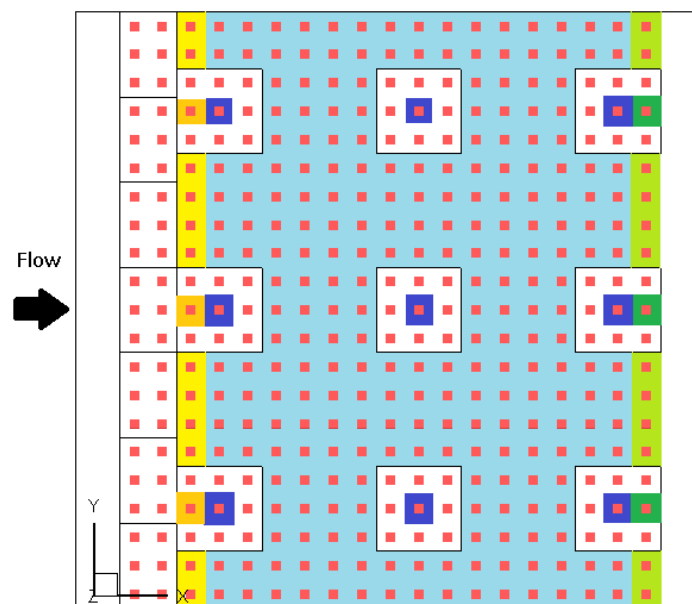
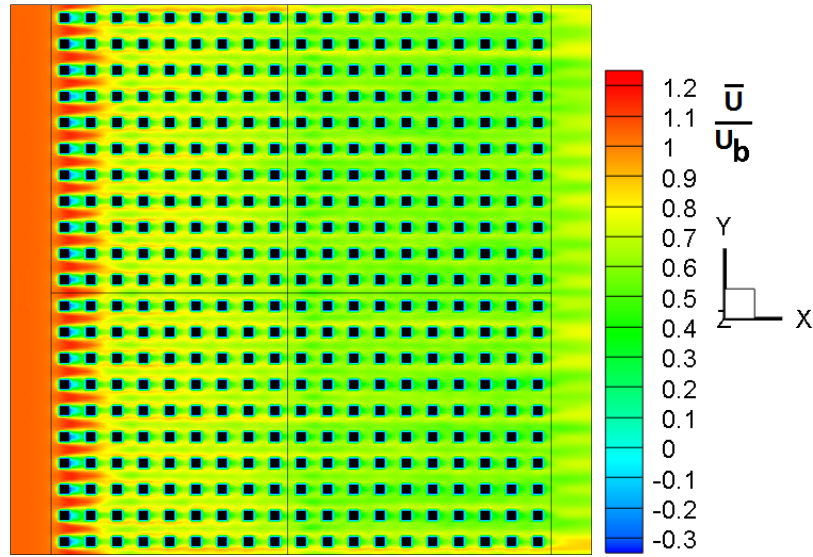


Figure 6.19 Top view of the domain for the partially embedded configuration. One red square corresponds to one micro-structure, while dark edges mark the boundaries of the embedded mesh blocks. Each other colour indicates a group of BS (light for mapped region, dark for source regions)

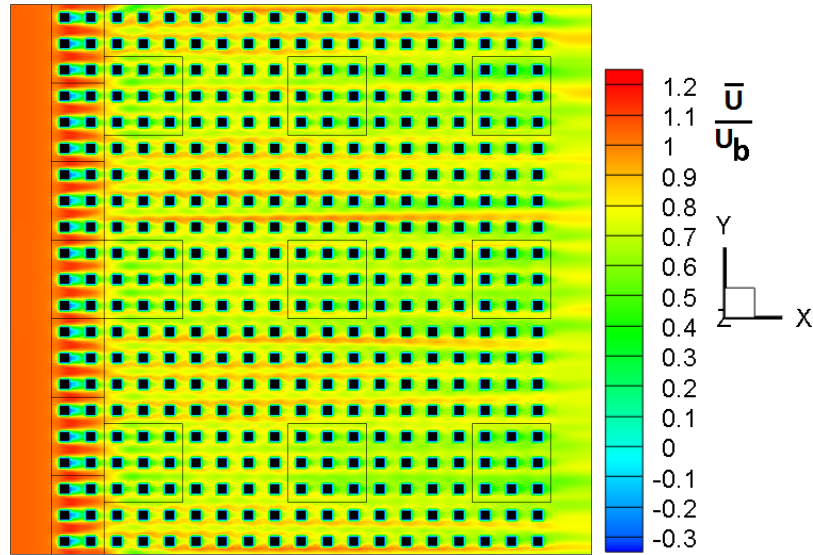
being the same as before. For simplicity, the embedded blocks cover a larger area, spanning across 6 or 9 micro-structures for the partially embedded setup. For the BS simulation, 3 groups of spectra are selected, as presented in Figure 6.19. The first, second and third groups are represented in yellow, blue and green, respectively. The first and third BS groups are applied only in the lateral direction, whereas the second group maps information in both directions. Eventually, the whole non-embedded region is mapped. The source regions are located within the refined by the fine mesh regions, surrounded by an embedded domain of 1 micro-structure long in each direction. One and two harmonics are chosen for the lateral and the freestream directions, respectively. Thus, 3 source regions are defined for each direction (for the periodic and the non-periodic spectra in the lateral and freestream direction, respectively).

The time-averaged freestream velocity contours at two X-Y planes are shown in Figures 6.20 and 6.21. The partially embedded without source terms configuration over-predicts the freestream velocity magnitudes due to the poor resolution of the flow displacement by the coarse mesh domain that is not updated by an embedded mesh. On the contrary, good agreement between the fully embedded and the partially embedded with BS source terms is observed. The relative error of the time-averaged axial velocity at the $Z = 3/5\ell$ and $Z = 4/5\ell$ is shown in Figure 6.22, with the area-averaged error being equal to 11.7% and 13.9%, respectively.

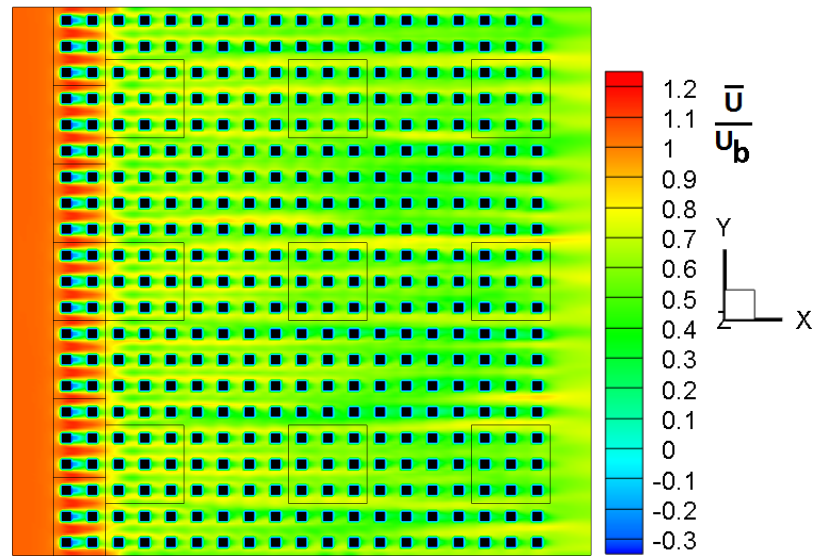
Regarding the computational resources utilised, performance metrics are presented in Table 6.5. C_{map} is 1.08 and the computational gain c_e is 2.10.



(a) Fully embedded dual-mesh

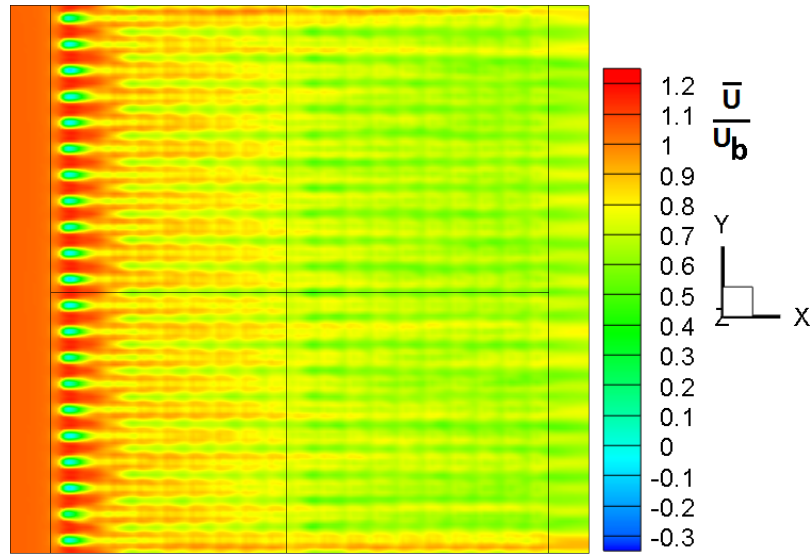


(b) Partially embedded (no source term)

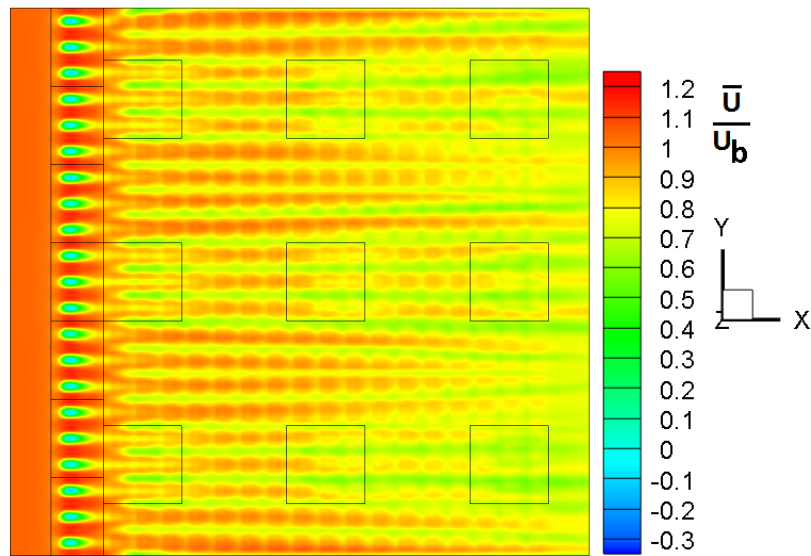


(c) Partially embedded (with BS source term)

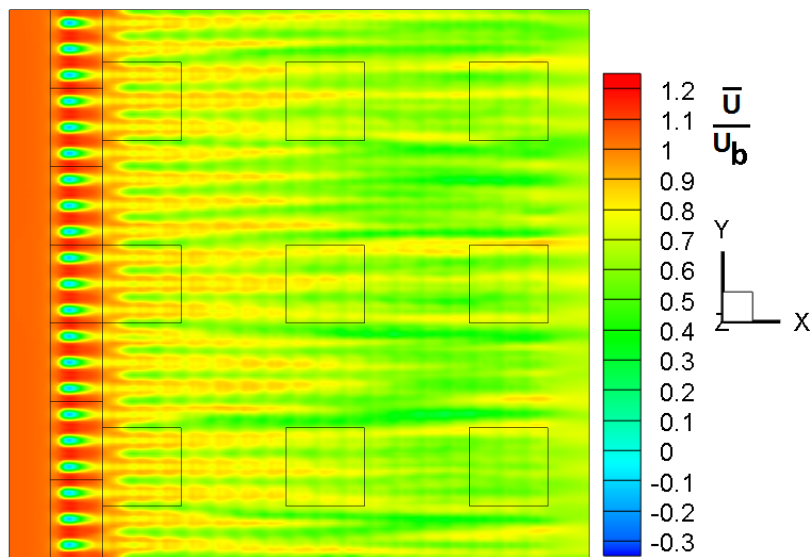
Figure 6.20 Contours of normalised time-averaged freestream velocity at $Z = 3/5\ell$. Internal edges reflect the boundaries of the embedded blocks



(a) Fully embedded dual-mesh



(b) Partially embedded (no source term)



(c) Partially embedded (with BS source term)

Figure 6.21 Contours of normalised time-averaged freestream velocity at $Z = 4/5\ell$. Internal edges reflect of the boundaries of the embedded blocks

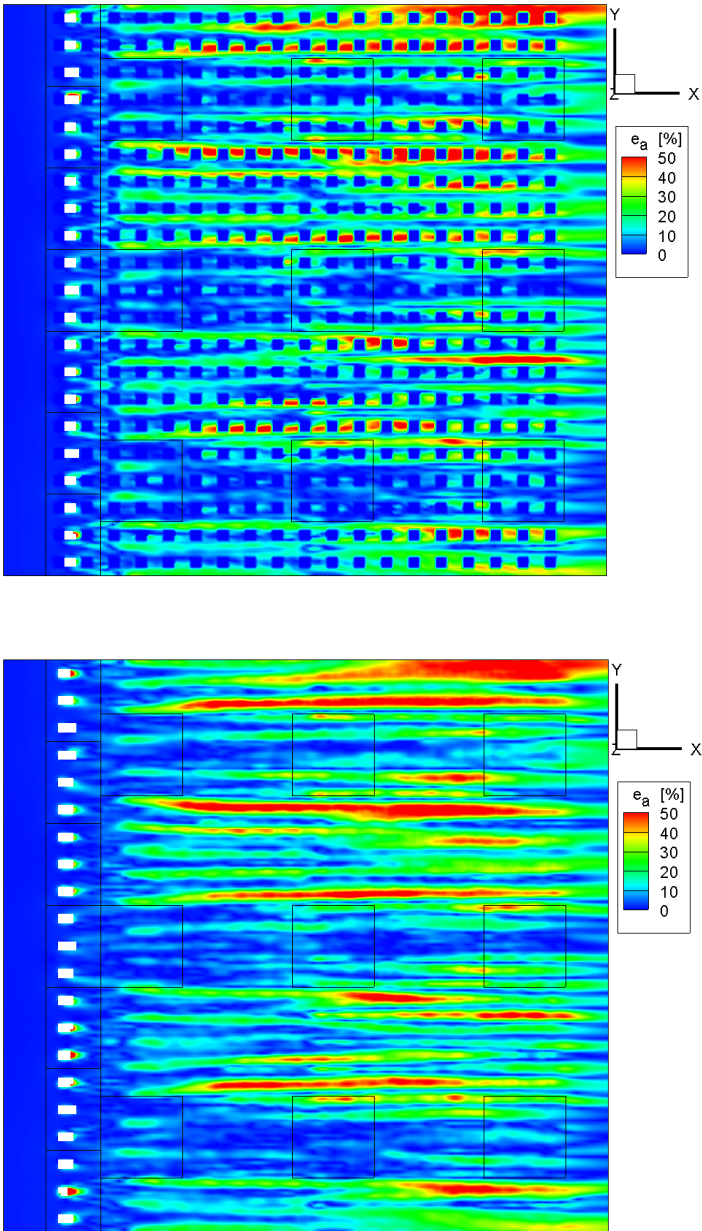


Figure 6.22 Relative error (e_a) at two X-Y planes when BS mapping in both directions is applied. Blank regions correspond to $\overline{U} \approx 0$

Table 6.5 Performance metrics for the BS mapping in the lateral direction. Metrics of the fully embedded configuration are used as reference values

ratio	Domain size	Time per iteration
fully embedded [5 cores]	1	1
partially embedded (no source term) [3 cores]	0.44	0.86
partially embedded (with BS source term) [3 cores]	0.44	0.93

6.2 Heat Transfer Cases

6.2.1 Trapezoidal Ribs

In this section the Block Spectral Method is applied to a surface fitted with iso-thermal trapezoidal ribs. The purpose of the presented case study is to demonstrate that the BSM can reproduce the correct time-averaged heat transfer at the viscous walls of the embedded domain, as if the whole near-wall region was refined by an embedded mesh. The satisfactory matching of the heat transfer predicted by the fully embedded and the partially embedded with BS source terms will indicate the accurate capturing of the upstream history effect.

The case study consists of 20 trapezoidal ribs, shown in Figures 6.23 and 6.24, distributed in an array pointing in the streamwise direction. Similarly to the previous section, the

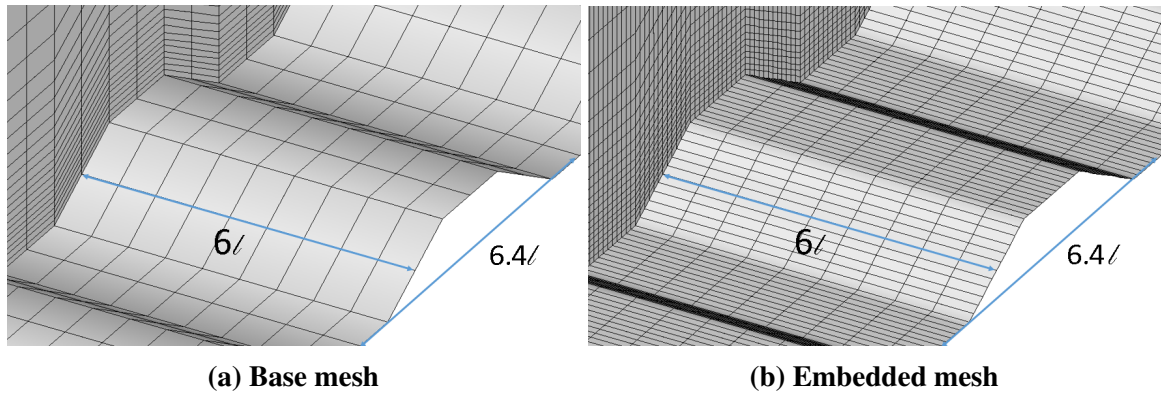


Figure 6.23 3D view of ribs

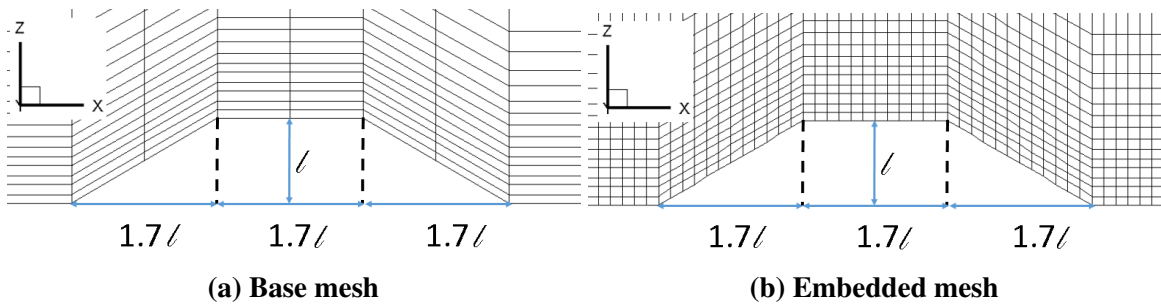


Figure 6.24 Side view of the ribs

geometry is defined based on the characteristic length $\ell = 0.5\text{mm}$. The height of the ribs

is $h = \ell$, their streamwise and spanwise dimension is $B_{st,sp} = 5.1\ell$ and the spacing in the streamwise direction is equal to 6.4ℓ . The height of the computational domain is $\approx 50\ell$ and its width is 6ℓ . The employed domain configuration is similar to the configuration for two harmonics of the subsection 8.1.1. In the present case, the coarse domain is refined by fine mesh regions which are joined together when they are adjacent, e.g. for the fully embedded configuration, the embedded domain consists of a single fine mesh block. The ribs consist of 6×9 coarse mesh cells in the X and Y direction, respectively. The embedded domain refines the coarse mesh domain only in the streamwise direction by 6 times.

The bulk flow properties are matched to those of the previous section ($Re_\ell \approx 2600$, $U_b \approx 57m/s$, $Ma_b \approx 0.17$), under which the flow transitions into unsteadiness, as presented in Figure 6.25. The viscous walls are iso-thermal with $T_{wall} = 1.11T_{0,in}$. In Figures 6.26 and 6.27 the time-averaged velocity, temperature and heat transfer are presented for the coarse and fine domain, respectively. The plotting lines are selected as follows:

- For the bulk flow properties $\bar{T}$ and $\bar{U}$, the reference line is located at the midspan at $Z = 7/5\ell$ (40% h above the ribs).
- For the $\bar{Nu}$ results, the data are obtained from the intersection line of the bottom wall and the midspan plane.

For the results of the coarse (base) mesh (Figure 6.26), good matching between the fully embedded and the partially embedded with source terms is observed. The partially embedded without source terms under-predicts the time-averaged velocity and heat transfer, and only when the Block Spectral Method is applied the partial mesh embedding generates the correct trends. More importantly, the heat transfer of the fine mesh (Figure 6.27c) of the fully embedded configuration is predicted by the BSM with excellent matching, with a clear contrast to the predictions of the partially embedded without source terms configuration. In

addition, the differences in the predicted heat transfer between the two partial embedded configurations are shown in Figures 6.26b and 6.27b where the predicted trends of $\bar{T}$ by the partially embedded without source terms are incorrect. The relative error e_a of the $\bar{Nu}$ is 8.0% and 7.6% for the base and the embedded domain, respectively.

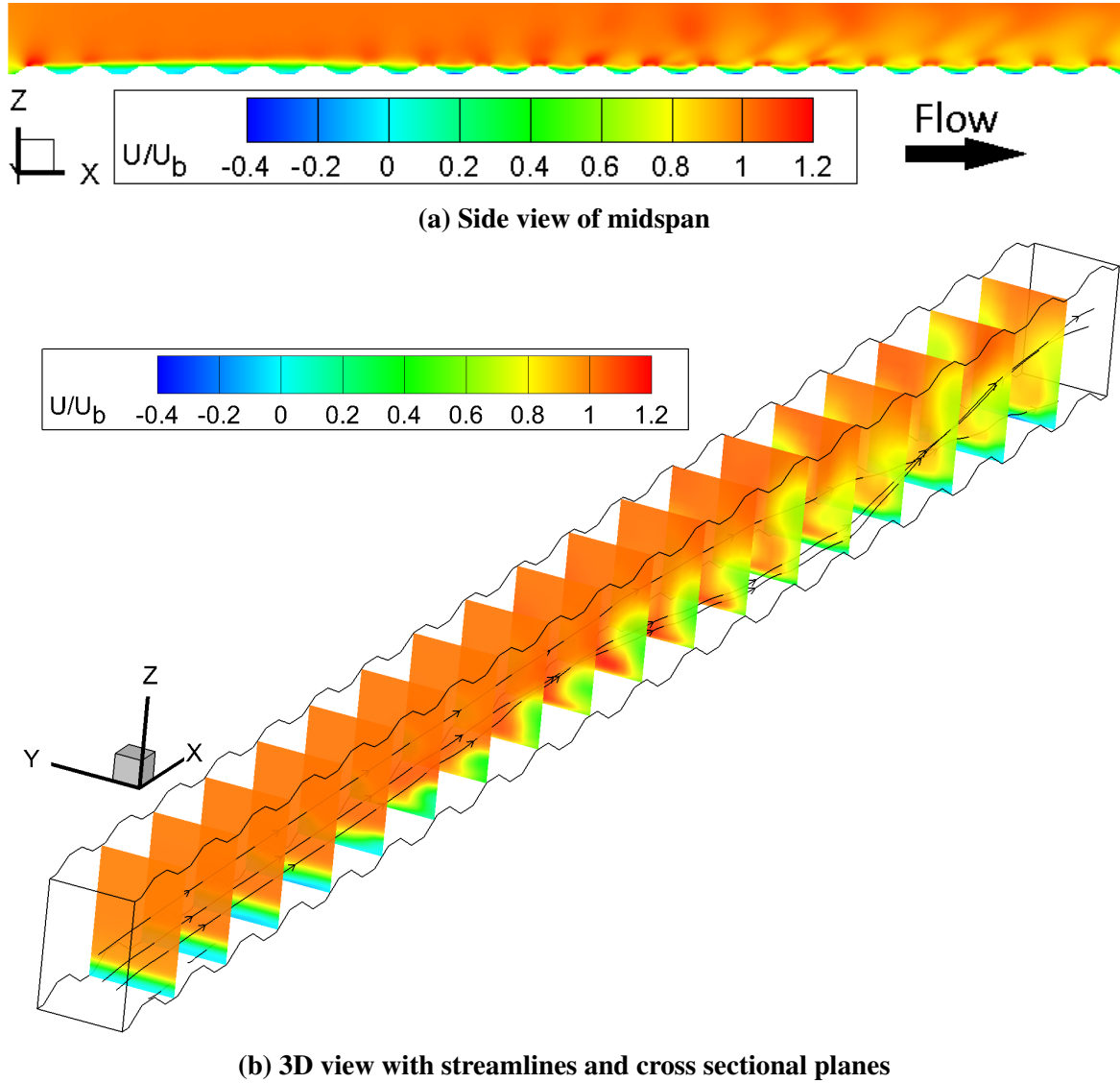
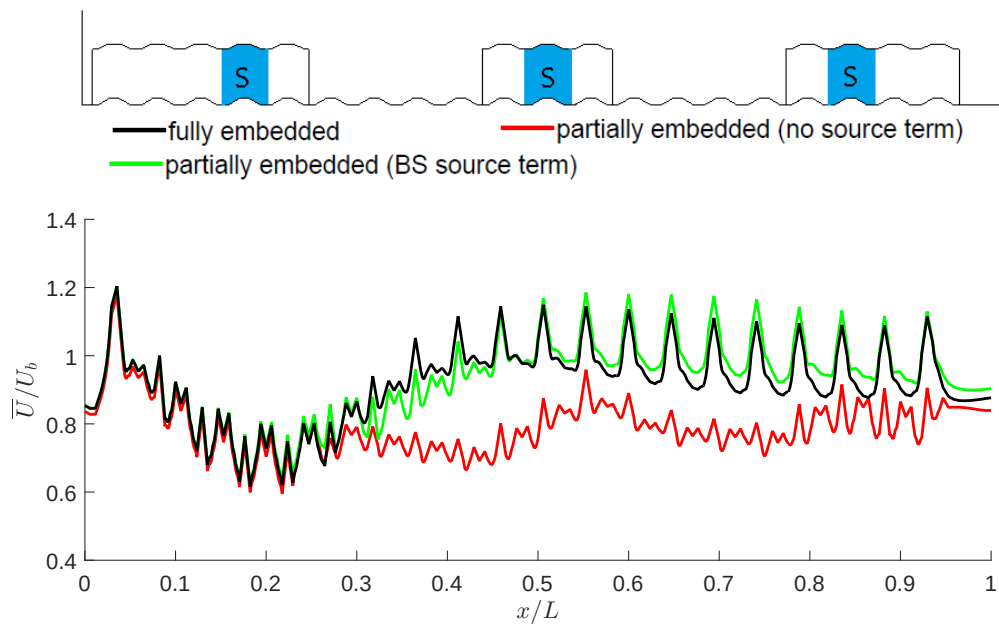
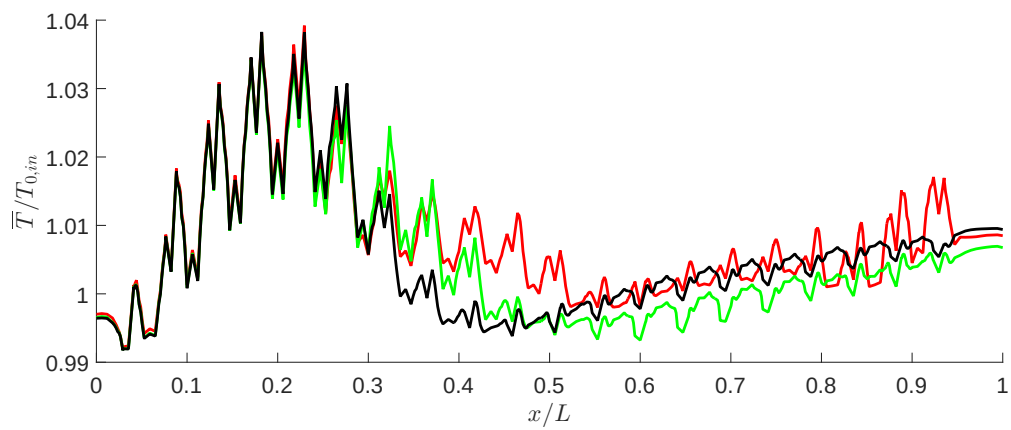


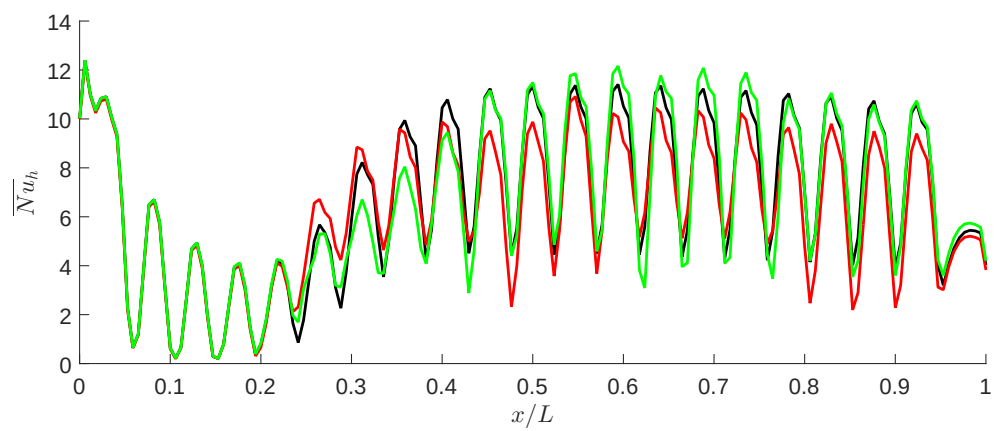
Figure 6.25 Instantaneous normalised freestream velocity of the embedded domain



(a) Normalised freestream velocity

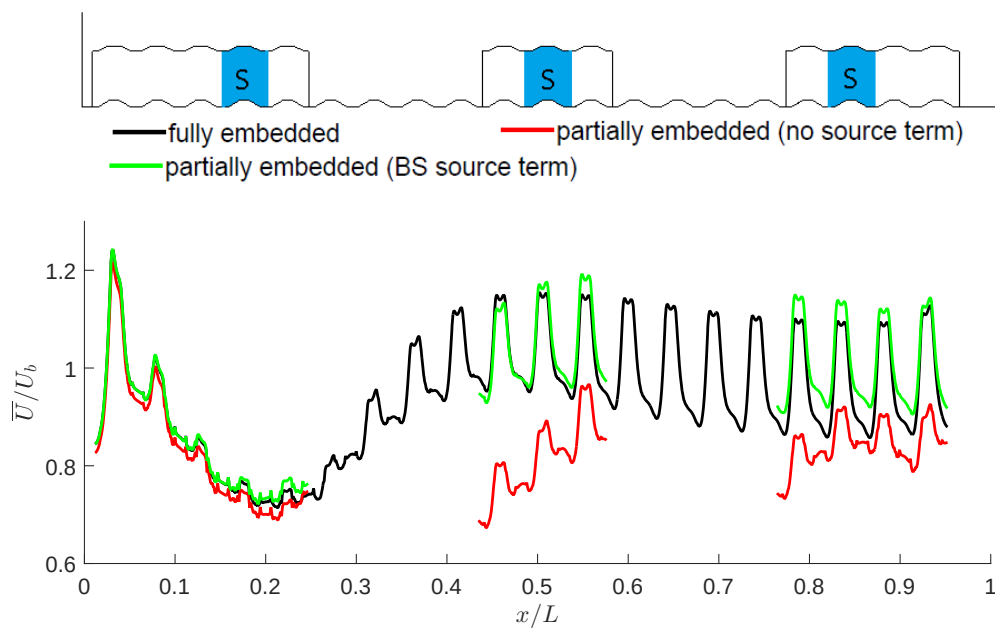


(b) Normalised static temperature

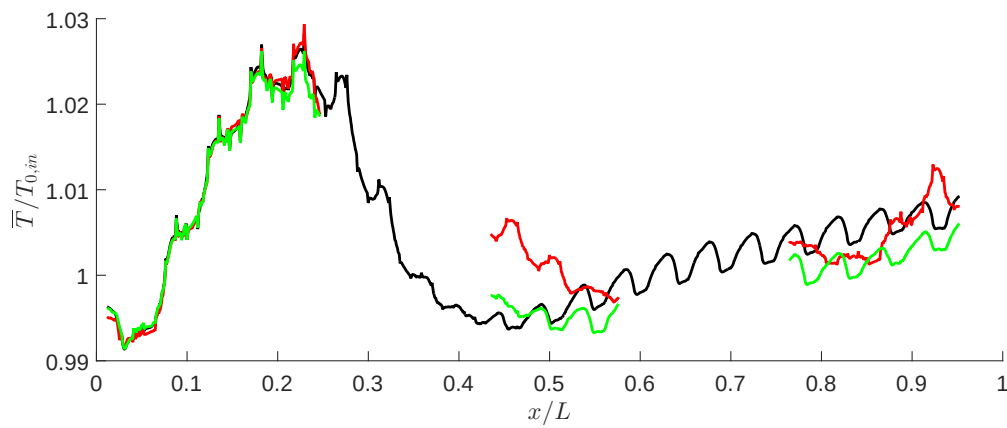


(c) Heat transfer

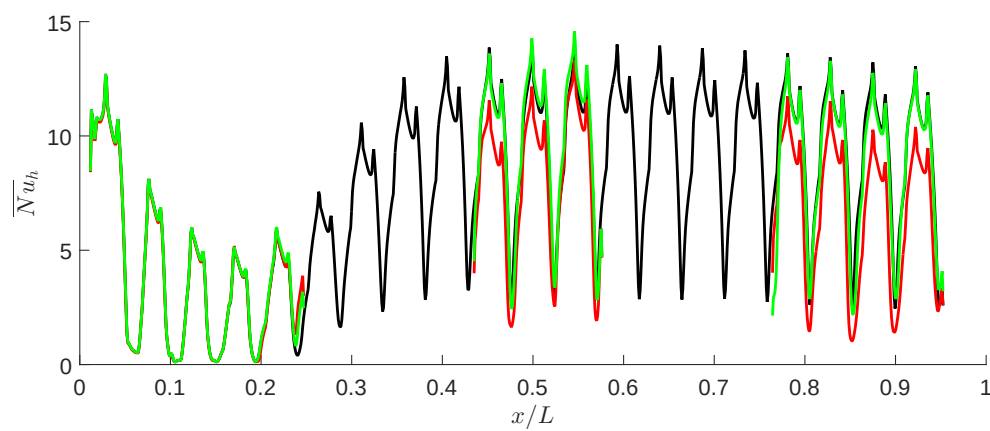
Figure 6.26 Base domain time-averaged solution



(a) Normalised freestream velocity



(b) Normalised static temperature



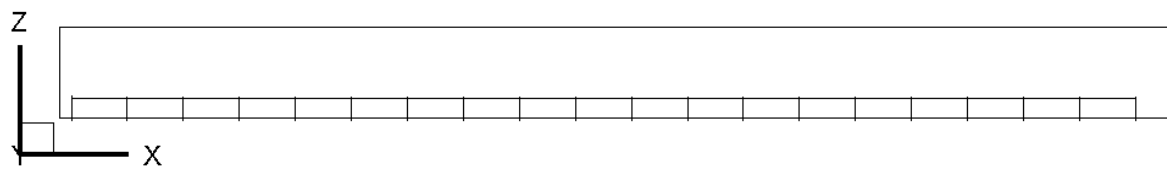
(c) Heat transfer

Figure 6.27 Embedded domain time-averaged solution

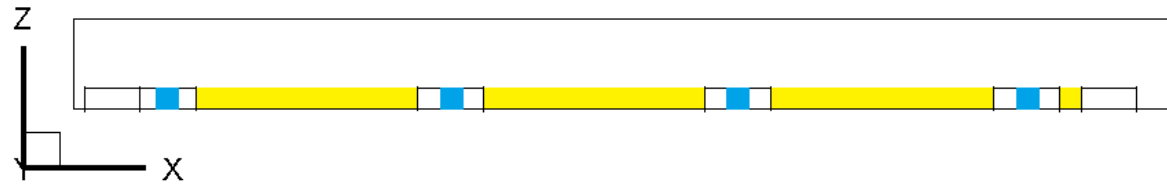
6.2.2 Micro-Structures in Channel Flow

Furthermore, a case is investigated which considers the micro-structures presented in Section 5.4 (channel flow with bisected pimples and dimples). The geometric characteristics of the micro-structures are the same as discussed in the related section. The length of the computational domain is slightly increased, reaching $12 \times D_h$ and the channel consists of three inviscid walls and one viscous wall. The viscous wall is fitted with multiple rows of micro-structures, where each row consists of 7 micro-structures. The coarse domain is represented by one base block and local refinement is performed at the vicinity of the viscous wall. The domain configuration can be seen in Figure 6.28. Two different CFD configurations are considered here: the fully embedded, where the embedded domain consists of 19 mesh blocks, and the partially embedded with BS source terms, where the embedded domain consists of 6 mesh blocks. Each micro-structure is resolved with 4×4 cells by the coarse mesh. The embedded domain refines the coarse mesh by $8 \times 8 \times 1$ as shown in Figure 6.29 (refined in the X and Y direction only). The domain configuration suggests that the BSM can only be applied in the freestream (X) direction. Three harmonics are utilised with mirrored source regions, thus four source regions need to be defined.

In Figure 6.30 the area-averaged, time-averaged Nu (as defined in Section 5.4) is presented, where good matching between the fully embedded and the partially embedded with BS source terms configurations is observed. The maximum relative error e_a is 9.6% at $x/D_h = 10.5$. Also, performance metrics are presented in Table 6.6 where C_{map} and c_e are 1.05 and 2.9, respectively.



(a) Fully embedded configuration



(b) Partially embedded configuration. Blue and yellow regions denote the source and the mapped regions, respectively

Figure 6.28 Side view of the computational domain. Inner edges denote the boundaries of the embedded mesh blocks

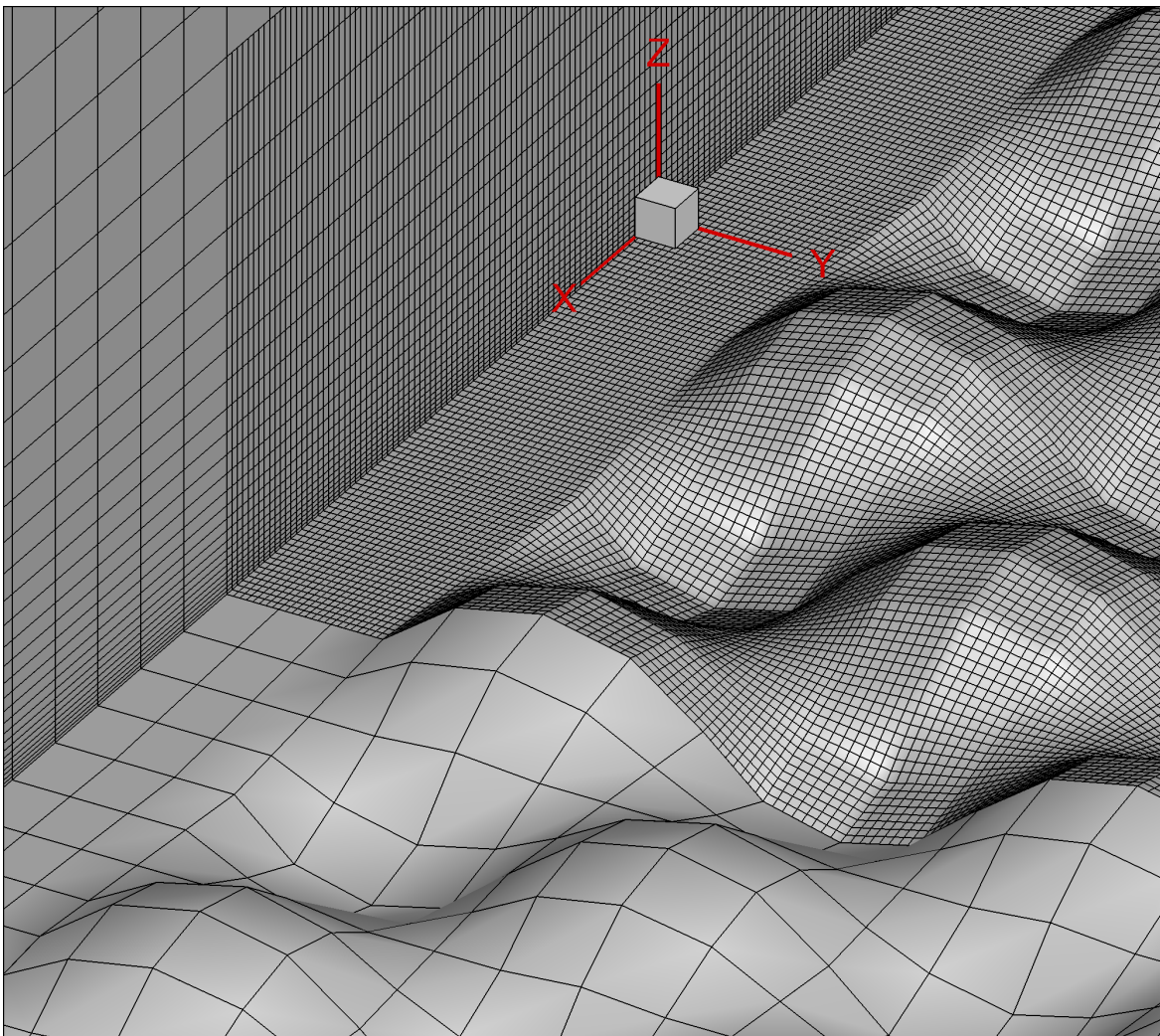


Figure 6.29 Close-up at of the coarse and embedded mesh

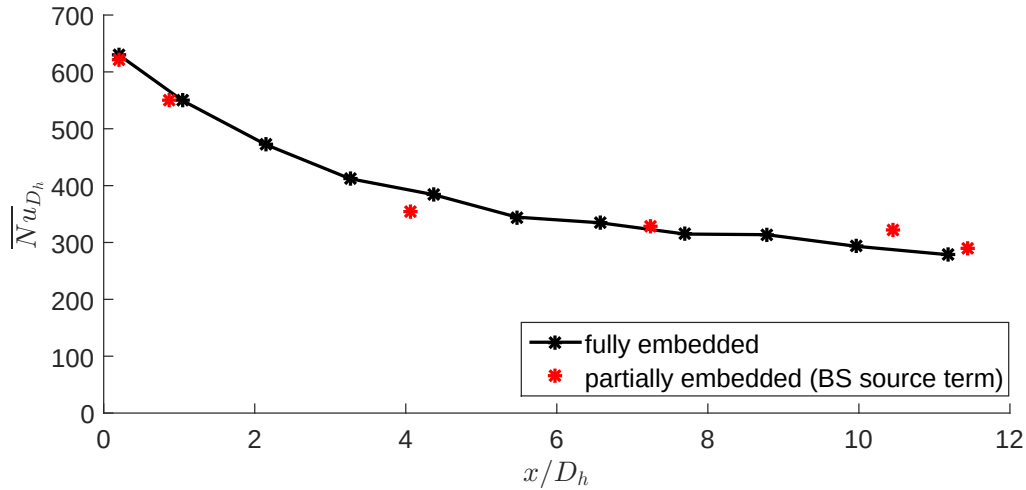


Figure 6.30 Comparison of area-averaged, time-averaged Nu of the embedded domain between fully embedded and the partially embedded with BS source terms configurations. Points are positioned at the centres of each embedded mesh block

Table 6.6 Performance metrics. Metrics of the fully embedded configuration are used as reference values

ratio	Domain size	Time per iteration
fully embedded [20 cores]	1	1
partially embedded (no source term) [7 cores]	0.33	0.82
partially embedded (with BS source term) [7 cores]	0.33	0.86

6.3 Block Spectral Method Scalability

In this section a domain with ~ 4000 micro-structures is considered in order to demonstrate the computational gain of the BS solver and the potential of the BSM regarding the sensible consumption of computational resources.

The large domain, shown in Figure 6.31, is created by simply extending the domain used in the subsection 8.1.3 (case study for BS mapping in both directions). The domain consists of 105 rows in the lateral direction and 38 columns in the freestream direction, resulting in a total of 3990 micro-structures. The base coarse mesh domain is split into 14 mesh blocks of 15 rows and 19 columns each, with each base mesh block consisting of $\approx 725,000$ cells.

The embedded mesh refines the coarse domain $5 \times 5 \times 5$ times. The total mesh size is $\approx 68 \times 10^6$ cells, with the fine domain accounting for the 85% of the total mesh size. One and three harmonics were considered for the BS mapping in the lateral and in the freestream direction, respectively.

Due to the high cell-count of the base domain, the 14 coarse mesh blocks are distributed among 7 cores, i.e. the calculation and distribution of the Fourier coefficients will occur between these 7 cores. Note that only $\approx 8\%$ of the total number of micro-structures is refined by the embedded domain, as a fully embedded dual-mesh configuration would create a mesh of $\approx 700 \times 10^6$ cells. Regarding the computational gain, as described in subsection 4.3.4 and expressed in Equation 4.31, for $M_{full} = 3990$ and $N_{extra} \times M_{BS} = 318 \approx 8\%M_{full}$ the ideal value is $c_e \sim 12.5$. Considering the measured relatively small time overhead for the BSM of $C_{map} \approx 1.15$, the present computational gain is $c_e \approx 10.9$.

6.4 Remarks

In summary, the validation of the Block Spectral Method study has been conducted considering mapping in the streamwise direction and lateral direction respectively, as well as in both directions. The analyses have demonstrated that the Block Spectral Method could potentially be used for predicting aerothermal impact of 3D micro-structures. Time-averaged solutions generated by the BSM have adequately and in some cases strongly matched those generated by fully embedded configurations.

Firstly, the BSM was applied in the freestream direction, in a domain consisting of rectangular micro-structures. The partially embedded with source terms solution matched the fully embedded solution, with improved matching as the number of harmonics increased, while the partially embedded without source terms demonstrated erroneous predictions.

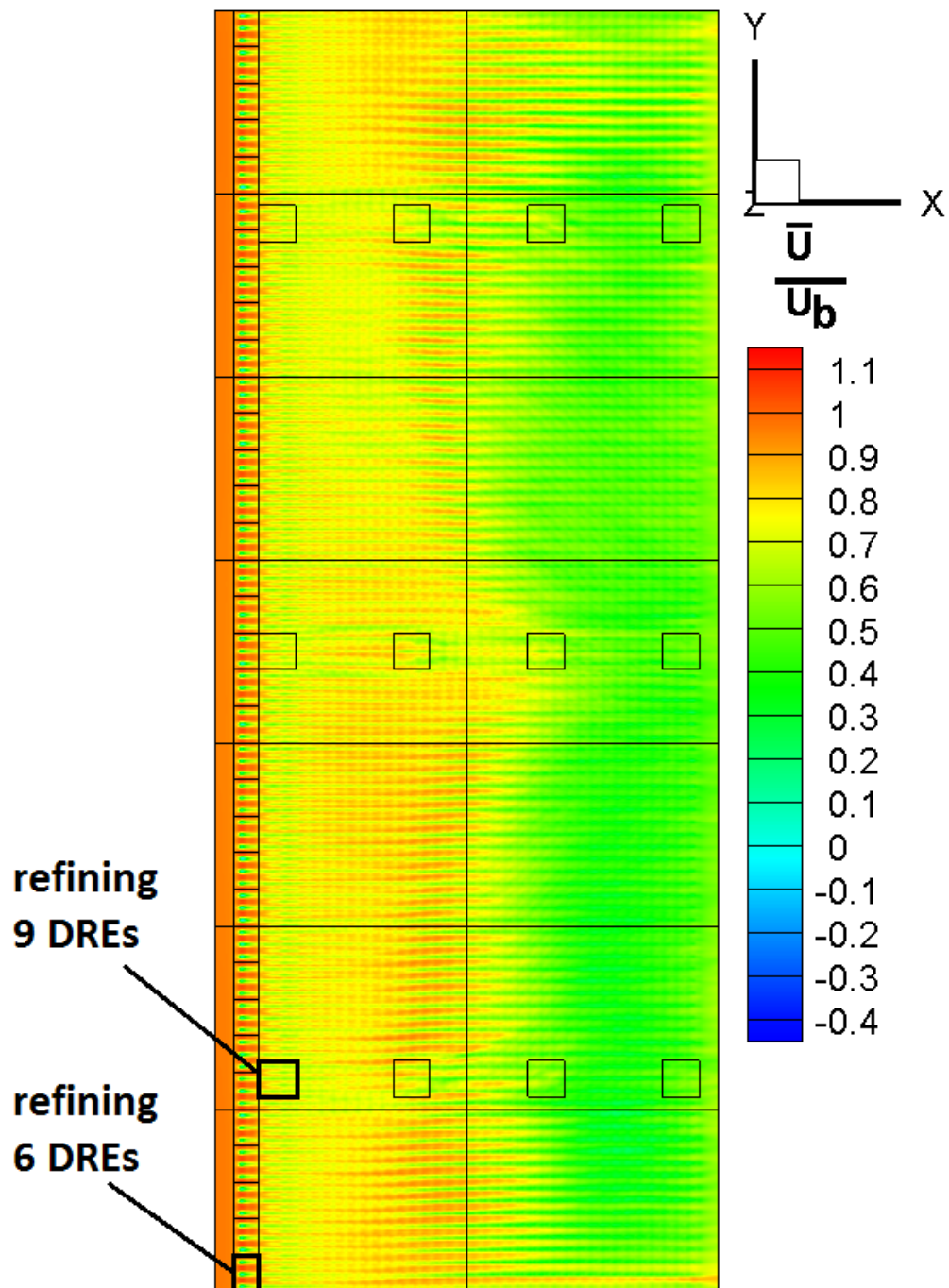


Figure 6.31 Scalability case: Contour of time-averaged freestream velocity at $Z = 4/5\ell$. Edges denote mesh block boundaries

Additionally, the mapping of the TST in the lateral direction was tested, matching well the fully embedded time-averaged solution. Following this, the 2D mapping of the BS source terms was tested on a domain with ~ 400 micro-structures, matching adequately the fully embedded solution.

Secondly, the heat transfer prediction capability of the BSM was investigated. Initially, trapezoidal ribs distributed in the streamwise direction were considered. The time-averaged Nu of the embedded domain closely matched the values predicted by the fully embedded configuration. In addition, a channel flow with pimples and dimples, which were also studied in the previous chapter, was presented where the heat transfer prediction of the fully embedded and the partially embedded with BS source terms configurations were matched.

Finally, the BSM was utilised for a test case consisting of ~ 4000 micro-structures, as a demonstration of the scaling capabilities of the BSM, where the equivalent fully embedded configuration would require ≈ 12.5 times more computational memory. The computational overhead of the BS calculations corresponded to the $\approx 13\%$ of the ideal computational gain, resulting in a satisfactory actual computational gain.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

Additive manufacturing has introduced great potential for designs at significantly fine scales, resulting in manufacturable surface micro-structures (regular roughness) potentially being included in the component design space. These emerged capabilities require more accurate predictive methods; methods that are lacking from the state of the art conventional CFD approaches. Thus, the question arises whether the shape and pattern of the regular roughness affect aerothermal performance. If they do, then approaches of capturing such fine scales efficiently should be sought.

Roughness can demonstrate positive aerothermal effects, dependent on the bulk flow environment, and the collective effect of the roughness features has been conventionally predicted by stochastic parameters. However, when it comes to regular roughness, recent investigations have implied that shape and pattern characterise the roughness aerothermal behaviour. Despite these investigations, current CFD approaches of introducing roughness effects still rely on empirical or semi-empirical modelling. Moreover, commonly used RANS methods lack the ability of accurately capturing flow separation and reattachment, a key trait

of the bluff shape of the micro-structures. High local refinement could be resolved by a multi-scale approach, but such techniques have not been widely tested on manufacturable roughness.

In the third chapter, the shape and pattern effects of the micro-structures in the relevant flow regime were studied. The case studies consisted of single arrays of few micro-structures of either rectangular or hemispherical shape. Initially, a recent experimental study, which investigated the impact of roughness on the onset of the rapid drag rise, was numerically verified. Moreover, the spacing effect was investigated in the fully turbulent regime using RANS. Additionally, the shape effect was investigated in both the fully turbulent (RANS) and transitional regimes (ILES). For both flow regimes a similar flow mechanism that caused the inception of rapid drag rise was identified: the separated shear layers shed by each micro-structure would cause higher drag increase when they would impinge to the downstream roughness element. In general, the shape and pattern of the micro-structures was found to have a substantial aerothermal effect, even when conventional parameters for the characterisation of stochastic roughness were matched.

In Chapter 4 the available serial solver and its characteristics were discussed. Given the requirement for high near-wall resolution, the embedded dual-mesh approach was presented. The problem is split into a two-scale domain configuration: one domain for the near-wall region capturing the fine scales and a domain including both the near-wall region and the bulk flow, adequately resolving the bulk flow structures. The two domains are coupled through interpolation of the coarse mesh flow properties and spatial-averaging of the fine mesh properties. For larger domains, the Block Spectral Method was considered. Discrete sub-regions of the near-wall regions are treated as a dual-mesh configuration and their effect is mapped on the near-wall coarse domain. The coarse mesh correction is performed by

adding source terms, named Total Source Terms, in the governing equations. For the mapping of the Total Stress Terms, the Fourier series were selected as a basis function due to their fast convergence, i.e. the required number of source regions, and therefore the required number of coarse mesh regions that are refined by embedded mesh blocks, is low.

The validation of the baseline solver was discussed in Chapter 5. Initially, the solver accuracy was tested on the 2D flat plate and 2D cylinder cases. The solver solutions matched analytical and experimental data, as well as the predictions of the commercial software FLUENT. Also, the dual-mesh interface treatment was tested by comparing a dual-mesh configuration against the prediction of an equivalent multi-block setup. The main case of the chapter consisted of a channel flow with a surface fitted with pimples and dimples. The ILES matched well with the experimental results, and the hybrid ILES, for which the boundary layer of the top plain surface was tripped using locally the SA modelling, generated the most accurate pressure drop and heat transfer predictions. On the contrary, the full domain RANS configuration failed to adequately predict the heat transfer on the surface fitted with micro-structures. The flow acceleration and impingement were identified as the primary causes of the heat transfer augmentation on the pimple. Also, the heat transfer of the dimples was linked to the vortical structures generated by the upstream flow, which enters the dimples, and the flow impingement on the downstream edge of the dimple. Most importantly, the chaotic behaviour of the unsteady flow structures generate an ordered, smoothly varied time-averaged flow field which is vital for the application of the Block Spectral Method.

Finally, in Chapter 6 the Block Spectral Method prediction capabilities for test cases with 3D micro-structures were studied. The investigation consisted of cases considering the Block Spectral source mapping in one direction (axial or lateral) as well as in both directions, consisting of rectangular micro-structures subject to a transitional flow regime. Additionally,

two test cases with heat transfer on an iso-thermal wall were demonstrated. The first one considered a surface fitted with trapezoidal ribs, while the latter investigated a channel flow with pimples and dimples. In all cases, the partially embedded solutions without source terms demonstrated poor aerothermal prediction when compared to the fully embedded solutions, while the partially embedded with BS source terms configurations generated time-averaged flow properties that matched those of the fully embedded configurations. In the end, the method was applied on a large case of which the equivalent fully embedded configuration would require impractical amounts of computational resources. Essentially, the Block Spectral Method was shown to generate accurate solutions for self-excited flows around 3D micro-structures, becoming a strong candidate as a tool for regular roughness aerothermal predictions.

7.2 Future Work

Stochastic roughness, either conventional or resident roughness from the AM process, aerothermal prediction is another potential application that could make use of the BSM. In order for the BSM to be applicable, two key requirements should be addressed:

1. The random roughness has to be characterised by regions with spatially similar geometric properties.
2. The BSM is linked to the pointwise propagation of the Total Source Terms, which is practically performed through a cellwise mapping. Thus, the mesh over the randomly rough surface has to be consistently generated, ensuring similar mesh topologies among all regions for the pointwise mapping of the source terms to be applied.

Regarding the heat transfer of the micro-structures, the bluff body property of common shapes could allow for conduction to have a more prominent role within the roughness

elements. Conjugate heat transfer could be employed to accurately resolve the relevant scales and the interplay between the flow and the solid domain. Although this would generate a three-scale problem, introducing further complexity to the coupling of the domains, conjugate heat transfer could provide a more insightful understanding of the mechanisms affecting the heat transfer of regular roughness.

References

- [1] Huang, Y., Leu, M. C., Mazumder, J., and Donmez, A., 2014. “Additive Manufacturing: Current State, Future Potential, Gaps and Needs, and Recommendations”. *Journal of Manufacturing Science and Engineering*, **137**(1), p. 014001.
- [2] Gao, W., Zhang, Y., Ramanujan, D., Ramani, K., Chen, Y., Williams, C. B., Wang, C. C., Shin, Y. C., Zhang, S., and Zavattieri, P. D., 2015. “The Status, Challenges, and Future of Additive Manufacturing in Engineering”. *Computer-Aided Design*, **69**, pp. 65–89.
- [3] Meissner, C., 2012. Materials by Design. Article from Lawrence Livermore National Laboratory, accessed Feb 2018. <https://str.llnl.gov/Mar12/spadaccini.html>.
- [4] Abrams, M., 2015. Jet Printing in 3D. Accessed on 18 June 2015. <https://www.asme.org/engineering-topics/articles/manufacturing-processing/jet-printing-in-3d>.
- [5] Severson, B., 2014. GE Considers 3D Printing Turbine Blades for Next Generation Boeing 777Xs GE9X Engines. Accessed on 28 July 2015. <http://3dprint.com/11266/3d-printed-lpt-ge9x-777x/>.
- [6] Kellner, T., 2017. Mind Meld: How GE And A 3D-Printing Visionary Joined Forces. Accessed on 6 July 2018. <https://www.ge.com/reports/mind-meld-ge-3d-printing-visionary-joined-forces/>.

- [7] Kellner, T., 2018. The Blade Runners: This Factory Is 3D Printing Turbine Parts For The Worlds Largest Jet Engine. Accessed on 6 July 2018. <https://www.ge.com/reports/future-manufacturing-take-look-inside-factory-3d-printing-jet-engine-parts/>.
- [8] Siebert, M., 2017. Breakthrough with 3D printed Gas Turbine Blades. Accessed on 6 July 2018. <https://www.siemens.com/innovation/en/home/pictures-of-the-future/industry-and-automation/additive-manufacturing-3d-printed-gas-turbine-blades.html>.
- [9] Vaezi, M., Seitz, H., and Yang, S., 2012. “A Review on 3D Micro-Additive Manufacturing Technologies”. *International Journal of Advanced Manufacturing Technology*, **67**, pp. 1721–1754.
- [10] Crawford, M., 2014. Taking Additive Manufacturing to the Micro Level. Accessed on 12 January 2016. <https://www.asme.org/engineering-topics/articles/manufacturing-processing/taking-additive-manufacturing-micro-level>.
- [11] Vyatskikh, A., Delalande, S., Kudo, A., Zhang, X., Portela, C. M., and Greer, J. R., 2018. “Additive Manufacturing of 3D Nano-Architected Metals”. *Nature Communications*, **9**(1), p. 593.
- [12] De Volder, M., Park, S., Tawfick, S., and Hart, A. J., 2014. “Strain-Engineered Manufacturing of Freeform Carbon Nanotube Microstructures”. *Nature Communications*, **4**(4512). <https://doi.org/10.1038/ncomms5512>.
- [13] Alexandria, P. <https://www.3dnatives.com/en/cytosurge-interview180420174/>. Accessed on 14 January 2019.
- [14] <https://www.lightmotif.nl/texturing/>. Accessed on 14 January 2019.

- [15] Yue, W., Wang, Z., Yang, Y., Chen, L., Syed, A., Wong, K., and Wang, X., 2012. “Electron-Beam Lithography of Gold Nanostructures for Surface-Enhanced Raman Scattering”. *Journal of Micromechanics and Microengineering*, **22**, p. 125007.
- [16] <https://researchmicrostamps.com/gallery/>. Accessed on 14 January 2019.
- [17] <https://www.nanoscribe.de>. Accessed on 14 January 2019.
- [18] Ford, S. L. N., 2014. “Additive Manufacturing Technology: Potential Implications for U.S. Manufacturing Competitiveness”. *Journal of International Commerce and Economics*.
- [19] Schlichting, H., 1979. *Boundary Layer Theory*, 7th ed. New York.
- [20] Bons, J. P., 2010. “A Review of Surface Roughness Effects in Gas Turbines”. *ASME Journal of Turbomachinery*, **132**(2), p. 021004.
- [21] Speidel, L., 1962. “Determination of the Necessary Surface Quality and Possible Losses Due to Roughness in Steam Turbines”. *Elektrizitwirtschaft*, **61**(21), pp. 799–804.
- [22] Barlow, D. N., and Kim, Y. W., 1995. “Effect of Surface Roughness on Local Heat Transfer and Film Cooling Effectiveness”. *ASME Paper 95-GT-14*.
- [23] Hummel, F., Lotzerich, M., Cardamone, P., and Fottner, L., 2005. “Surface Roughness Effects on Turbine Blade Aerodynamics”. *ASME Journal of Turbomachinery*, **127**(3), pp. 453–461.
- [24] Bogard, D. G., Schmidt, D. L., and Tabbita, M., 1996. “Characterization and Laboratory of Turbine Airfoil Surface Roughness and Associated Heat Transfer”. *ASME Journal of Turbomachinery*, **120**(2), pp. 337–342.

- [25] Nikuradse, J., 1933. "Laws of Flow in Rough Pipes". *VDI Forschungsheft*, p. 361.
- [26] Montomoli, F., Hodson, H., and Haselbach, F., 2010. "Effect of Roughness and Unsteadiness on the Performance of a New Low Pressure Turbine Blade at Low Reynolds Numbers". *ASME Journal of Turbomachinery*, **132**(3), p. 031018.
- [27] Bons, J. P., 2002. " St and C_f augmentation for Real Turbine Roughness With Elevated Freestream Turbulence". *ASME Journal of Turbomachinery*, **124**(4), p. 632.
- [28] Bons, J. P., and McClain, S. T., 2004. "The Effect of Real Turbine Roughness with Pressure Gradient on Heat Transfer". *ASME Journal of Turbomachinery*, **126**(3), pp. 385–394.
- [29] Bons, J. P., McClain, S. T., Wang, Z. J., Chi, X., and Shih, T. I., 2008. "A Comparison of Approximate Versus Exact Geometrical Representations of Roughness for CFD Calculations of C_f and St ". *ASME Journal of Turbomachinery*, **130**(2), p. 021024.
- [30] Erickson, E. L., Ames, F. E., and Bons, J. P., 2012. "Effects of a Realistically Rough Surface on Vane Heat Transfer Including the Influence of Turbulence Condition and Reynolds Number". *ASME Journal of Turbomachinery*, **134**(2), p. 021013.
- [31] Chu, K.-H., Enright, R., and Wang, E. N., 2012. "Structured Surfaces for Enhanced Pool Boiling Heat Transfer". *Applied Physics Letters*, **100**(24).
- [32] Cardwell, N. D., Sundaram, N., and Thole, K. A., 2006. "Effect of Midpassage Gap, Endwall Misalignment, and Roughness on Endwall Film-Cooling". *ASME Journal of Turbomachinery*, **128**(1), pp. 62–70.

- [33] Goodhand, M. N., Walton, K., Blunt, L., Lung, H. W., Miller, R. J., and Mardsen, R., 2016. "The Limitations of Using R_a to Describe Surface Roughness". *ASME Journal of Turbomachinery*, **138**(10), p. 101003.
- [34] Chatzikyriakou, D., Buongiorno, J., Caviezel, D., and Lakehal, D., 2015. "DNS and LES of Turbulent Flow in a Closed Channel Featuring a Pattern of Hemispherical Roughness Elements". *International Journal of Heat & Fluid Flow*, **53**(-), pp. 29–43.
- [35] Kirsch, K. L., and Thole, K. A., 2014. "Heat Transfer Measurements of Oblong Pins". *ASME Journal of Turbomachinery*, **137**(7), p. 071004.
- [36] Kirsch, K. L., Ostanek, J. K., and Thole, K. A., 2013. "Comparison of Pin Surface Heat Transfer in Arrays of Oblong and Cylindrical Pin Fins". *ASME Journal of Turbomachinery*, **136**(4), pp. 136–145.
- [37] Siw, S. C., Chyu, M. K., and Alvin, M. A., 2012. "Heat Transfer Enhancement of Internal Cooling Passage with Triangular and Semi-Circular Shaped Pin-Fin Arrays". *13th IEEE ITherm Conference Paper No. GT2012-69266*.
- [38] Lorenz, M., Schulz, A., and Bauer, H.-J., 2012. "Experimental Study of Surface Roughness Effects on a Turbine Airfoil in a Linear Cascade - Part I: External Heat Transfer". *ASME Journal of Turbomachinery*, **134**(4), p. 041006.
- [39] Lorenz, M., Schulz, A., and Bauer, H.-J., 2012. "Experimental Study of Surface Roughness Effects on a Turbine Airfoil in a Linear Cascade - Part II: Aerodynamic Losses". *ASME Journal of Turbomachinery*, **134**(4), p. 041007.
- [40] Roberts, S. K., and Yaras, M. I., 2006. "Effects of Surface-Roughness Geometry on Separation-Bubble Transition". *ASME Journal of Turbomachinery*, **128**(2), p. 349.

- [41] Stripf, M., Schulz, A., and Wittig, S., 2005. "Surface Roughness Effects on External Heat Transfer of a HP Turbine Vane". *ASME Journal of Turbomachinery*, **127**(1), pp. 200–208.
- [42] Waigh, D. R., and Kind, R. J., 1998. "Improved Aerodynamic Characterization of Regular Three-Dimensional Roughness". *AIAA Journal*, **36**(6), pp. 1117–1119.
- [43] Sigal, A., and Danberg, J. E., 1990. "New Correlation of Roughness Density Effect on the Turbulent Boundary Layer". *AIAA Journal*, **28**(3), pp. 554–556.
- [44] Hosni, M. H., Coleman, H. W., Garner, J. W., and Taylor, R. P., 1993. "Roughness Element Shape Effects on Heat and Transfer and Skin Friction in Rough-Wall Turbulent Boundary Layers". *International Journal of Heat and Mass Transfer*, **36**(1), pp. 147–153.
- [45] Bons, J., 2005. "A Critical Assessment of Reynolds Analogy for Turbine Flows". *ASME Journal of Turbomachinery*, **127**(5), pp. 472–485.
- [46] Aupoix, B., and Spalart, P., 2003. "Extensions of the Spalart-Allmaras Turbulence Model to Account for Wall Roughness". *International Journal of Heat and Fluid Flow*, **24**(4), pp. 454–462.
- [47] Knopp, T., Eisfeld, B., and Calvo, B., 2008. "A New Extension for $k - \omega$ Turbulence Models to Account for Wall Roughness". *International Journal of Heat and Fluid Flow*, **30**(2009), pp. 54–65.
- [48] Eca, L., and Hoekstra, M., 2010. "Numerical Aspects of Including Wall Roughness Effects in the SST $k - \omega$ Eddy-Viscosity Turbulence Model". *Computers & Fluids*, **40**, pp. 299–314.

- [49] Suga, K., Craft, T. J., and Iacovides, H., 2006. “An Analytical Wall-Function for Turbulent Flows and Heat Transfer over Rough Walls”. *International Journal of Heat and Fluid Flow*, **27**(5), pp. 852–866.
- [50] McClain, S. T., Hodge, B. K., and Bons, J. P., 2004. “Predicting Skin Friction and Heat Transfer for Turbulent Flow Over Real Gas Turbine Surface Roughness Using the Discrete Element Method”. *ASME Journal of Turbomachinery*, **126**(2), pp. 259–267.
- [51] Patel, V. C., 2015. “Perspective: Flow at High Reynolds Number and Over Rough Surfaces - Achilles Heel of CFD”. *ASME Journal of Fluids Engineering*, **120**(3), pp. 434–444.
- [52] Stripf, M., Schulz, A., and Bauer, H.-J., 2008. “Modeling of Rough-Wall Boundary Layer Transition and Heat Transfer on Turbine Airfoils”. *ASME Journal of Turbomachinery*, **130**(2), p. 021003.
- [53] Spalart, P., 2015. “Philosophies and Fallacies in Turbulence Modeling”. *Progress in Aerospace Sciences*, **74**, pp. 1–15.
- [54] Bernardini, M., Pirozzoli, S., Orlandi, P., and Lele, K., 2012. “Compressible Boundary Layer Transition Induced by Isolated Roughness Elements”. *Center for Turbulence Research Proceedings of the Summer Program*.
- [55] Rizzetta, D. P., and Visbal, M. R., 2007. “Direct Numerical Simulations of Flow Past an Array of Distributed Roughness Elements”. *AIAA Journal*, **45**(8), pp. 1967–1976.
- [56] Citro, V., Luchini, P., Giannetti, F., and Auteri, F., 2015. “Boundary-layer Flows Past an Hemispherical Roughness Element: DNS, Global Stability and Sensitivity Analysis”. *Procedia IUTAM*, **14**, pp. 173–181. <https://doi.org/10.1016/j.piutam.2015.03.038>.

- [57] Choudhari, M., Li, F., Wu, M., Chang, C.-L., Edwards, J., and Kegerise, M., 2010. "Laminar-Turbulent Transition behind Discrete Roughness Elements in a High-Speed Boundary Layer". *AIAA Paper 2010-1575*.
- [58] Orlandi, P., and Leonardi, S., 2006. "DNS of Turbulent Channel Flows with Two- and Three-Dimensional Roughness". *Journal of Turbulence*, **7**(53).
- [59] Stephani, K. A., and Goldstein, D. B., 2009. "DNS Study of Transient Disturbance Growth and Bypass Transition Due to Realistic Roughness". *AIAA Paper 2009-585*.
- [60] He, L., and Fadl, M., 2016. "Multi-scale Time Integration for Transient Conjugate Heat Transfer". *International Journal of Numerical Methods in Fluids*, **83**, p. 887904.
- [61] Jameson, A., 1991. "Time Dependent Calculations Using Multigrid, with Applications to usteady Flows Past Airfoils and Wings". *AIAA Paper 91-1596*.
- [62] Lad, B., and He, L., 2013. "Use of an Immersed Mesh for High Resolution Modeling of Film Cooling Flows". *ASME Journal of Turbomachinery*, **135**(-), p. 011022.
- [63] He, L., 2012. "Block-Spectral Approach to Film-Cooling Modeling". *ASME Journal of Turbomachinery*, **134**(2), p. 021018.
- [64] He, L., 2013. "Block-Spectral Mapping for Multi-scale Solution". *Journal of Computational Physics*, **250**(-), pp. 13–26.
- [65] He, L., 2013. "Fourier Spectral Modelling for Multi-scale Aerothermal Analysis". *International Journal of Computational Fluid Dynamics*, **27**(2), pp. 118–129.
- [66] He, L., 2017. "Multiscale Block Spectral Solution for Unsteady Flows". *International Journal of Numerical Methods in Fluids*. <http://dx.doi.org/10.1002/fld.4472>.

- [67] Kapsis, M., and He, L., 2018. “Analysis of Aerothermal Characteristics of Surface Micro-Structures”. *ASME Journal of Fluids Engineering*. doi:10.1115/GT2017-63582.
- [68] Sutherland, W., 1893. “The Viscosity of Gases and Molecular Force”. *Philosophical Magazine*, **5**(36), pp. 507–531.
- [69] Richardson, L. F., 1910. “The Approximate Arithmetical Solution of Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam”. *Transactions of the Royal Society of London*, **210**, pp. 307–357.
- [70] He, L., 1993. “New Two-Grid Acceleration Method for Unsteady Navier-Stokes Calculations”. *AIAA Journal of Propulsion and Power*, **9**(2), pp. 272–280.
- [71] He, L., 2000. “Three-Dimensional Unsteady Navier-Stokes Analysis of Stator-Rotor Interaction in Axial-Flow Turbines”. *Proceedings of the Institution of Mechanical Engineers*, **214**, p. A03199.
- [72] He, L., 2008. “Harmonic Solution of Unsteady Flow Around Blades with Separation”. *AIAA Journal*, **46**(6), p. 12991307.
- [73] Cada, M., and Torrilhon, M., 2009. “Compact Third-order Limiter Functions for Finite Volume Methods”. *Journal of Computational Physics*, **228**(11), Jun, pp. 4118–4145.
- [74] Spalart, P. R., and Allmaras, S. R., 1992. “A One-equation Turbulence Model for Aerodynamic Flows”. *AIAA Paper 92-0439*.
- [75] Menter, F. R., 2009. “Review of the Shear-Stress Transport Turbulence Model Experience from an Industrial Perspective”. *International Journal of Computational Fluid Dynamics*, **23**(34).

- [76] Smagorinsky, J., 1963. “General Circulation Experiments with the Primitive Equations”. *Monthly Weather Review*, **91**(3), p. 99164.
- [77] Reynolds, O., 1895. “On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion”. *Philosophical Transactions of the Royal Society of London*, **186**, pp. 123–164.
- [78] Ning, W., and He, L., 2001. “Some Modeling Issues on Trailing-Edge Vortex Shedding”. *AIAA Journal*, **39**(5), pp. 787–793.
- [79] Churchill, S. W., and Ozoe, H., 1973. “Correlations for Laminar Forced Convection in Flow Over an Isothermal Flat Plate and in Developing and Fully Developed Flow in an Isothermal Tube”. *Journal of Heat Transfer*, pp. 416–419.
- [80] Sunden, B., 1980. “Conjugated Heat Transfer from Circular Cylinders in Low Reynolds Number Flow”. *International Journal of Heat and Mass Transfer*, **23**(10), pp. 1359–1367.
- [81] Schaenzer, M., 2018. “Siemens Internal Report”.
- [82] Gupta, G., Little, Z., Reyes, S., Calderon, L., Fernandez, E., and Kapat, J. S., 2017. “Experimental and Numerical Study of Flow in a Square Duct with Positive and Negative Spherical Wall Features”. *AIAA Paper 2017-4978*.
- [83] Elyayan, M. A., and Tafti, D. K., 2008. “Large Eddy Simulation Investigation of Flow and Heat Transfer in a Channel With Dimples and Protrusions”. *ASME Journal of Turbomachinery*, **130**(4), p. 041016.

- [84] Kapsis, M., He, L., Li, Y. S., Wells, R., Valero, O., Krishnababu, S., Gupta, G., Kapat, J., and Schaefer, M., 2019. “Multi-scale Parallelised CFD Modelling Towards Resolving Manufacturable Roughness”. *ASME Paper GT2019-90314*.
- [85] Krall, K. M., and Eckert, E. R. G., 1973. “Local Heat Transfer around a Cylinder at Low Reynolds Number”. *Journal of Heat Transfer*, **95**(2), pp. 273–274.
- [86] Hartnett, J. P., and Irvine, T. F. J., 1972. *Advances in Heat Transfer*, Vol. 8. Academic Press.
- [87] Kosar, A., and Peles, Y., 2006. “Thermal-Hydraulic Performance of MEMS-based Pin Fin Heat Sink”. *Journal of Heat Transfer*, **128**(2), pp. 121–131.
- [88] Liou, M.-S., and Steffen, C. J., 1993. “A New Flux Splitting Scheme”. *Journal of Computational Physics*, **107**(1), pp. 23–39.
- [89] Liou, M.-S., 1996. “A Sequel to AUSM: AUSM⁺”. *Journal of Computational Physics*, **129**(2), pp. 364–382.
- [90] Liou, M.-S., 2006. “A Sequel to AUSM, Part II: AUSM⁺-up for all Speeds”. *Journal of Computational Physics*, **214**(1), pp. 137–170.
- [91] Phongthanapanich, S., 2014. “A Study and of the Numerical and Instability of the AUSM Scheme”. *Science & Technology Asia*, **19**(4), pp. 60–68.
- [92] Halder, P., Sinhamahapatra, K. P., and Singh, N., 2011. “Numerical Investigation of Supersonic Wake of a Wedge Using AUSM⁺ Scheme on Unstructured Grid”. *International Journal of Applied Mathematics and Mechanics*, **11**, pp. 46–68.
- [93] Chima, R. V., and Liou, M., 2003. “Comparison of the AUSM⁺ and H-CUSP Schemes for Turbomachinery applications”. *AIAA Paper 2003-4120*.

-
- [94] Shen, C., Sun, F.-X., and Xia, X.-L., 2013. “Analysis on Transient Conjugate Heat Transfer in Gap-Cavity-Gap Structure Heated by High Speed Airflow”. *International Journal of Heat and Mass Transfer*, **67**, pp. 1030–1038.
- [95] Thakur, S., Shyy, W., and Liou, M.-S., 1995. “Investigation of Convection and Pressure Treatment with Splitting Techniques”. *Nasa Technical Memorandum Paper No. 106868*.
- [96] von Neumann, J., 1945. “First Draft of a Report on the EDVAC”. *Moore School of Electrical Engineering, University of Pennsylvania*, June. accessed on Dec 2017.
- [97] Dumas, J. D. I., 2017. *Computer Architecture: Fundamentals and Principles of Computer Design*, 2nd ed. CRC Press.
- [98] Gropp, W., Lusk, E., and Skjellum, A., 1999. *Using MPI*, second ed. The MIT Press.

Appendix A

Validation Cases for Chapter 3

Validation cases, which may be regarded as internal flow cases, are shown in order to demonstrate proper setup and execution of the commercial CFD solver ANSYS FLUENT 14.5 for the relevant Re regime considered in Chapter 3. Three validation studies are carried out for cylindrical configurations, for which well-established experimental or semi-analytical results are available. They consist of a 2D isolated cylinder (Case 1), a bank of 2D cylinders (Case 2) and a bank of 3D cylinders (Case 3). Case 2 considers of two domains, one with three pins in the streamwise directions and one with five.

A.1 Computational Configuration

Validation is mostly occurred in the laminar regime, and when flow unsteadiness is observed the implicit LES (ILES) is utilised with Dual Time Stepping (DTS) with a second order integration in the real-time regime. In Table A.1 the Re_D at which the ILES is utilised are reported. The implicit SIMPLEC numerical scheme is used with second order upwind discretisation. Structured meshing is applied for all cases, with fitted O-grids around the bluff bodies (cylindrical pedestals).

Table A.1 Re_D where laminar and ILES solutions are obtained

Case	1	2	3
Laminar	28.1, 40, 94.3	20, 40, 60, 80	9, 24, 37
ILES	371, 4640	-	74

The computational domains of Cases 1 and 2 consist of periodic boundaries and iso-flux heated walls. The stagnation and static pressure conditions are specified at the inlet and exit, respectively, matching the Re . For the simulations of Case 3, a narrow domain is taken to minimise the consumption of computational resources. Hence, symmetry boundaries are applied. For the inlet velocity profile the seventh root rule is assumed, while constant heat flux is applied at the bottom of the domain. The fluid and solid domains are coupled via the viscous walls. The top boundary is assumed inviscid and adiabatic. An overview of the configuration for the first three cases is presented in Figure A.1.

A.2 Metrics

For monitoring the heat transfer of the cases the Nu is used, where the chosen characteristic length is the diameter D of a single pin (cylinder):

$$Nu_D = \frac{qD}{(T_{wall} - T_{inlet})K_{cond}} \quad (A.1)$$

where K_{cond} is the thermal conductivity of the fluid. Due to the low Ma the recovery factor is considered minimal, hence this Nu definition is chosen. For Cases 1 and 2, the Nu is the area-averaged Nu on the pins. For Case 3, the area-average Nu consists of both the surfaces of the pins and the bottom smooth viscous wall (interface of fluid and solid).

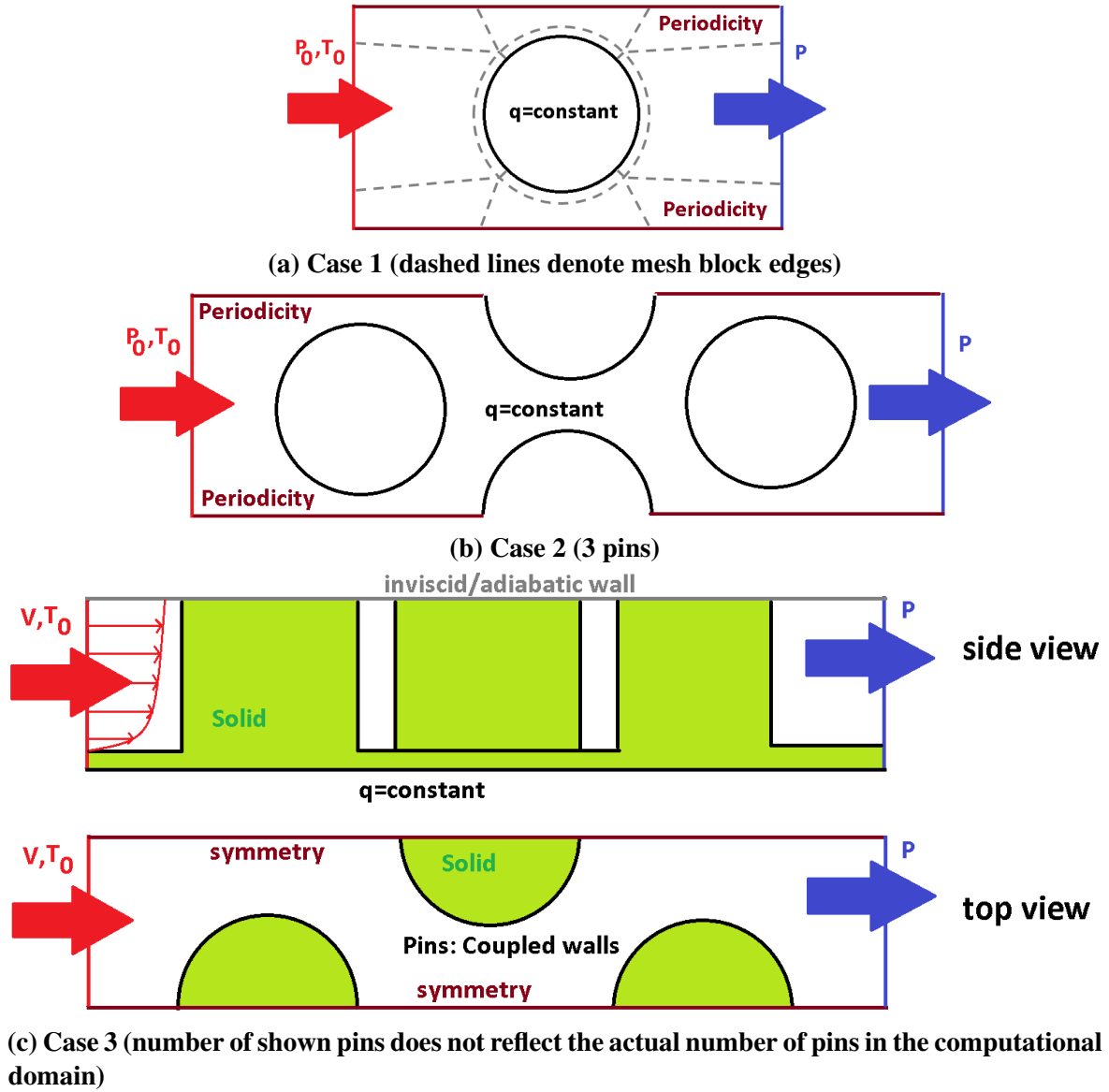


Figure A.1 Computational domains of validation test cases

The Reynolds number is defined based on the inlet bulk velocity as:

$$Re_D = \frac{U_b D}{\nu} \quad (\text{A.2})$$

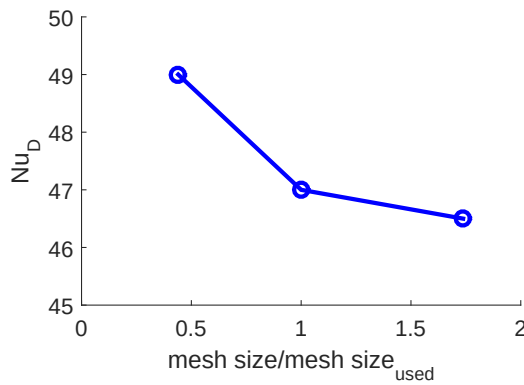
Finally, the relative error between the numerical and the experimental results is defined

as:

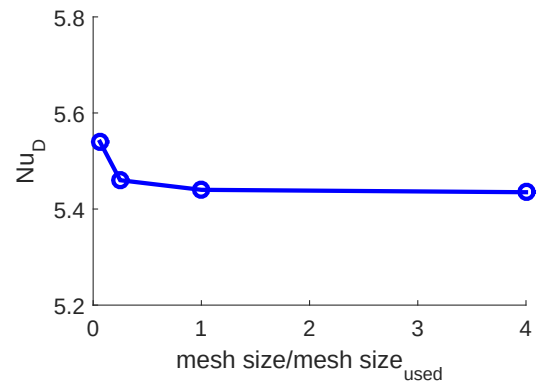
$$e_a = \left| \frac{Nu_{CFD} - Nu_{EXP}}{Nu_{EXP}} \right| \quad (\text{A.3})$$

A.3 Mesh Sensitivity for Validation Cases

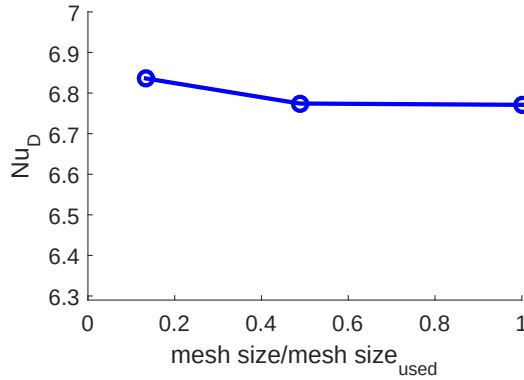
Firstly, mesh sensitivity is assessed prior to validating against well-established data. The most significant results are shown in Figure A.2. The nominal (used) meshes generate mesh independent results, with the maximum relative error of the area-averaged Nu_D between the nominal and the densest mesh being 1% and .1% for cases 1 and 2 (5 pins), respectively. For case 3 the relative error between the nominal and the intermediate mesh is 0%.



(a) Case 1: $Re_D = 4640$. $mesh_{used} = 66 \times 10^3$ cells (2D)



(b) Case 2 (5 pins): $Re_D = 80$, $\lambda_{sp} = 6$. $mesh_{used} = 89 \times 10^3$ cells (2D)



(c) Case 3: $Re_D = 75$. $mesh_{used} = 13 \times 10^6$ cells (3D)

Figure A.2 Mesh dependency study of the area-averaged Nu_D

A.4 2D Single Cylinder at Low Reynolds Number

Initially, a 2D cylinder in the laminar regime is considered. The first set is an experimental case performed by Krall and Eckert [85] of the Nu distribution around the midspan of a cylinder in a low density wind tunnel, with air as the operating medium and $Ma_{inlet} = 0.2$. Moreover, included for comparison is a semi-analytical solution of heat transfer produced by Sunden [80] for 2D steady laminar flow cases.

The surface distributions of the Nu are shown in Figure A.3. The angle is measured such that 0° and 180° correspond to the leading and trailing edge, respectively. For the simulations an incompressible flow model is used and the density is set to match the Re . The results demonstrate a fairly good agreement ($e_a < 7\%$ for area-averaged Nu_D , as reported in Table A.2), especially for the test cases $Re_D = 40, 94$ and 4640 . For $Re_D \geq 371$ vortex shedding starts to occur and, therefore, the time averaged results over a shedding period are shown. The shedding frequency (Strouhal number) is found to be in agreement with the conventional correlation.

Table A.2 Case 1: Relative error e_a of area-averaged $\overline{Nu}_D$

Re_D	28.1	40	94.3	371	4640
e_a	0.1%	0.0%	2.0%	6.9%	6.4%

A.5 2D Staggered Cylinders at Low Reynolds Number

The next step is to validate a two dimensional array of staggered cylinders in the laminar regime. For this case, the correlation proposed by Zukauskas [86] is compared:

$$\overline{Nu}_D = CF Re_{D,Umax}^n Pr^m \left(\frac{Pr}{Pr_{wall}} \right)^{0.25} \quad (A.4)$$

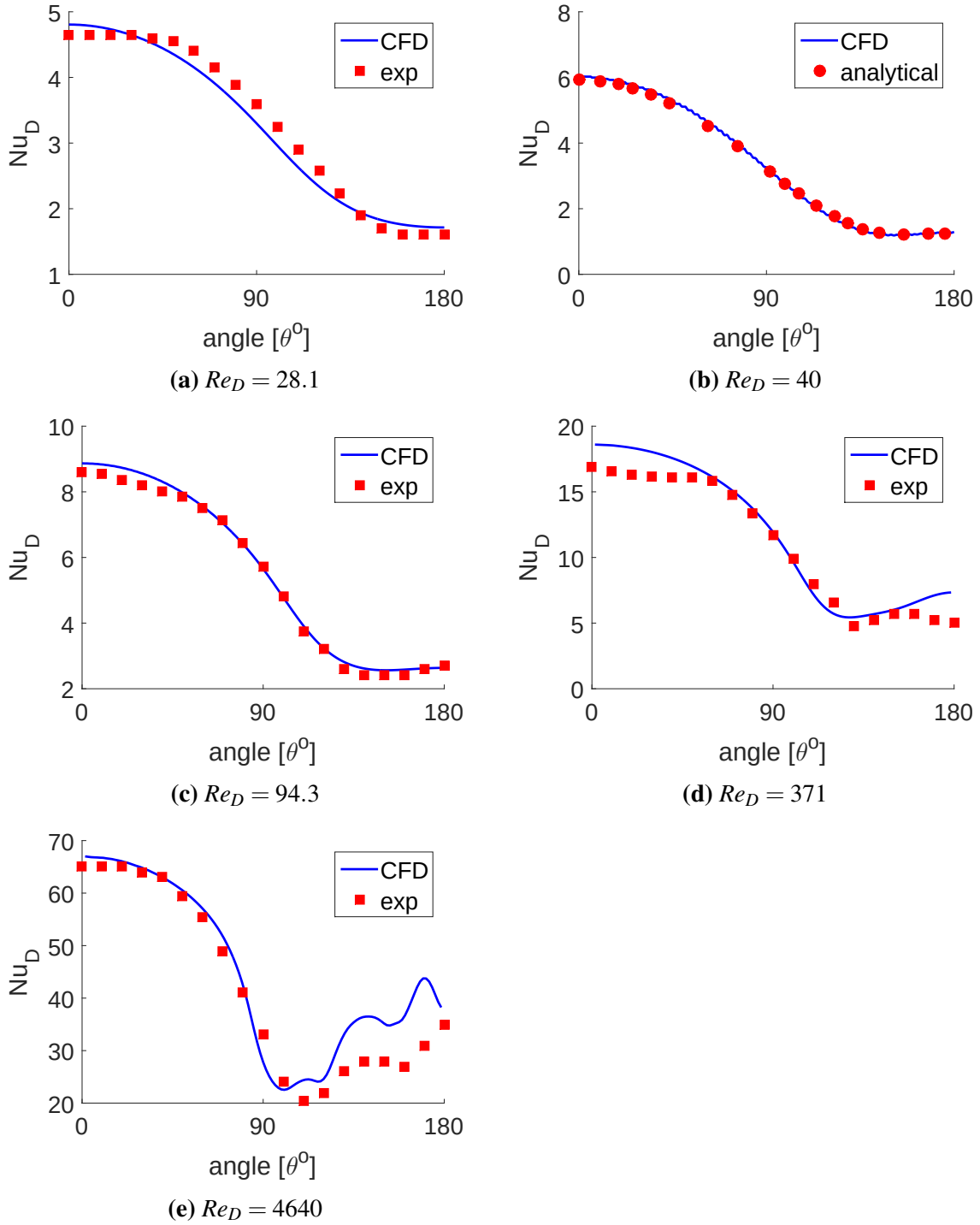
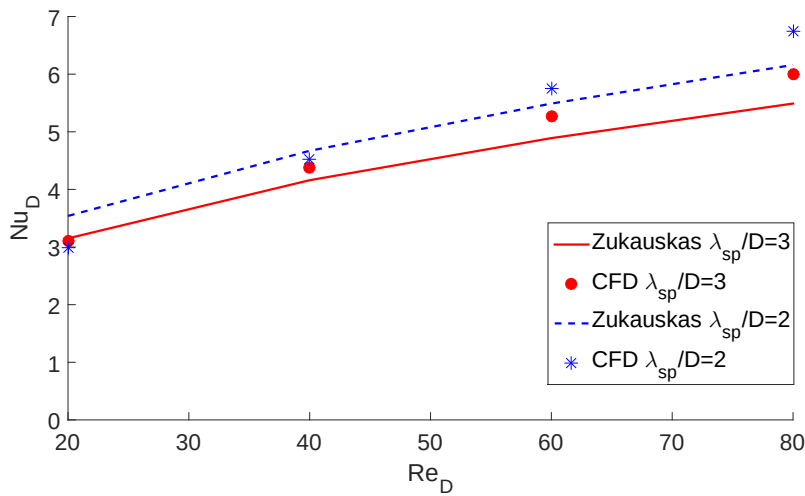


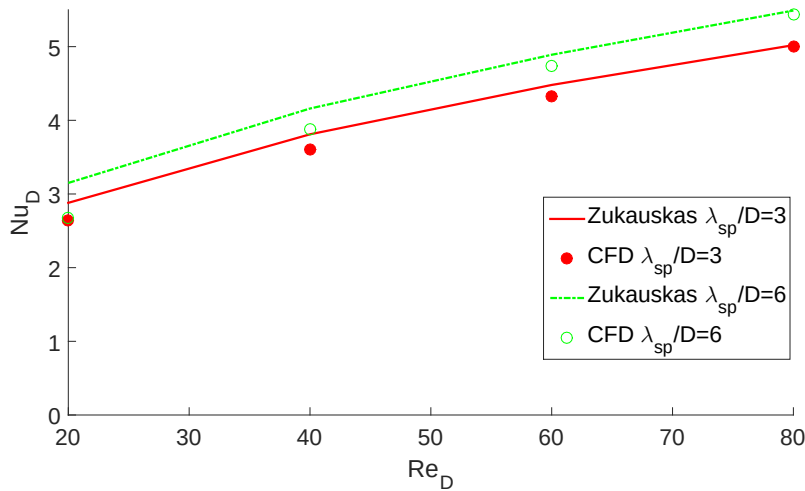
Figure A.3 Case 1 validation results ($Re_D = 40$ from [80], rest from [85])

where m , n , C and F are constants for the tested Reynolds regime, while the ratio Pr/Pr_{wall} is unity for constant fluid properties (here $Pr = 0.744$, $Ms \ll 0.1$). U_{max} is calculated at the narrowest cross section. Several spanwise spacings are examined $\lambda_{sp}/D = 2, 3, 6$ under the streamwise spacing $\lambda_{st}/D = 3$, and the results are shown in Figure A.4. There is good

agreement between the correlations and the present CFD results, as the trends of the variations with the spacing are well predicted. The relative error e_a of the area-averaged Nu_D is reported in Table A.3, and the maximum values occur for the lowest Re_D (less than 20%). Interestingly, when the array of pins is extended in the streamwise direction (five pins), the errors drop with the Re_D . Generally, the error values of the three pins case are higher than those of the five pin case, and for $\lambda_{sp}/D = 3$ the relative error is consistently lower than 10%.



(a) Three pins in the axial direction



(b) Five pins in the axial direction

Figure A.4 Case 2: Area-averaged $\overline{Nu_D}$ comparison between numerical and experimental results

Table A.3 Case 2: Relative error e_a of area-averaged $\overline{Nu}_D$

Re_D	20	40	60	80
3 pins, $\lambda_{sp}/D = 2$	15.5%	3.2%	2.9%	9.6%
3 pins, $\lambda_{sp}/D = 3$	1.6%	5.3%	7.8%	9.3%
5 pins, $\lambda_{sp}/D = 3$	8.3%	5.5%	3.3%	0.4%
5 pins, $\lambda_{sp}/D = 6$	15.2%	6.7%	3.1%	0.9%

A.6 3D Staggered Cylinders at Low Reynolds Number

The present computations are then validated for a three-dimensional configuration (Case 3) subject to a laminar flow regime and the results are compared to the corresponding experimental data. Kosar and Peles [87] investigated the heat transfer in an aluminium micro-sink with cylindrical pins for MEMS cooling applications. The sink is heated from beneath with constant heat flux and is cooled with water. The provided experimental flow properties and conditions are matched in the present computations. The presented results are produced for a steady inlet velocity profile given by the seventh root assumption. Due to the pin aspect ratio, conduction effects are considerable, thus the conjugate heat transfer model is used. The configurations of the experiments and the simulations are shown in Tables A.4 and A.5, respectively.

In Figure A.5, the experimental results show a change in the slope of the total heat load along Re_D that is not present in the numerical results. Nevertheless, considering the uncertainties in the experimental setup and the matching of the boundary conditions, the overall comparison is regarded as fair, and the discrepancies between the experimental and computational results are comparable with the experimental errors as indicated by the error bars. The maximum relative error is found to be 70% at the lowest Re_D , however it drops rapidly and monotonically to 5.6%, as shown in Table A.6.

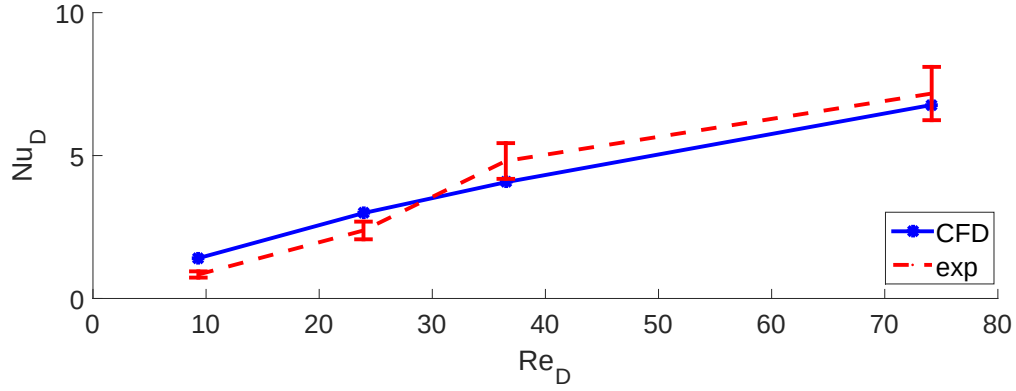


Figure A.5 Case 3: Area-averaged $\overline{Nu}_D$ comparison between numerical and experimental results

Table A.4 Case 3: Experimental conditions

D	H/D	λ_{sp}/D	λ_{st}/D	$N_{st,pins,exp}$	Solid
$99.5\mu m$	2.4422	1.5075	3.0151	66	Al

Table A.5 Case 3: Simulation conditions

Side Boundary	Top Boundary	Bottom Boundary	$N_{st,pins,sim}$
Symmetry	Symmetry	Iso-heat flux viscous wall	40

Table A.6 Case 3: Relative error e_a of area-averaged $\overline{Nu}_D$

Re_D	9	24	37	74
e_a	70%	26%	15%	5.6%

Appendix B

Advection Upstream Splitting Method

This scheme was proposed by Liou and Steffen in 1991 [88] and it was later on updated [89, 90] to tackle issues spotted in the initial algorithm and to accommodate solutions for the whole Mach number range. The purpose was to develop a numerical scheme which joins the efficiency of Flux-Vector Splitting (FVS) and the accuracy of the Flux-Difference Splitting (FDS) schemes [88, 89, 91]. FVS schemes are known for high speed, simplicity and good capturing of strong shocks or rarefaction waves but, on the other hand, highly dissipative results at the boundary layers. On the other hand, FDS schemes show accuracy and results of high quality but with a higher computational cost due to the Jacobian matrix calculation [91, 92]. In the present solver the $AUSM^+$ version has been implemented.

For simplicity the following analysis considers the one-dimensional unsteady Euler equation:

$$\frac{\partial \vec{\Phi}}{\partial t} + \frac{\partial \vec{F}_v}{\partial x} = \vec{0} \quad (\text{B.1a})$$

$$\vec{\Phi} = (\rho, \rho u, \rho E)^T \quad (\text{B.1b})$$

$$\vec{F}_v = (\rho u, \rho u^2 + P, \rho u H_0)^T \quad (\text{B.1c})$$

The onus behind AUSM is to treat the inviscid fluxes vector as the summation of two separate fluxes, the convective (cp) and the pressure (pp) parts:

$$\vec{F}_v = \overbrace{\begin{pmatrix} \rho u \\ \rho u^2 \\ \rho u H_0 \end{pmatrix}}^{\vec{F}_{v,cp}} + \overbrace{\begin{pmatrix} 0 \\ P \\ 0 \end{pmatrix}}^{\vec{F}_{v,pp}} \quad (\text{B.2a})$$

The Jacobian matrices $\tilde{J}$ for the decoupled fluxes are:

$$\frac{\partial \vec{F}_{v,cp}}{\partial \vec{\Phi}} = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 & 2u & 0 \\ -\gamma \frac{E}{\rho} u + (\gamma - 1)u^3 & \gamma \frac{E}{\rho} - \frac{3}{2}(\gamma - 1)u^2 & \gamma u \end{bmatrix} \quad (\text{B.3a})$$

$$\frac{\partial \vec{F}_{v,pp}}{\partial \vec{\Phi}} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{\gamma - 1}{2}u^2 & -(\gamma - 1)u & \gamma - 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{B.3b})$$

The eigenvalues of the above matrices are:

$$\tilde{J}_{cp} - \vec{\Lambda}_{cp} \tilde{I} = \vec{0} \Rightarrow \vec{\Lambda}_{cp} = (0, 2u, \gamma u)^T \quad (\text{B.4a})$$

$$\tilde{J}_{pp} - \vec{\Lambda}_{pp} \tilde{I} = \vec{0} \Rightarrow \vec{\Lambda}_{pp} = (0, -(\gamma - 1)u, 0)^T \quad (\text{B.4b})$$

Therefore, the convective flux propagates downstream, thus convected with a Mach number that is properly defined at the cell interface [90, 93, 94], while the pressure flux propagates upstream due to the acoustic waves, thus convected by a properly defined pressure at the cell interface. Liou and Steffen, based on this physical difference, discretised them separately, hence the benefits of this numerical scheme [88–90]. This signal propagation coupling mitigates the appearance of false oscillations [95]. Below the first order accuracy $AUSM^+$ split functions are mentioned. It is noted that by implementing higher order limiters the scheme increases its accuracy into the corresponding order.

Consider two adjacent cells left (L) and right (R) and their interface ($1/2$). Assuming the positive flow is heading from the left (L) to the right (R) cell the convective and pressure fluxes are:

$$\vec{F}_{v,cp,1/2} = \dot{m}_{1/2} \vec{\Xi}_{1/2} \quad (\text{B.5a})$$

$$\vec{\Xi} = (1, u, H_0)^T \quad (\text{B.5b})$$

$$\vec{F}_{v,pp,1/2} = P_{1/2} (0, \vec{n}, 0)^T \quad (\text{B.5c})$$

where

$$\vec{\Xi}_{1/2} = \begin{cases} \vec{\Xi}_L, & \text{if } \dot{m}_{1/2} > 0 \\ \vec{\Xi}_R, & \text{if } \dot{m}_{1/2} \leq 0 \end{cases} \quad (\text{B.6a})$$

$$\dot{m}_{1/2} = c_{1/2} Ma_{1/2} \begin{cases} \rho_L, & \text{if } Ma_{1/2} > 0 \\ \rho_R, & \text{if } Ma_{1/2} \leq 0 \end{cases} \quad (\text{B.6b})$$

The interface Mach number is found as:

$$Ma_{1/2} = f_4^+(Ma_L) + f_4^-(Ma_R) \quad (\text{B.7})$$

Similarly for the interface pressure:

$$P_{1/2} = f_5^+(Ma_L)P_L + f_5^-(Ma_R)P_R \quad (\text{B.8})$$

The interface speed of sound is chosen as:

$$c_{1/2} = \min(\hat{c}_L, \hat{c}_R) \quad (\text{B.9a})$$

$$\hat{c} = \frac{c^{*2}}{\max(c^*, |u|)} \quad (\text{B.9b})$$

Assuming ideal gas:

$$c^{*2} = \frac{2(\gamma - 1)}{\gamma + 1} H_0 \quad (\text{B.10})$$

And thus, the Mach number at each cell is calculated as:

$$Ma_{L/R} = \frac{u_{L/R}}{c_{1/2}} \quad (\text{B.11})$$

The aforementioned split functions f are polynomial (with the degree mentioned at the subscript) and defined as:

$$f_1^\pm(Ma) = 0.5(Ma \pm |Ma|) \quad (\text{B.12a})$$

$$f_2^\pm(Ma) = \pm 0.25(Ma \pm 1)^2 \quad (\text{B.12b})$$

$$f_4^\pm(Ma) = \begin{cases} f_1^\pm(Ma), & \text{if } |Ma| \geq 1 \\ f_2^\pm(1 \mp 2f_2^\mp), & \text{if } |Ma| < 1 \end{cases} \quad (\text{B.12c})$$

$$f_5^\pm(Ma) = \begin{cases} \frac{f_1^\pm}{Ma}, & \text{if } |Ma| \geq 1 \\ f_2^\pm[(\pm 2 - Ma) \mp 3Ma f_2^\mp], & \text{if } |Ma| < 1 \end{cases} \quad (\text{B.12d})$$

Those properties that are calculated at the interface (1/2) are then used in calculating the flux at the interface.

Appendix C

Parallelisation of the Embedded

Dual-Mesh Multi-Scale Solver

In this chapter the parallelisation efforts of the dual-mesh solver are discussed. Firstly, some background information is presented that will determine the parallelisation approach. Then, the parallelisation approach is discussed, followed by a validation case and a parallel efficiency investigation. The modifications performed aim to generate a solver of sensible parallel efficiency, without increasing significantly the user-defined parameters and the complexity of the code.

C.1 Introductory Remarks

In this section, the main hardware and software characteristics that introduce challenges to the parallelisation are briefly presented.

C.1.1 Hardware Architecture

Nowadays, computer configurations follow the standard stored-programme architecture as introduced by von Neumann [96]. The standard model requires that programmes and data

are both stored in memory and the information is transferred through a (usually) common bus between the memory and the processors, which perform the calculations. This basic configuration constitutes a node and an overview of a typical node architecture is shown in Figure C.1. A single node consists of a single or multiple sockets, or as alternatively called Central Processing Units (CPUs). In turn, each socket may consist of multiple cores. Also, each node includes the Random Access Memory (RAM), where the data and programmes are stored. This architecture results in restrictions on the number of cores that can simultaneously access the RAM. Consequently, restrictions are introduced to the number of simultaneous node-to-node communications, where information is transferred between the memories of each node.

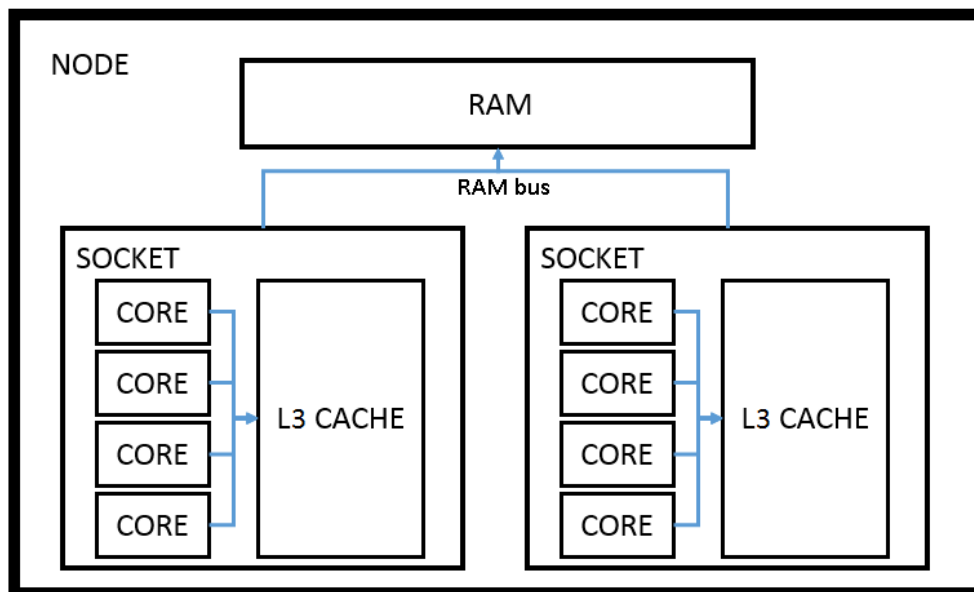


Figure C.1 Hardware architecture of a single node

Retrieving information from the memory introduces latency, especially nowadays that core computing cycles are shorter than those of accessing the RAM. To mitigate this issue, various modifications have been applied, such as the cache memory, which is faster than RAM but with limited storage capabilities. There are different levels of cache memory, from the fastest, which is individual to each core (L1 cache), to the slowest, which is shared by all

the cores of the socket (L3 cache), i.e. the higher the cache frequency, the more limited its storage capabilities are. Consequently, additional hardware acceleration techniques have been explored [97], attempting to mitigate the hardware architecture limitations. Nevertheless, regarding the parallelisation approach, the serial solver is modified in a generic way, without considering specialised, non-universal hardware characteristics.

C.1.2 Message Passage Interface

For performing large computations in parallel, there are two main approaches that can be followed:

- Shared memory: This approach can be applied to groups of sockets (or cores) that share a single memory device. For example, multiple sockets (and cores) can simultaneously perform calculations to different partitions of the same matrix.
- Distributed memory: This approach has emerged to allow for a problem to be solved by multiple nodes, i.e. there are multiple memory devices indirectly connected. This requires appropriate node and core communication.

Considering large scale computational simulations (e.g. CFD) the latter approach is fairly suitable, as the computational domain is partitioned and distributed among the RAM of each node. As a result, it is easier to satisfy memory hardware constraints. In order to ensure portability and universality of the developed parallelised codes, the Message Passing Interface (MPI) library standard has been formed [98].

C.1.3 Blocking and Non-Blocking Communication

In message-passing, when one core needs information that is stored in the local memory of another core, a communication between those two cores is established. There are various

types of communication between two processors, mostly variants of the two main categories: blocking and non-blocking communications.

In blocking communication, when each processor encounters a communication event, it pauses execution. When the processors, which are participating to the communication, have reached this point, communication is initiated. Every processor participating in the communication resumes execution only after communication is completed. This characteristic allows only one communication to occur for a specific processor. Due to this trait, not properly ordered blocking communications may lead to a "bottleneck", i.e. the "bottleneck" occurs when two communication events are triggered only when the other communication is performed.

In contrast, non-blocking communication tackles the "bottleneck" issue by allowing each sender-processor to continue execution even if communication has not been established. The sender-processor of the information stores the data in a buffer from which the receiver-processor copies the data when it is available. The sender will continue execution even if the receiver has not accessed the buffer yet, and since the buffer is of a finite size, large packages of data may cause it to overflow leading to undefined behaviour. Hence, non-blocking is considered less safe than blocking communication when large data are handled [98].

Non-blocking communication may offer significant efficiency improvement for unbalanced core loading, however well-balanced core-loading configurations are mostly considered here. Blocking communication is chosen to ensure consistent results, as well as portability across clusters.

C.2 Halo Updating

The computational domain is partitioned from the start of the solver; each core is assigned its own mesh blocks to handle, thus it is up to the user or a pre-processor to secure load balancing and minimum processor communication. The implication is that core communication is required at multiple stages, either between cores of the same node or between cores of different nodes. For example, considering Figure 4.1, in the **Configure block connectivity** action it is required to exchange global domain information such as coordinates at the block boundaries. The most essential information transfer occurs in the Runge-Kutta integration, where the boundaries of each mesh block are updated. These data exchange events are invoked multiple times during the execution of the **Main loop**.

For updating the boundaries of each mesh block, the grid of the mesh block is extended so that the transferred information can be stored. These virtual partitions of the mesh block are called halos or dummy cells. When a halo requires updating by a mesh block that is handled by a different core, a core communication is established between a pair of cores, where each core sends to the counter core the information requested. It is noted that establishing the communication does consume time, therefore it is sought to transfer as much information as possible in a single data exchange.

For the following analysis the linear communication model is assumed. This means that each transfer of a data cell requires a single unit of communication time. Multiple data cell transfers that occur simultaneously count as a single data cell transfer, i.e. there is no difference in time needed between a pair of cores performing a data transfer while the rest of the cores are idle, and multiple pairs of cores simultaneously performing data transfers of the same bandwidth.

Table C.1 Data exchange cycles for the configuraiton shown in Figure C.2

Exchange / Transfer cycles	Data transfer
1 / 1	Update left side of interface "A", e.g. (2,2) $\longrightarrow$ (2, 1)
1 / 2	Update right side of interface "A", e.g. (2,2) $\longleftarrow$ (2, 1)
2 / 3	Update left side of interface "B", e.g. (2,2) $\longrightarrow$ (2, 3)
2 / 4	Update right side of interface "B", e.g. (2,2) $\longleftarrow$ (2, 3)
3 / 5	Update top side of interface "C", e.g. (2,2) $\longleftarrow$ (1, 2)
3 / 6	Update bottom side of interface "C", e.g. (2,2) $\longrightarrow$ (1, 2)
4 / 7	Update top side of interface "D", e.g. (2,2) $\longleftarrow$ (3, 2)
4 / 8	Update bottom side of interface "D", e.g. (2,2) $\longrightarrow$ (3, 2)

adopted. Constructing individual communication lists while ensuring the absence of "deadlock" is challenging. Hence, the solver identifies all the core pairs that need to exchange their halos and constructs a list of core pairs, or list of communications, that is identical for all cores. Each core has information readily available regarding which block face values have to be requested from other cores and which face values have to be sent to other cores. The solver, after it realises which cores need to communicate with each other, constructs a communication list consisting of all the pairs of communicating cores and performs the data exchange by sequentially scanning the list to avoid deadlock. Despite the sequential accessing, data exchanges between cores can still be performed simultaneously when core IDs appear just once in adjacent pairs. The solver considers configurations that consist of the following two distinctive categories:

- Parallel multi-block configuration, where core communication is required to update only the halos of the base mesh blocks. The embedded mesh either does not exist or it is assigned to the core that handles the corresponding base domain partition.

- Parallel dual-mesh configuration, where core communication is required to perform only the dual-mesh coupling. In this scenario, the base domain is handled by a single core and the corresponding fine mesh blocks are handled by the rest of the cores.

C.2.1 Multi-Block Configuration

A conventional multi-block mesh configuration would only consist of base blocks. Let us consider a rectangular multi-block domain consisting of 4 base blocks, which are distributed among four cores as shown in Figure C.3. Thus, data for updating the halos are exchanged between the cores 0 and 1, 0 and 2, 1 and 3, and 2 and 3. Each core has to send the same amount of information to two other cores. Based on the above, one way of constructing the core communication list is shown in Table C.2.

In Table C.3 the data exchanges performed per cycle are presented. In the first data exchange, cores 0 and 1 perform a communication, however cores 2 and 3 remain idle. This is due to the fact that they require cores 0 and 1, respectively, to be free of any data transferring in order to establish a communication. Once the first data exchange is completed, during cycles 3 and 4, two data transfers can occur simultaneously. Still, the order of the pairs in the list of communications may affect the total time of the communications, thus in Table C.4 a different sequence of core pairs is considered. This alternative list results in fewer data exchange cycles, as shown in Table C.5. Here, each core is occupied during every cycle, similarly to the simple example described in the previous section. This should be taken into account to potentially accelerate the data exchange.

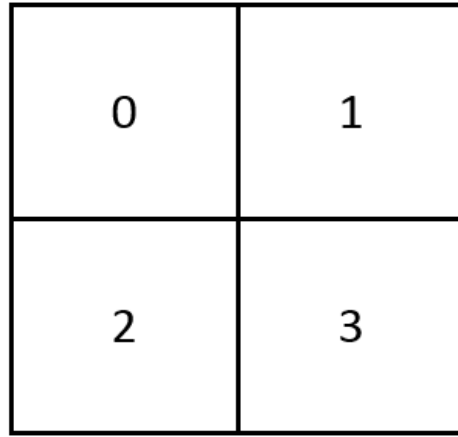


Figure C.3 Example case domain partitioning. Numbers denote the rank of the core

Table C.2 Communication list of the domain shown in Figure C.3

Pair	1	2	3	4
cores	0 - 1	0 - 2	1 - 3	2 - 3

Table C.3 Data exchange cycles for the list shown in Table C.2

Transfer Cycle	1	2	3	4	5	6
Data transfers	0 $\longrightarrow$ 1	0 $\longleftarrow$ 1	0 $\longrightarrow$ 2	0 $\longleftarrow$ 2	2 $\longleftarrow$ 3	2 $\longrightarrow$ 3
			1 $\longrightarrow$ 3	1 $\longleftarrow$ 3		

Table C.4 Alternative communication list

Pair	1	2	3	4
Cores	0 - 1	2 - 3	0 - 2	1 - 3

Table C.5 Data exchange cycles for the list shown in Table C.4

Transfer Cycle	1	2	3	4
Data transfers	0 $\longrightarrow$ 1	0 $\longleftarrow$ 1	0 $\longrightarrow$ 2	0 $\longleftarrow$ 2
	2 $\longleftarrow$ 3	2 $\longrightarrow$ 3	1 $\longrightarrow$ 3	1 $\longleftarrow$ 3

Moreover, it is reminded that each embedded mesh block corresponds to a specific base mesh block. A dual-mesh setup behaves as a conventional parallel multi-block configuration when the base blocks and the corresponding embedded blocks are assigned to the same core.

In this scenario, the core communication is only performed to update the halos of the base mesh blocks.

C.2.2 Dual-Mesh Configuration

Let us now consider the base (coarse mesh) domain and the embedded mesh (fine) being assigned to different cores. The existence of an embedded domain introduces challenges when transitioning to a parallel version of the solver. Let us assume that the coarse mesh is confined in one or a few cores. The embedded blocks, potentially 1000 times denser than the coarse mesh, are assigned to multiple different cores. In order to execute the coarse-fine domain coupling, data will be exchanged between the cores that handle the fine mesh and those that handle the coarse mesh. For simplicity, let us assume that one core handles the base domain and the rest of the cores handle the embedded domain. In this case, the cores that hold the embedded domain need to communicate with that single "base domain" core in order to perform the dual-mesh coupling. These data exchanges can only occur in a sequence, since the "base domain" core can exchange data with only one core at each data exchange cycle.

Considering the dual-mesh coupling, as described in the previous chapter and illustrated in Figure 4.10, the actions of **Update $flux_{C_F}$** , **Update Φ_{C_F}** and **Update boundaries of F domain** require core communication. Still, the parallel efficiency does not drop significantly because only coarse mesh properties can be exchanged for the dual-mesh coupling. This can be achieved since the embedded domain requires base flow properties to update the fine mesh boundaries. Moreover, The base domain requires spatially-averaged flow properties from the embedded domains. Thus, for high embedded grid density, the sequentially performed data exchange would account for a short period of time relative to the time required for the solver calculations.

C.2.3 Parallel Solver Algorithm

In Figure C.4 the main actions of the parallelised solver are illustrated. Initially, each core

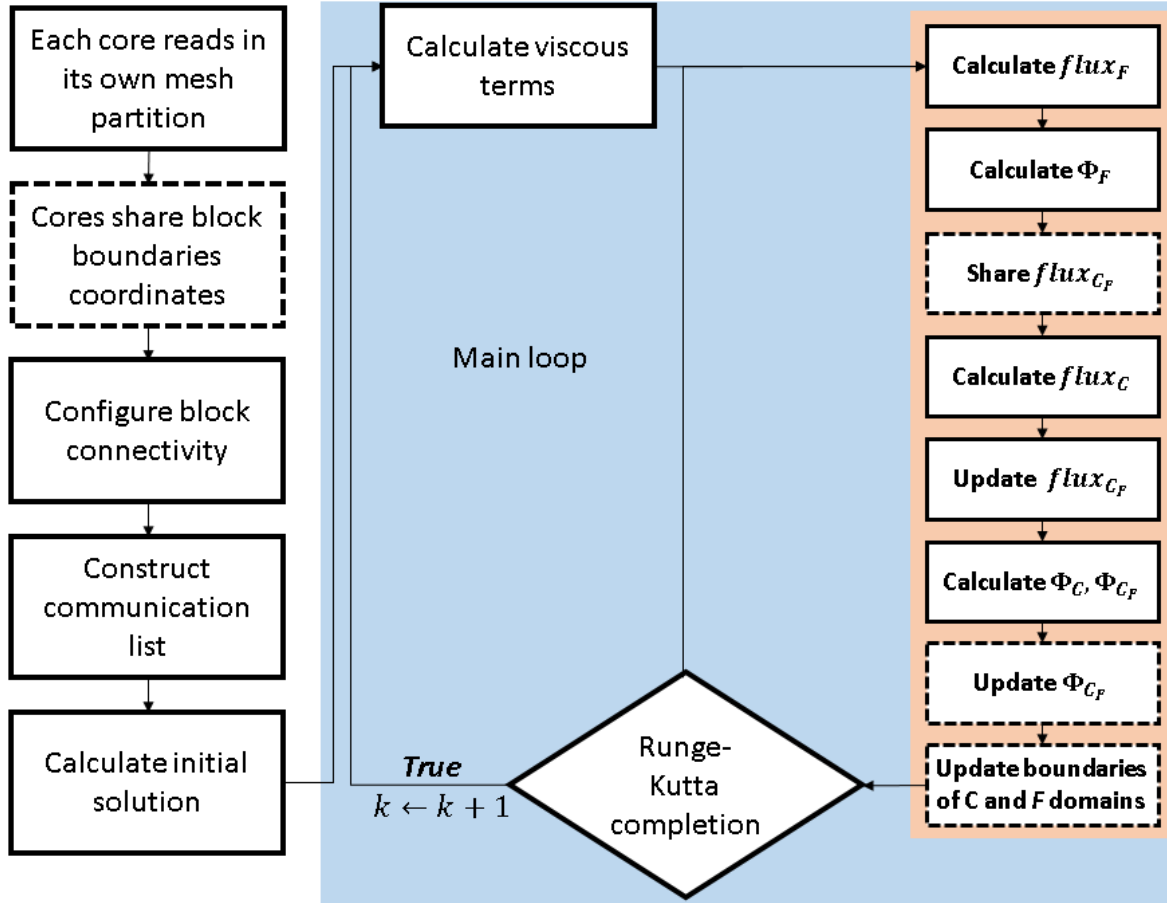


Figure C.4 Algorithm of the parallel solver. Dashed outlines denote required core communication

builds up the mesh of the domain partition they have been assigned. In the **Cores share block boundaries coordinates** all cores acquire the face cells coordinates of all the mesh block boundaries. This information is required to establish the connectivity of the mesh blocks and in turn the pairs of cores that need to be included in the communication list. During the solver iterations, the embedded domain is evaluated first, as the fine mesh boundary fluxes are required when evaluating the base domain. In a parallel multi-block configuration, core communication is not required for sharing the $flux_{C_F}$, which are the area-averaged fine mesh fluxes at the base-embedded domain interface, and for updating the C_F regions. Prior

to the following iteration, the mesh block boundaries are updated by scanning through the communication list.

C.3 Example of Verification of the Parallel Version

The parallel solver was verified against the serial version. For example, a case with four 3D square pedestals is considered here, as shown in Figure C.5. The width of the domain is equal to the square side dimension times six. Here, embedded features are applied to define the embedded solid and refine the region around the pedestals. There is one base block and six embedded blocks refining the mesh 9 times in each direction (Figure C.6). Embedded walls are fitted to determine the solid domain of the pedestals. Both solvers are compiled with the same options and the parallel execution performed on various partition configurations. The agreement between the two versions can be seen in Figures C.7a and C.7b.

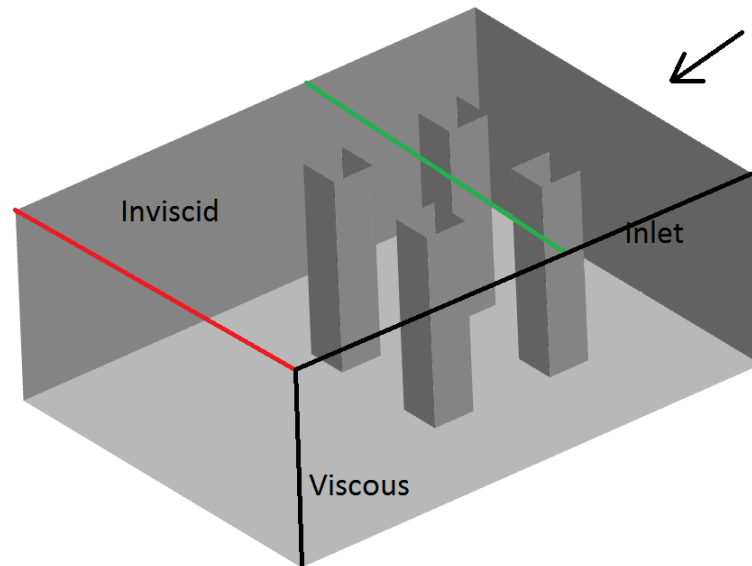


Figure C.5 Cross section of the physical domain at a symmetry plane. Green line indicates the plotting line at Figure C.7b, and it is located between the pedestals, equidistantly from the upstream and downstream pedestals. Red line, which is located at the exit plane, sets the location of Figure C.7a

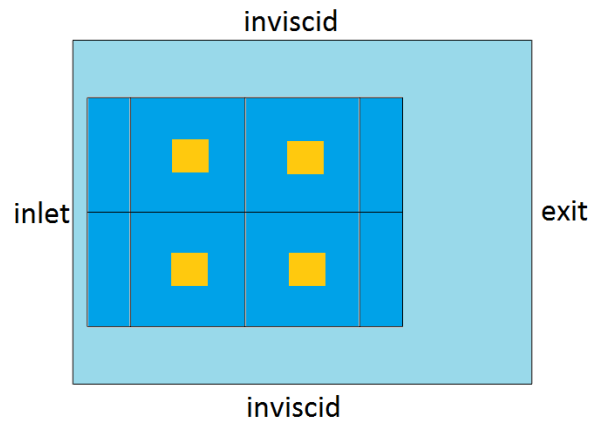
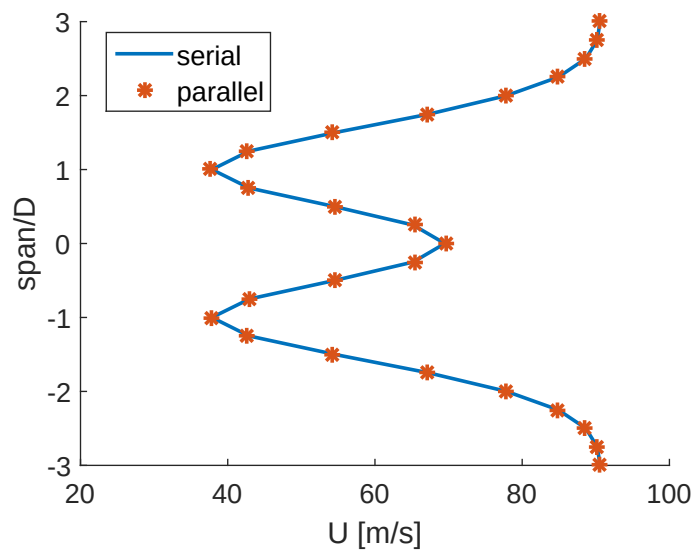
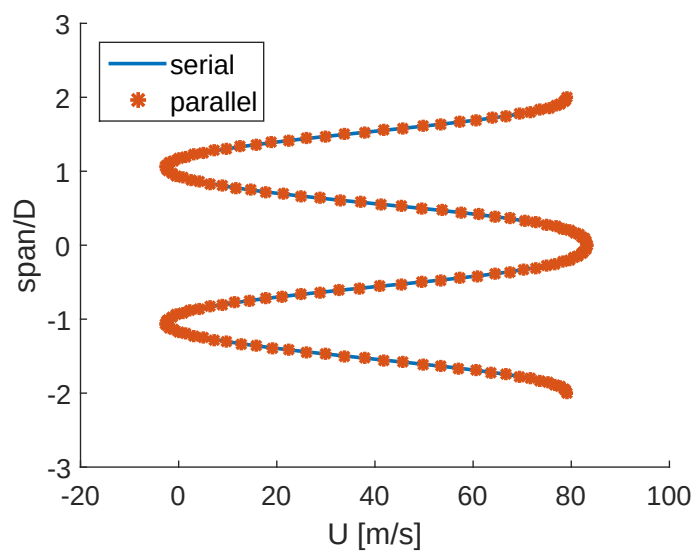


Figure C.6 Top view of the block distribution. Light-blue region is the base block, dark-blue regions are the embedded blocks. Yellow squares indicate solid domains



(a) Red line (coarse mesh)



(b) Green line (fine mesh)

Figure C.7 Comparison of axial velocity U distribution along the plotting lines illustrated in Figure C.5

C.4 Parallel Efficiency

In general, there are two ways of investigating the parallel efficiency:

- **Strong Scaling:** Total load of the case is maintained constant as the number of cores increases. Thus, the load per core decreases. This approach investigates the ability of the solver to efficiently increase the speed of the calculations.
- **Weak Scaling:** Load per core is maintained constant, while more partitions are included in the solution. Thus, the total load increases as the number of cores increases. This approach investigates the ability of the solver to increase the size of the problem without significantly increasing the calculation time.

Here, the weak scaling will be used to determine code efficiency, as the interest lies in running larger test cases. Efficiency/scaling is determined as:

$$\eta_p = \frac{\text{run time for } N_{ref} \text{ cores}}{\text{run time for } N \text{ cores}} \quad (\text{C.1})$$

where N_{ref} is the reference number of cores, i.e. the least number of cores used in the study.

When selecting N_{ref} , a couple of things need to be considered. Firstly, for a single node, as the number of utilised cores increases, regardless of the existence of communication between the spawned processes, the performance declines. As mentioned earlier, the memory bandwidth is not limitless, as there is one bus to transfer data from/to RAM. Hence, as the number of spawned serial processes increases, performance drops. Considering this, the number of cores added, as the size of the problem increases, should be equal to $\text{integer} \times N_{ref}$, with each node consisting of N_{ref} cores. Now, let us assume that there is a node cluster consisting of identical nodes. For an identical node setup, as nodes are added the configuration

remains consistent. However, this does not apply when the node setup is increased from a single node to two, as the single node does not require data exchange through node-connecting cabling. Based on the above, N_{ref} is chosen to be equal to the number of cores of two nodes (here $N_{ref} = 32$ cores).

Two cases are considered for this study, with the first one investigating the multi-block parallel configuration efficiency and the latter the dual-mesh parallel configuration. Same hardware and compilation parameters are employed for all the simulations³. Sampling was performed by CPU wall-clock readings at the start and end of the **Main loop**.

C.4.1 Multi-Block Configuration

The first study is focused on an ideal multi-block case. The load per core is constant and balanced among all cores and the number of communications per core is constant as well. The mesh blocks are distributed in a 2D manner, as shown in Figure C.8, and each block is assigned to a different core. As the number of cores increases the domain is extended in the direction of the arrow. Due to this configuration, each core needs to communicate with three cores. Each block is a cube of 80^3 elements and core communication should be responsible for any efficiency reduction.

The weak scaling is presented in Figure C.9 where the performance drops significantly, but remains at acceptable values with a declining slope. Yet, efficiency could potentially be improved in the future. In the previous section, it was discussed how a domain partitioned in a Cartesian manner demonstrates a constant total number of cycles of required data transfers, e.g. eight for a two-dimensionally partitioned domain. Considering the arbitrary partitioning,

³ Compilation: MVAPICH2 (version 2.1.0) wrapper for Intel Fortran compiler (version 2016), with optimisation (-O3) and profiling (-pg) flags. Increased memory cap with -mcmmodel=medium

Execution: mpirun with Uniform Memory Access was utilised.

Hardware: (Arcus-Haswell IBM/Lenovo NeXtScale cluster) two sockets per node, 8 physical cores (Intel Xeon E5-2640 v3 at 2.60GHz) per socket

a less efficient yet robust approach was adopted. The implication is that the total number of cycles required will scale up, as the communication list includes all the communication core pairs. Reformulating the single list into multiple communication lists should improve parallel efficiency, as the lists would be simultaneously executed. Such approach would require careful consideration of potential "deadlock" events.

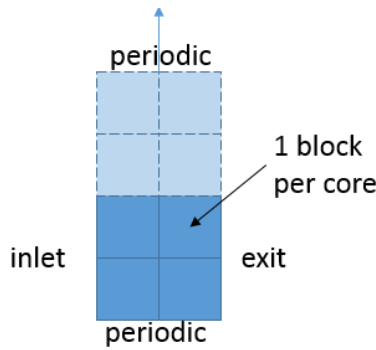


Figure C.8 Multi-block setup: Domain

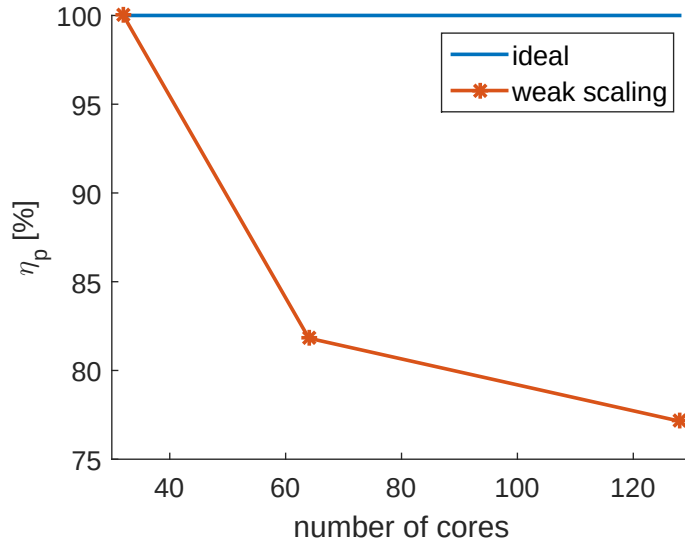


Figure C.9 Multi-block configuration: Weak scaling

C.4.2 Dual-Mesh Configuration

The second case investigates the parallel efficiency of a parallel dual-mesh setup case. The domain is illustrated in Figure C.10, where the coarse base block consists of $62 \times 212 \times 10$ cells and it is assigned to one core exclusively. The volume is refined 10 times in each direction by the embedded blocks and the volumes refined by each embedded block is of 10^3 cells each, resulting in fine blocks of 100^3 cells. Each embedded block is assigned to one core exclusively, thus the size of the total coarse volume that is refined is dependent on the number of cores. The embedded blocks are organised in a $N_x \times 6$ pattern, where N_x is the number of columns and it varies according to the number of cores.

In this configuration, the core with the base block has to send information to the rest of the cores to update the halo cells of each fine mesh blocks. Vice versa, each core handling a fine mesh block has to send information to update the volume cells of the coarse mesh block and the halo cells for correcting the fluxes. This results in a sensible efficiency drop as presented in Figure C.11, although the core pairs of the communication list are sequentially executed. Core communication demand grows linearly, as $(N_{cores} - 1)$ cores need to exchange data with one specific core. Still, the efficiency remains within acceptable levels due to the relatively small size of the transferred halos.

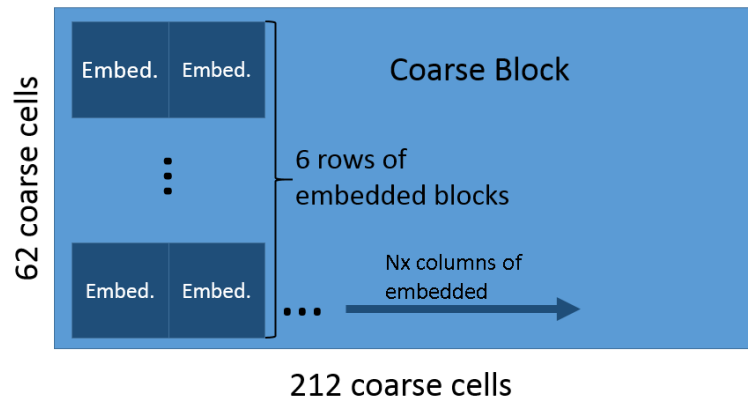


Figure C.10 Dual-mesh configuration: Domain

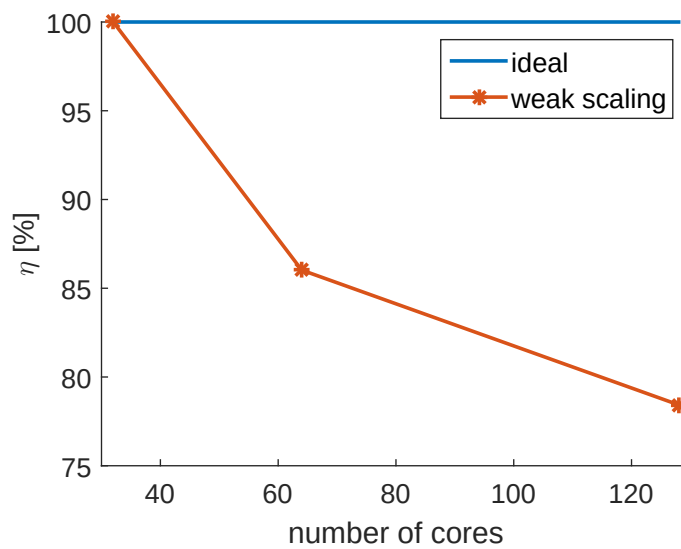


Figure C.11 Dual-mesh configuration: Weak scaling

C.5 Remarks

In this appendix, the parallelisation efforts of the embedded dual-mesh solver were presented. Initially, a basic node architecture was illustrated. MPI was selected as a parallelisation approach, since it allows the harvesting of the computational power of multiple nodes. Updating of the halos was performed through blocking communication, which requires careful considerations of the sequence of data exchanges to avoid "deadlock".

Due to the ability of the solver to handle arbitrary mesh block connectivities, a conservative approach was adopted. A global list, common to all cores, is constructed that determines which data exchange is performed. The list is sequentially scanned, and the data exchanges of adjacent pairs of cores are able to be simultaneously performed if each core is invoked once in these pairs. Still, data transfer between a core with a partition of the base domain and the cores with the corresponding embedded mesh blocks can only be sequentially performed. Fortunately, only coarse mesh flow properties are required to be exchanged to perform the dual-mesh coupling. The parallel version was verified against the serial version, and a sample was presented.

In the end, a parallel efficiency study was performed. Although improvements are possible for accelerating the halo updating, the results are still regarded as adequate, with the parallel efficiency curves of the first test case seemingly converging to a flat plateau and with values over 75% for the tested range. The parallel dual-mesh configuration case demonstrated similar efficiency levels, despite the sequential execution of the communication list, since only coarse domain values need to be transferred.

Appendix D

Parallelisation of the Block Spectral

Multi-scale Solver

Having discussed the modifications to accommodate direct multi-scale simulations in parallel, the addition of the Block Spectral Method to the parallelised code is now presented. In terms of the code structure, the BSM implementation is an extension of the embedded dual-mesh multi-scale solver. The formulation of the BSM results in all the BS calculations being only performed on the base domain, i.e. no information from the embedded domain is directly required.

Ideally, the whole base domain that is involved to the BS mapping is assigned to a single core. In this case, since all the calculations of the BSM are performed on the coarse mesh, there is no need for core communication for the BS calculations. However, in some occasions the base domain may be required to be partitioned across more than one cores, thus the parallelisation of the BS solver should be explored. Therefore, the current chapter also describes the parallelisation approach of the subroutine **Calculate Spectra** TST_{CC} from Chapter 4, Figure 4.15 for this scenario. This approach differs from the one described

in the previous appendix, as now global information is required to build the pointwise region-to-region spectral representation. The rest of the actions performed for the BS mapping do not require any core communication.

Initially, the parallelisation approach adopted when the base domain is partitioned across multiple cores is discussed. Following this, the approach is presented for the case where the base domain is not partitioned across multiple cores. In the final section, a parallel efficiency study is conducted for the two scenarios.

D.1 Approach

D.1.1 Partitioned Base Domain

As discussed in Chapter 4, the TST_{C_C} mapping is applied directly to the cells of the base mesh block(s), resulting in requiring a low number of calculations. Since the Fourier series are adopted as a basis function, the calculation of the coefficients of the spectral representation, here A, B, C, D as described in Chapter 4, Equation 4.29, requires global information. Thus, the BS calculation occurs in parallel only when the base mesh is partitioned and scattered across multiple cores.

This process can be broken into two steps: the **gather** and the **broadcast** stages. These stages are performed through collective MPI commands that utilise efficient tree algorithms, as briefly described in Appendix E:

- **Gather:** This stage is performed either by the **MPI Reduce** or the **MPI All-Gather** commands. The former reduces information from all cores, resulting in the final result being stored at the master core. The latter shares partitioned data among all cores, resulting in all the cores possessing all the data partitions.

- **Broadcast:** This stage broadcasts information from the master core to the rest of the cores and it is performed by the **MPI Broadcast** command .

The positions of each step determine the amount of information that is gathered and broadcast. In addition, the positions of each step determine the amount of calculations performed at each core. For the analysis of this section, the efficiency of five methods is explored.

Method A

Method A is shown in Algorithm 1. Initially, each core that handles a partition with source regions performs a sub-calculation of the A, B, C, D coefficients. From Equation 4.29, the coefficients are calculated for each harmonic and each mesh point (i, j, k) of the repeated mesh topology. Each term of the summation is linked to a single source region (C_F). Hence, it is possible to perform the summation partially at each core and then sum their contributions.

Algorithm 1 Method A

- 1: For each core perform the following loop:
 - for each mesh point (i, j, k)
 - for each harmonic M_h, N_h
 - perform A, B, C, D calculation: summation (Equation 4.29) for the source regions handled by each core
 - 2: Reduce(A, B, C, D)
 - 3: Broadcast(A, B, C, D)
 - 4: Calculate source propagation at each core
-

Method B

Similarly to the first algorithm, the global data needed for the calculations are gathered in one core. As shown in Algorithm 2, the master core receives all the required information to calculate the Fourier coefficients. Effectively, the master core gathers the information of all the source terms. Once the needed data are gathered, the master core calculates the A, B, C, D

coefficients and then broadcasts them. From that point each core propagates the sources on their own.

Algorithm 2 Method B

- 1: Copy needed data to a single, master core
 - 2: For the master core perform the following loop:
 - for each mesh point (i, j, k)
 - for each harmonic M_h, N_h
 - perform A, B, C, D calculation: summation (Equation 4.29) for *all* source regions
 - 3: Broadcast (A, B, C, D)
 - 4: Calculate source propagation at each core
-

Method C

In Method C a different approach is adopted, as shown in Algorithm 3. All required information to construct the Fourier coefficients is gathered and shared among all cores by the **MPI All-Gather** command. Following this, each core constructs the mapped source terms on their own.

Algorithm 3 Method C

- 1: All-Gather: Share all needed data to calculate A, B, C, D among the involved cores
 - 2: For each core perform the following loop:
 - for each mesh point (i, j, k)
 - for each harmonic M_h, N_h
 - perform A, B, C, D calculation: summation (Equation 4.29) for *all* source regions
 - 3: Calculate source propagation at each core
-

Method D

The fourth method, which is shown in Algorithm 4, is a variation of method A. Here, instead of utilising **MPI Reduce** and **MPI Broadcast**, the **MPI All-Gather** command shares

the partial summation contributions among the cores. Following this, each core performs the reduction and the calculation of the Fourier coefficients simultaneously.

Algorithm 4 Method D

- 1: For each core perform the following loop:
 - for each mesh point (i, j, k)
 - for each harmonic M_h, N_h
 - perform A, B, C, D calculation: summation (Equation 4.29) for the source regions handled by each core
 - 2: All-Gather(partial A, B, C, D)
 - 3: Reduction(A, B, C, D) at each core
 - 4: Calculate source propagation at each core
-

Method E

Method E, illustrated in Algorithm 5, is provided as an example of an inefficient approach. Each core performs the partial calculations of the Fourier coefficients, which are then reduced in a single master core. Then, the master core calculates the mapped source terms, which are in turn broadcast to the rest of the cores.

Algorithm 5 Method E

- 1: For each core perform the following loop:
 - for each mesh point (i, j, k)
 - for each harmonic M_h, N_h
 - perform A, B, C, D calculation: summation (Equation 4.29) for the source regions handled by each core
 - 2: Reduce(A, B, C, D) at the master core
 - 3: Calculate source propagation at the master core
 - 4: Broadcast(TST_{C_c})
-

The BSM with the Fourier basis function aims for the number of mapped regions to be significantly higher than the number of the source regions. The fast convergence of the Fourier series over the number of harmonics, and thus the number of source areas, would

allow such configuration. Therefore, it would be less time consuming to avoid broadcasting the mapped source terms.

Comparison

Let us assume a problem with identical base blocks, each corresponding to one core. Since the base domain is partitioned, global data exchange is unavoidable. For simplicity, it is assumed that there are in total $N_{S,g}$ source regions that are uniformly distributed. Hence, each base block is assigned $N_{S,l}$ source regions and the size of each source region is M_c coarse mesh cells. The source terms are 2D mapped onto a total of $N_{C,g}$ regions, with $N_{C,l}$ regions being assigned to each core.

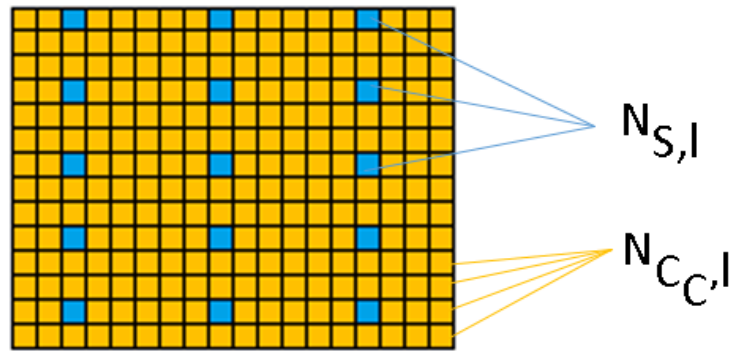


Figure D.1 Schematic of the regions of each base mesh block

As in the previous appendix, the linear communication model is adopted. By reference to Appendix E, the time units required by each collective MPI command may be identified. Also, the same idea is adopted for measuring computation times: each calculation on a data cell requires a single unit of calculation time. Multiple data cell calculations that occur simultaneously count as a single calculation time unit. Based on these assumptions, the total transfer time is predicted for each method.

Method A requires $\log_2 N_{cores}$ cycles for both the gather and broadcast stages. During each cycle, a core receives $M_c \times N_h$ cells and this process needs to be performed four times (for 2D mapping) to calculate the whole set of the Fourier coefficients. For calculations,

each core processes $4 \times M_c \times N_h \times N_{S,l}$ cells for the partial summation of the Fourier and $4 \times M_c \times N_h \times N_{C,l}$ cells for the required amount of the mapped TST.

Method B requires $(N_{cores} - 1)$ cycles for the gather stage and during each cycle the master core receives $M_c \times N_{S,l}$ cells. The broadcast step is identical to that of Method A. For calculations, the master core processes $4 \times M_c \times N_h \times N_{S,g}$ cells for the summation of the Fourier coefficients. Each core processes $4 \times M_c \times N_h \times N_{C,l}$ cells for the required amount of mapped sources.

Method C requires N_{cores} data exchange cycles for the gather stage. Each data transfer cycle is repeated twice due to the two-way communication and during each transfer cycle each core receives or sends $M_c \times N_{S,l}$ cells. For calculations, each core processes $4 \times M_c \times N_h \times N_{S,g}$ cells for the summation of the Fourier coefficients. Each core processes $4 \times M_c \times N_h \times N_{C,l}$ cells for the required number of mapped sources.

Method D requires N_{cores} data exchange cycles for the gather stage. Each data transfer cycle is repeated twice due to the two-way communication and during each transfer cycle each core receives or sends $4 \times M_c \times N_h$ cells. For the calculations, each core processes $4 \times M_c \times N_h \times N_{S,l}$ cells both for the partial summation, prior to the gather phase, and for the full summation for the reduction phase of the Fourier coefficients. Also, each core processes $4 \times M_c \times N_h \times N_{C,l}$ cells for the required number of mapped sources.

Method E requires $\log_2 N_{cores}$ cycles for the gather stage and during each cycle the master core receives $M_c \times N_h$ cells. This is repeated for all the Fourier coefficients. The broadcast stage is performed through $4 \times \log_2 N_{cores}$ cycles, each one transferring $M_c \times N_h \times N_{C,g}$ cells. For the calculations, the master core processes $4 \times M_c \times N_h \times N_{S,g}$ cells for the summation of the Fourier coefficients. Also, the master core processes $4 \times M_c \times N_h \times N_{C,g}$ cells for the total amount of mapped sources.

The predictions are summed up in Tables D.1 and D.2. Considering total calculation time units Method A ranks first in efficiency, whereas Method E last. Comparing the total communication time units does not generate a clear conclusion. Method E is shown to be the least efficient, and Method D requires more time to complete the gather stage than Method A. Any comparisons between methods A, B and C do not provide a definite conclusion, as their efficiency differences are dependent on the magnitudes of the N_h and $N_{S,l}$. Method A is characterised by a straightforward implementation and provides the minimum amount of calculations, hence it is selected for the following parallel efficiency study.

Table D.1 Total communication time units

Method	Gather	Broadcast
A	$4 \times \log_2 N_{cores} \times M_c \times N_h$	$4 \times \log_2 N_{cores} \times M_c \times N_h$
B	$(N_{cores} - 1) \times M_c \times N_{S,l}$	$4 \times \log_2 N_{cores} \times M_c \times N_h$
C	$2 \times N_{cores} \times M_c \times N_{S,l}$	-
D	$2 \times 4 \times N_{cores} \times M_c \times N_h$	-
E	$4 \times \log_2 N_{cores} \times M_c \times N_h$	$4 \times \log_2 N_{cores} \times M_c \times N_h \times N_{C,g}$

Table D.2 Total calculation time units

Method	Master core (only)	Each core
A	-	$4 \times M_c \times N_h \times (N_{S,l} + N_{C,l})$
B	$4 \times M_c \times N_h \times N_{S,g}$	$4 \times M_c \times N_h \times N_{C,l}$
C	-	$4 \times M_c \times N_h \times (N_{S,g} + N_{C,l})$
D	-	$4 \times M_c \times N_h \times (2 \times N_{S,l} + N_{C,l})$
E	$4 \times M_c \times N_h \times (N_{S,g} + N_{C,g})$	-

D.1.2 Non-Partitioned Base Domain

The above subsection explored the different parallelisation variations in the case of a partitioned base domain. As the amount of communications scales with the number of cores,

the weak scaling is significantly reduced, as it will be also presented in the following section. Therefore, a largely partitioned base domain should not be the primary strategy of choice and the above analysis should be relevant only when the base domain is required to be partitioned across few cores.

Ideally, the whole base domain is handled by a single core, a plausible scenario considering its low cell-count and thus the low amount of required RAM. It was mentioned in Chapter 4 that all the required BS calculations are directly applied on to the base domain, as expressed by the source terms of the right hand side of Equation 4.25. Therefore, for a non-partitioned base domain no core communication is required to perform the BSM calculations. In other words, the parallel BS solver behaves as a parallel dual-mesh solver when the embedded mesh and the corresponding base domain are assigned to different cores.

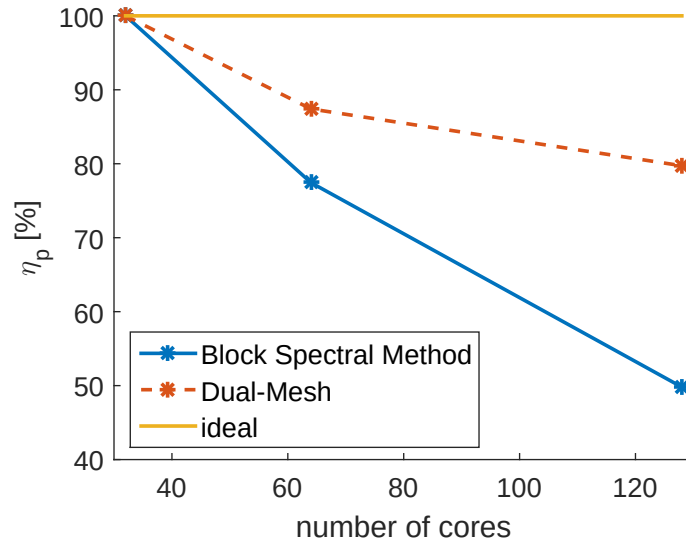
D.2 Parallel Efficiency

For this parallel efficiency study the hardware, the value of N_{ref} and the scaling investigated are the same as in the parallel efficiency study of Appendix C.

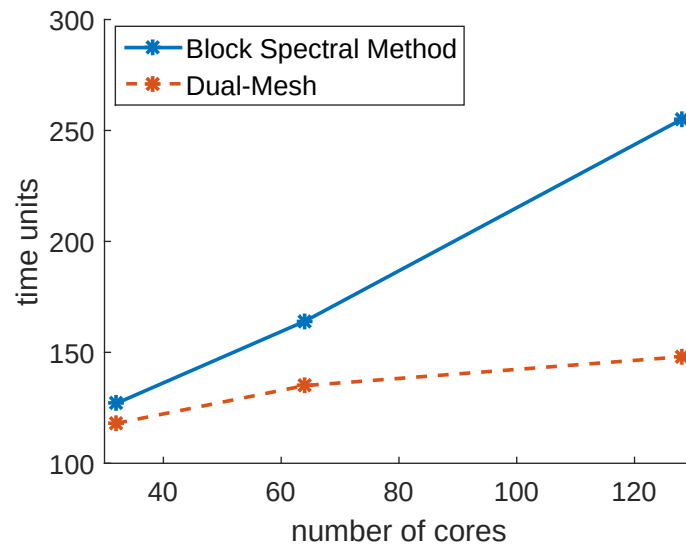
D.2.1 Case of Partioned Base Domain

The domain considered is similar to the one used above as an example. Each embedded block and its respective coarse mesh partition can be assigned to one core. The amount of source regions per block is unity, hence $N_{S,l} = 1$. The number of harmonics is calculated from the Fourier relation $N_h = (N_{S,g} - 1)/2$. The number of local mapped areas per base block does not affect communication times for the tested method, thus $N_{C,l} = 1$. Each base block is $80 \times 80 \times 1$ cells, each embedded block is $100 \times 100 \times 1$ and the coarse domain is refined $10 \times 10 \times 1$ times.

In Figures D.2a and D.2b the efficiency of the parallel BS solver and the execution time units are presented, respectively. Also, the scaling of the dual-mesh solver, which in parallel is



(a) Weak scaling



(b) Time comparisons

Figure D.2 Partitioned base domain

effectively demonstrating a multi-block behaviour, is included for comparison. The efficiency of the BS solver declines rapidly ($e_{a,max} \approx 22\%$ when compared to the dual-mesh solver), while the execution time follows an equivalent trend, presenting a 72% increase.

The steep gradient of the parallel efficiency can be identified in the aforementioned analysis. From Table D.1 the ideal time for completing the *Reduce* and *Broadcast* commands

of Method A would correspond to:

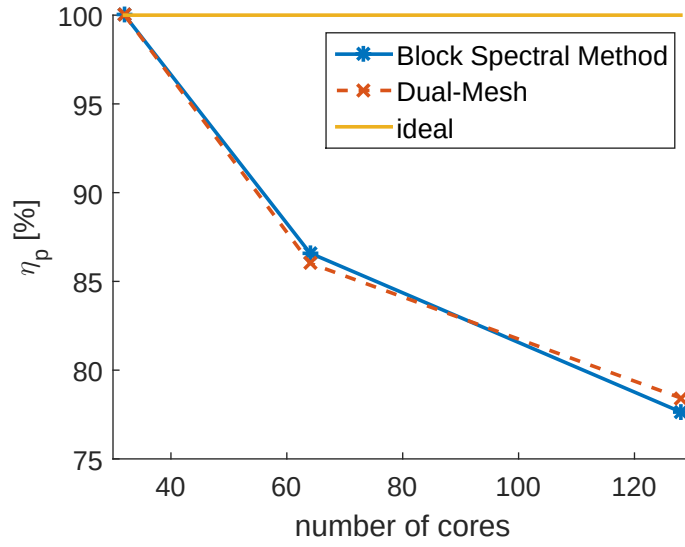
$$2 \times 4 \times \log_2 N_{cores} \times M_c \times N_h \quad (\text{D.1})$$

where $N_h \propto N_{S,g} \propto N_{cores}$ in the present test case. Thus the amount of transferred information increases proportionally to $N_{cores} \log_2(N_{cores})$. Since global data transfers are required, the efficiency is reduced and adequate weak scaling cannot be achieved. Still, it is noted that this configuration is used only for identifying the weak scaling, as the base domain regions participating to the BS mapping would not be realistically distributed across a large number of cores.

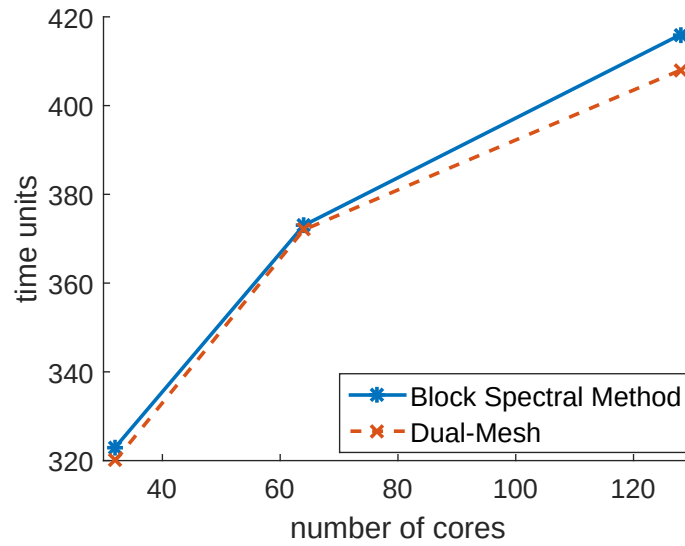
D.2.2 Case of Non-Partitioned Base Domain

As discussed in the previous section, the sensible approach is to partition the coarse mesh the least plausible. If the coarse mesh is included in just a single mesh block/core then the parallel behaviour of the BS solver converges to the parallel dual-mesh configuration, as discussed in the previous appendix. In Figure D.3a it is shown that the parallel efficiency is unchanged when compared to the trend of the parallel dual-mesh configuration ($e_{a,max} \approx 1\%$). Moreover, in Figure D.3b the required time units are presented, with the maximum computational overhead for the calculation of the Fourier coefficients being equal to $\approx 2\%$.

It is noted that the parallel efficiency is not identical with the computational gain c_e defined in Chapter 4, Equation 4.31. Ideally, the weak scaling of the BSM in parallel would be unity, meaning that as the number of participating cores increases, the communication time remains the same. The computational gain is related to the reduced amount of the employed computational resources when compared to the fully embedded dual-mesh configuration. An ideal weak scaling does not necessarily imply satisfactory computational gain.



(a) Weak scaling



(b) Time comparisons

Figure D.3 Non-partitioned base domain

Additionally, parallel behaviour would affect the values of C_{map} . However, the above results should not be identified as the value of C_{map} . The values of both parameters are identified in Chapter 8 where more realistic test cases are considered.

D.3 Remarks

In this appendix the parallel BSM subroutine is analysed. The additional challenge was discussed and the communication approach was described.

In the majority of the applications, a single core (or node) would suffice for the calculations of the base domain, due to its coarse mesh properties. Since the BS calculations are directly applied only to the coarse mesh domain, no MPI communication is required for these calculations. As a result, the BS solver in parallel demonstrates the parallel efficiency of the parallel dual-mesh configuration.

Still, the possibility of a partitioned domain across a non-unity number of cores should be addressed. In this case, the core communication is required for the calculation of the TST_{CC} . The propagation of the source terms requires global information, due to the nature of the Fourier series, a trait that is detrimental to parallel efficiency. The positions in the algorithm of gathering and broadcasting may vary, affecting which information is broadcast and how many reduction steps take place in the master core. For the chosen approach (Method A), the size of the transferred information is dependent upon the number of the Fourier harmonics and the size of a single C_C (C_F) region, but independent of the total number of the C_C regions. This would allow the number of mapped regions to be extended with a small impact on the parallel efficiency. Nevertheless, the base domain should have a sufficiently low cell count, allowing its distribution across a small number of cores.

Appendix E

MPI Collective Commands

Collective MPI subroutines perform communications among all the cores belonging to the same communication group in an optimised manner. They do so through tree algorithms, as shown in Figure E.1. **MPI Broadcast** copies data from the root core to the rest of the cores, while **MPI Reduce** performs reduction calculations (e.g. finding max/min values, summation, multiplication etc.). When calculating the BSM spectra, a sub-communicator is established, thus only the cores with the relevant base blocks participate in the collective communications. For the **MPI Broadcast** and **MPI Reduce**, two points are shown:

- The number of data transfer events (number of communications) is approximately

$$N_{cores}/2.$$

- The number of data transfer **cycles** is following the $\log_2 N_{cores}$ law.

For **MPI All-Gather**, considering that during each core communication information is passed in both directions, the following is true:

- The number of data transfer events (number of communications) is $N_{cores} \times (N_{cores} - 1)$.

- The number of data exchange cycles is equal to N_{cores} , meaning the number of data transfer **cycles** is equal to $2N_{cores}$.

In any attempt to calculate the time needed to perform such actions, each cycle contributes just one data transfer event, since the data transfers in that cycle are ideally being performed simultaneously.

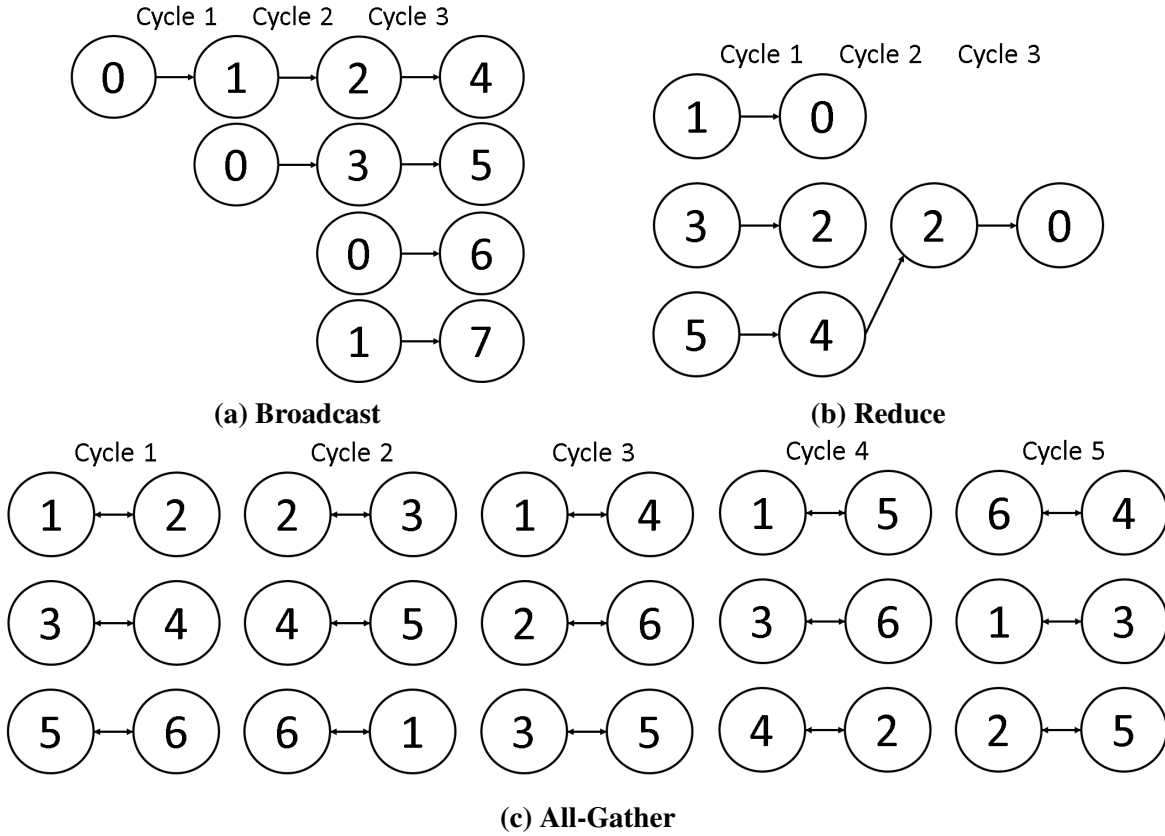


Figure E.1 Schematic of collective MPI commands. Each number denotes a core, with 0 being the master core. Each one-way arrow denotes a data transfer event, and each two-way arrow denotes a data exchange event. Each column of connecting lines denotes simultaneous data transfers for (a), (b) and simultaneous data exchanges for (c)

Appendix F

Configuration of the Distribution of Source Regions

In this appendix two examples of configuring the source regions distribution from Chapter 6 are presented. This discussion is introduced due to the periodic nature of the Fourier series, i.e. due to the implemented source term mapping performed by the selected basis function, and not due to the Block Spectral Method itself. Initially, a demonstration of configuring a periodic mapping is given, followed by an example considering a mirrored source regions distribution.

F.1 Periodic Mapping

This example is linked to the subsection 8.1.2 titled *Block Spectral Mapping Applied in the Lateral (Y) Direction*, where the flow patterns in the mapping direction vary periodically because the participating regions are perpendicularly distributed to the bulk flow. As a result, the first and the last regions are considered to be adjacent in the frequency domain. Here, an identical configuration is discussed, as shown in Figure F.1. All regions are assigned a

phase ID, i.e. all phases correspond to a real location of the simulated domain. The green regions effectively belong to the mapped regions, however since they are updated directly by the fine mesh, the mapped source terms are not taken into account in these regions. For $N_h = 1$ harmonic, the required number of source regions is $2 \times N_h + 1 = 3$. Since there are 18 regions, the phase step angle is $2\pi/18$ and the phase angle difference between consecutive source regions is equal to $2\pi/3$. In Table F.1 the phase angles, as numbered in Figure F.1, are reported.

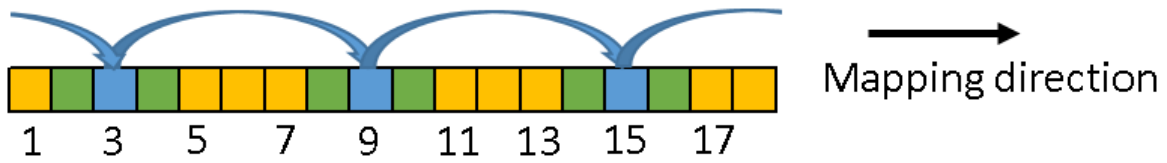


Figure F.1 Periodic mapping: Each square denotes a participating region. Blue, green and yellow denote C_F source regions, C_F non-source regions and C_C regions, respectively. Numbers denote phase ID

Table F.1 Periodic mapping: Phase angle of each region

Phase ID	1	2	3	4	5	6	7	8	9
Phase angle	0	$\frac{\pi}{9}$	$\frac{2\pi}{9}$	$\frac{\pi}{3}$	$\frac{4\pi}{9}$	$\frac{5\pi}{9}$	$\frac{2\pi}{3}$	$\frac{7\pi}{9}$	$\frac{8\pi}{9}$
Phase ID	10	11	12	13	14	15	16	17	18
Phase angle	π	$\frac{10\pi}{9}$	$\frac{11\pi}{9}$	$\frac{4\pi}{3}$	$\frac{13\pi}{9}$	$\frac{14\pi}{9}$	$\frac{5\pi}{3}$	$\frac{16\pi}{9}$	$\frac{17\pi}{9}$

F.2 Non-Periodic Mapping

Here, a configuration is discussed that requires mirrored source regions. The following example is linked to the subsection 8.1.1 titled *Block Spectral Mapping Applied in the Freestream (X) Direction*. Such configuration is needed when the last phase is physically located downstream of the first phase, i.e. there is an upstream history effect, hence it would not be physically sound to assume that the last phase is adjacent to the first phase in the

frequency domain. To bypass this issue, it is assumed that the regions participating in the BS mapping constitute a subgroup of a symmetric region distribution.

For example, let us assume a test case where 11 regions are considered for the BS mapping and one harmonic is employed to perform the mapping using the Fourier series. For the conventional configuration 3 source regions are required, with the last phase being adjacent to the first phase in the frequency domain. To satisfy the smooth variation of the region flow properties in the frequency domain, the 11 regions are mirrored in the frequency domain, i.e. 10 auxiliary regions are added downstream of the eleventh region, without being assigned a real location in the simulated domain. The added regions are a mirrored image of the phases of the actual domain phases, and their association is presented in Table F.2. In this way the smooth variation in the neighbourhood of the phase 1 in the frequency domain is ensured. This is illustrated in Figure F.2 by comparing between a periodic and a non-periodic source region distribution. In addition, the mirroring process is illustrated in Figure F.3. It is noted that the propagated source terms (TST) of the regions with phase ID ranging from 12 to 21 are not considered in the BS execution, as these regions do not correspond to an actual location in the physical domain. The step of the phase angles is $2\pi/21$ and the phase angle difference between consecutive source regions is equal to $2\pi/3$. In Table F.3 the phase angles, as numbered in Figure F.3, are reported.

Table F.2 Non-Periodic Mapping: Phases of the domain regions and their mirrored counterparts (auxiliary phases)

Domain phases	1	2	3	4	5	6	7	8	9	10	11
Mirrored phases	-	21	20	19	18	17	16	15	14	13	12

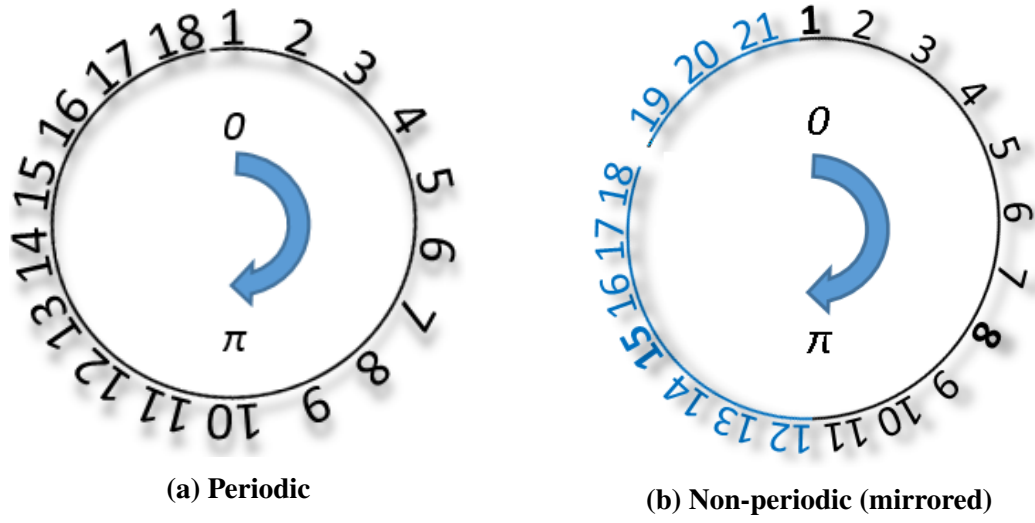


Figure F.2 Illustration of the regions locations in the frequency domain by comparison between a periodic and a mirrored configuration. IDs coloured in blue denote auxiliary phases

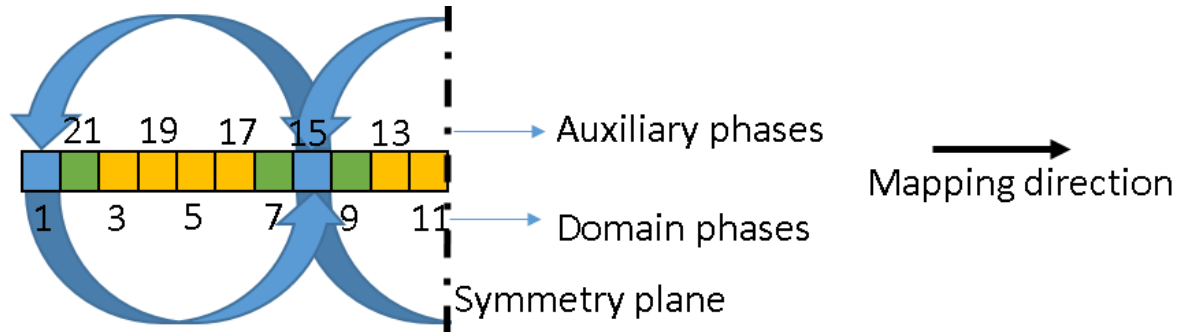


Figure F.3 Non-periodic Mapping: Each square denotes a participating region. Blue, green and yellow denote C_F source regions, C_F non-source regions and C_C regions, respectively. Numbers denote phase ID

Table F.3 Non-Periodic Mapping: Phase angle of each region

Phase ID	1	2	3	4	5	6	7	8	9	10	11
Phase angle	0	$\frac{2\pi}{21}$	$\frac{4\pi}{21}$	$\frac{2\pi}{7}$	$\frac{8\pi}{21}$	$\frac{10\pi}{21}$	$\frac{4\pi}{7}$	$\frac{2\pi}{3}$	$\frac{16\pi}{21}$	$\frac{6\pi}{7}$	$\frac{20\pi}{21}$
Phase ID	12	13	14	15	16	17	18	19	20	21	
Phase angle	$\frac{22\pi}{21}$	$\frac{8\pi}{7}$	$\frac{26\pi}{21}$	$\frac{4\pi}{3}$	$\frac{10\pi}{7}$	$\frac{32\pi}{21}$	$\frac{34\pi}{21}$	$\frac{12\pi}{7}$	$\frac{38\pi}{21}$	$\frac{40\pi}{21}$	