



City Research Online

City St George's, University of London

Citation: Meng, Xiangyin (2017). Coupling of nanofluid flow, heat transfer and nanoparticles sedimentation using OpenFOAM. (Unpublished Doctoral thesis, City, University of London)

This is the accepted version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

Permanent repository link: <https://city-ref-test.eprints-hosting.org/id/eprint/17350/>

Copyright and Reuse: Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to [City Research Online policy](#).



**COUPLING OF NANOFLUID FLOW, HEAT TRANSFER AND
NANOPARTICLES SEDIMENTATION USING OpenFOAM**

By

Xiangyin Meng

M.Eng, B.Eng

Supervisor

Prof. Qingwei Ma

A thesis submitted in fulfilment of

requirement of degree of

Doctor of Philosophy

School of Mathematics, Computer Science & Engineering

City, University of London

April, 2017

CONTENTS

CONTENTS.....	1
LIST OF FIGURES	6
LIST OF TABLES	10
ACKNOWLEDGEMENTS.....	11
DECLARATION.....	12
ABSTRACT.....	13
LIST OF SYMBOLS	15
1. INTRODUCTION	20
1.1 Background	20
1.1.1 Nanofluid and applications	20
1.1.2 Numerical investigations of nanofluid flow and heat transfer	22
1.1.3 OpenFOAM	23
1.2 Purposes and significance of this study	24
1.3 The main contributions of this study.....	25
1.4 Outline.....	25
2. LITERATURE REVIEW.....	26
2.1 Nanofluid preparation	26
2.1.1 One-step method	26
2.1.2 Two-step method and anti-sedimentation treatments	28
2.1.2.1 Two-step method.....	28
2.1.2.2 Adding surfactant to keep nanofluid stable	29
2.1.2.3 Performing physical agitations to disperse nanoparticles	30

2.2	Nanofluid thermal conductivity enhancement	33
2.2.1	Mechanisms	33
2.2.1.1	Brownian motion of nanoparticles.....	33
2.2.1.2	Liquid layering at particle-liquid interface	34
2.2.1.3	Nanoparticles clustering effects.....	34
2.2.2	Measurements of nanofluid thermal conductivity.....	35
2.3	Nanofluid heat transfer performance	36
2.3.1	Nanofluid natural convection.....	36
2.3.2	Nanofluid forced convection.....	39
2.4	CFD simulations for nanofluid flow and heat transfer.....	41
2.4.1	Different CFD approaches	41
2.4.2	Single-phase approach	42
2.4.3	Eulerian-Lagrangian approach.....	44
2.4.4	Eulerian approaches	45
2.5	Conclusions.....	48
3.	CFD BASICS OF FLUID FLOW AND HEAT TRANSFER.....	50
3.1	Introduction.....	50
3.2	Governing equations for CFD simulation.....	50
3.2.1	Continuity equation.....	50
3.2.2	Momentum equations.....	51
3.2.3	Energy equation	53
3.3	Main features of OpenFOAM for CFD.....	54
3.3.1	Tensors	55
3.3.2	Discretisation of computational domain and boundary conditions	56

3.4	Treatment of N-S equations in OpenFOAM	61
3.4.1	Pressure equation	62
3.4.2	SIMPLE, PISO and PIMPLE algorithms	63
3.5	Conclusions	65
4.	EULERIAN-LAGRANGIAN APPROACH FOR NANOFLUID SIMULATION WITH OpenFOAM	67
4.1	Introduction	67
4.2	Discrete element method (DEM) in OpenFOAM	68
4.2.1	Basic idea	68
4.2.2	Parcels	70
4.3	Forces consideration for nanoparticles	71
4.3.1	Surface and body forces	71
4.3.2	Particle-particle and particle-wall contacting forces	75
4.3.3	Order-of-magnitude analysis-the significance of DEM for nanofluid	78
4.4	Development of ‘nanofluidELFoam’	80
4.5	Solver test	83
4.6	Conclusions	88
5.	EULERIAN-MIXTURE APPROACH FOR NANOFLUID SIMULATION WITH OpenFOAM	89
5.1	Introduction	89
5.2	Mixture implementations in OpenFOAM	89
5.2.1	Mixture continuity equation	89
5.2.2	Mixture momentum equation	91
5.2.3	Continuity equation for the dispersed phase	92

5.2.4	Solver ‘driftFluxFoam’ and test	93
5.3	Development of ‘nanofluidMixtureFoam’	97
5.3.1	Basic idea	97
5.3.2	Solver development in OpenFOAM	98
5.4	Solver test.....	100
5.5	Conclusions.....	103
6.	APPLICATIONS OF ‘nanofluidMixtureFoam’ TO THREE TYPICAL CASES.....	104
6.1	Introduction.....	104
6.2	Nanofluid natural convection combining nanoparticles sedimentation in a two-dimensional square cavity.....	104
6.2.1	Introduction.....	104
6.2.2	Geometrical configuration and boundary conditions	105
6.2.3	Discussions and conclusions	107
6.3	Nanofluid natural convection combining nanoparticles sedimentation in a three-dimensional horizontal cylinder.....	112
6.3.1	Introduction.....	112
6.3.2	Geometrical configuration and boundary conditions	113
6.3.3	Discussions and conclusions	114
6.3.3.1	Approach1 for the three-dimensional horizontal cylinder	114
6.3.3.2	Coupling of nanofluid natural convection and nanoparticles sedimentation	117
6.4	Forced convection in a two-dimensional channel with open cavity	121
6.4.1	Introduction.....	121
6.4.2	Geometrical configuration and boundary conditions	122

6.4.3	Discussions and conclusions	125
6.5	Conclusions.....	128
7.	CONCLUSIONS AND RECOMMENDATIONS.....	130
7.1	Summary and conclusions	130
7.2	Recommendations and future development	132
	REFERENCE.....	134

LIST OF FIGURES

Fig. 1.1 TEM images of Au/de-ionized water nanofluid (Buongiorno et al., 2009)	20
Fig. 1.2 Typical radiator of a vehicle engine (Saidur et al., 2011)	21
Fig. 1.3 Schematic nanofluid-based direct absorption solar collector (Tyagi et al., 2009)...	21
Fig. 2.1 Stability of Al ₂ O ₃ /water nanofluid (without any stabilizers) changing with time (Wen et al., 2009).....	29
Fig. 2.2 A gravity settling experiment of 0.5 wt% Al ₂ O ₃ /water (Witharana et al., 2012)...	29
Fig. 2.3 Relationship between thermal conductivity k and nanoparticles cluster packing fraction ϕ (Kebllinski et al., 2002)	35
Fig. 2.4 Schematic experimental apparatus in (Putra et al., 2003).....	37
Fig. 2.5 Schematic experimental apparatus reported in (Hu et al., 2014).....	38
Fig. 2.6 Experimental setup reported in (Wen and Ding, 2004)	39
Fig. 3.1 Finite control volume in space (Anderson Jr, 2009).....	51
Fig. 3.2 Right handed Cartesian coordinate system.....	55
Fig. 3.3 Discretisation of solution and time domains (Guide, 2011)	57
Fig. 3.4 Key parameters in finite volume discretisation (Guide, 2011)	57
Fig. 3.5 Schematic mesh description in OpenFOAM	58
Fig. 4.1 Flow chart of solver ‘icoUncoupledKinematicParcelFoam’	69
Fig. 4.2 Two particles contacting with overlap δ	76
Fig. 4.3 Basic idea of solver ‘nanofluidELFoam’	81
Fig. 4.4 An extreme case with very few numerical particles in computational region	82
Fig. 4.5 The geometry and uniform mesh strategy with 50×50 celss	84
Fig. 4.6 The initialized 40000 particles in cavity at $t = 0s$	84
Fig. 4.7 Temperature features with (a) and without (b) colliding particles.....	85

Fig. 4.8 Particles distribution is changed by temperature driven flow.....	86
Fig. 4.9 The comparison between theoretical Poisson distribution and particles distribution predicted by ‘nanofluidELFoam’	86
Fig. 4.10 Non-uniform particles distribution	87
Fig. 4.11 Non-uniform nanofluid thermal conductivity due to non-uniform particles distribution.....	87
Fig. 5.1 The flow chart of solver ‘driftFluxFoam’	94
Fig. 5.2 Mesh strategy of the two-dimensional square enclosure	95
Fig. 5.3 The sedimentation layer heights at $t = 7 \text{ hours}$ given by strategies 50×50 and 60×60 in grid independence check.....	95
Fig. 5.4 The sedimentation simulation of 0.64% $\text{Al}_2\text{O}_3/\text{water}$ nanofluid in a two-dimensional cavity by solver ‘driftFluxFoam’	96
Fig. 5.5 Stability of 0.64% $\text{Al}_2\text{O}_3/\text{water}$ nanofluids (without any stabilizer) changing with time (Wen et al., 2009).....	97
Fig. 5.6 Basic idea of solver ‘nanofluidMixtureFoam’	98
Fig. 5.7 The flow chart of solver ‘nanofluidMixtureFoam’	100
Fig. 5.8 Schematic features (a) and mesh strategy (b) for the cavity case	101
Fig. 5.9 Velocity and temperature features in the first test ($Ra = 1 \times 10^4$)	102
Fig. 5.10 Dimensionless velocity component u^* along $x = L/2$ in the first test	102
Fig. 5.11 Dimensionless velocity component v^* along $y = L/2$ in the first test.....	102
Fig. 5.12 Dimensionless T^* along $y = L/2$ in the first test.....	103
Fig. 6.1 Grid independence check ($Y = 0.5L$, $Ra = 107$)	107
Fig. 6.2 0.64% $\text{Al}_2\text{O}_3/\text{water}$ nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^6$ ($t = 7 \text{ hours}$).....	108
Fig. 6.3 0.64% $\text{Al}_2\text{O}_3/\text{water}$ nanofluid temperature driven flow and nanoparticles	

sedimentation interactions in case $Ra = 5.0 \times 10^6$ ($t = 7$ hours).....	108
Fig. 6.4 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^7$ ($t = 7$ hours).....	109
Fig. 6.5 Average Nusselt number against Rayleigh number of 0.64% Al_2O_3 /water nanofluid	109
Fig. 6.6 Temperature contours comparison of 0.64% nanofluid at $Ra = 1.0 \times 10^6$ ($t = 7$ hours).....	111
Fig. 6.7 Temperature contours comparison of 0.64% nanofluid at $Ra = 5.0 \times 10^6$ ($t = 7$ hours).....	111
Fig. 6.8 Temperature contours comparison of 0.64% nanofluid at $Ra = 1.0 \times 10^7$ ($t = 7$ hours).....	112
Fig. 6.9 The schematic model of the cylinder in Gmsh 2.9.2	113
Fig. 6.10 The non-uniform mesh strategy for the horizontal cylinder	113
Fig. 6.11 Temperature distribution with a three-dimensional view, $Ra = 1.0 \times 10^6$ ($t = 7$ hours).....	115
Fig. 6.12 Temperature distributions on central planes of two- and three-dimensional cases, $Ra = 1.0 \times 10^6$ ($t = 7$ hours).....	115
Fig. 6.13 Temperature T along $y = 0.02m$, $Ra = 1.0 \times 10^6$ ($t = 7$ hours).....	115
Fig. 6.14 Temperature T along $y = 0.02m$, $Ra = 5.0 \times 10^6$ ($t = 7$ hours).....	116
Fig. 6.15 Temperature T along $y = 0.02m$, $Ra = 1.0 \times 10^7$ ($t = 7$ hours).....	116
Fig. 6.16 Velocity U_y along $y = 0.02m$, $Ra = 1.0 \times 10^6$ ($t = 7$ hours)	117
Fig. 6.17 Velocity U_y along $y = 0.02m$, $Ra = 5.0 \times 10^6$ ($t = 7$ hours)	117
Fig. 6.18 Velocity U_y along $y = 0.02m$, $Ra = 1.0 \times 10^7$ ($t = 7$ hours)	117
Fig. 6.19 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^6$ ($t = 7$ hours).....	118

Fig. 6.20 0.64% Al ₂ O ₃ /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 5.0 \times 10^6$ ($t = 7$ hours).....	118
Fig. 6.21 0.64% Al ₂ O ₃ /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^7$ ($t = 7$ hours).....	119
Fig. 6.22 Average Nusselt number against Rayleigh number predicted by different approaches ($t = 7$ hours)	119
Fig. 6.23 Average Nusselt number comparison between two- and three-dimensional cases ($t = 7$ hours).....	120
Fig. 6.24 Temperature contours comparison at $Ra = 1.0 \times 10^6$ ($t = 7$ hours)	121
Fig. 6.25 Temperature contours comparison at $Ra = 5.0 \times 10^6$ ($t = 7$ hours)	121
Fig. 6.26 Temperature contours comparison at $Ra = 1.0 \times 10^7$ ($t = 7$ hours)	121
Fig. 6.27 Schematic of the channel with an open cavity heated at bottom	123
Fig. 6.28 Geometrical variations of the cavity with heating bottom.....	123
Fig. 6.29 Different geometry models created by Gmsh	124
Fig. 6.30 Non-uniform scheme for mesh in following cases	124
Fig. 6.31 Heating bottom temperatures of the nine geometries ($Re = 500$)	126
Fig. 6.32 Heating bottom temperatures of the nine geometries ($Re = 1000$).....	127

LIST OF TABLES

Tab. 1.1 Some basic solvers officially provided by OpenFOAM 2.3.1	24
Tab. 2.1 Summary of some nanofluids prepared with physical agitation	30
Tab. 3.1 Basic tensor classes in OpenFOAM.....	56
Tab. 3.2 Seven units in OpenFOAM.....	56
Tab. 3.3 Four lists in OpenFOAM to describe mesh.....	57
Tab. 3.4 Stored data in ' <i>fvMesh</i> ' class.....	58
Tab. 3.5 Differencing schemes for face flux estimation in OpenFOAM	60
Tab. 3.6 Two boundary types in OpenFOAM.....	61
Tab. 4.1 Features comparison for the three solvers.....	67
Tab. 4.2 Interaction considerations in different coupling approaches.....	83
Tab. 4.3 Water fluid and Al ₂ O ₃ particle properties.....	85
Tab. 5.1 Solvers features	89
Tab. 5.2 Key parameters for solver 'nanofluidMixtureFoam' validation.....	101
Tab. 6.1 Summary of the four approaches in this section	105
Tab. 6.2 Codes of the nine geometrical models	123

ACKNOWLEDGEMENTS

In the end of my PhD study, I want to express my sincerely thanks to my supervisor, Prof Qingwei Ma at City University London. Without his continuous kind help since my first day in London, I am afraid I could not finish this work at all. It is definitely not a compliment to say that, Prof Qingwei Ma has changed my life, as well as my family's future.

Also, I would like to thank Dr Shiqiang Yan and the whole Hydrodynamic Research Group at School of Mathematics, Computer Science & Engineering, City, University of London. I would never forget the memorable four years we spent together in London. Thank you so much for turning me into an outgoing team-worker.

And, I must say 'thank you' to my dear wife, Dr Hui Yan and my parents in China, who are always behind me with the strongest support in the world. It is really hard to calculate how much do I owe you. But the thing I know with no doubt is, you are definitely my power source to go ahead.

I also want to express my sincerely appreciations to all my colleagues at School of Marine Science and Technology, Newcastle University. Without your help, I would not be able to get the balance between my teaching loads and research work. You are the best colleagues to work with.

DECLARATION

No portion of the work referred to in this thesis has been submitted in support of an application for other degree or qualification of this or any other university or other institute of learning.

I grant powers of discretion to City University Library to allow this thesis to be copied in whole or in part without any reference to me. This permission covers only single copies made for study purpose subject to normal condition of acknowledgement.

ABSTRACT

Nanofluid is a suspension containing a certain quantity of nanoscaled solid particles in a conventional cooling liquid. Compared to pure liquid in micro channels, nanofluid shows notably better heat transfer performance but without erosion and clogging problems as normal two-phase suspensions. Due to such advantages, nanofluid is increasingly applied as an ideal coolant in engineering. For a better understanding of nanofluid flow and heat transfer performance, many investigations have been carried out recently in both experimental and numerical ways.

In numerical investigations, computational fluid dynamics (CFD) is playing a dominant role due to its maturity in the area of fluid flow and heat transfer research. However, in previous CFD studies, the problem of nanoparticles sedimentation is always ignored based on the assumption that nanofluid is stable with homogeneous properties throughout the simulation. To some extreme cases in which nanoparticles sedimentation would happen soon after nanofluid preparation, such assumption could induce inaccurate numerical results.

To investigate the relationships between nanofluid flow, heat transfer and nanoparticles sedimentation, an open source CFD package, OpenFOAM is employed as the basis to develop several numerical solvers in multi-phase way for the first time. More specifically, nanofluid CFD simulations are carried out by several newly developed OpenFOAM solvers under both Eulerian-Lagrangian and Eulerian-Mixture (a simplified Eulerian-Eulerian approach) frames. By comparing present numerical results to previous published experimental and numerical investigations, it can be concluded that the newly developed solvers under both Eulerian-Lagrangian and Eulerian-Mixture frames are capable to investigate nanofluid flow and heat transfer performance coupling with nanoparticles sedimentation. However, with the considerations of computational resource requirement, Eulerian-Mixture approach is believed to be better to achieve the balance between accuracy and computational effort.

With an assumption that no appropriate stabilizing treatments have been applied after nanofluid preparation, CFD simulations are carried out for 0.64% Al_2O_3 /water nanofluid in three most typical geometries by the newly developed solver ‘nanofluidMixtureFoam’. According to the present research, it can be confirmed that nanofluid heat transfer and nanoparticles

sedimentation have considerable impacts to each other in nanofluid natural convections (in both two- and three-dimensional cases). More specifically, temperature driven flow leads to thicker nanoparticles sedimentation layer than that in normal sedimentation case. On the other hand, nanoparticles sedimentation layer induces worse nanofluid natural convection heat transfer performance. Furthermore, for forced convection problems in a horizontal channel with an open cavity, nanoparticles sedimentation is likely to occur at cavity bottom and leads to higher temperature at heating surface. For better heat transfer performance of the cooling blocks with similar geometries, lower fins (cavity depths) in blocks are recommended to reduce possible nanoparticles sedimentation. In summary, the newly developed OpenFOAM solvers and numerical observations in this thesis are expected to guide future nanofluid CFD study and correlative practical applications.

LIST OF SYMBOLS

a	acceleration (ms^{-2}) Lattice constant (nm) Matrix constant in equation discretisation
$\mathbf{a}$	Tensor vector
A	Square matrix storing the coefficients for a set of algebraic equations Settlement coefficient
A_p	Particle projection area in moving direction (m^2)
Ag	Silver
Ag^-	Silver ion
$AgNO_3$	Silver nitrate
Al_2O_3	Alumina
Au	Gold
b	Collision constant
$\mathbf{b}$	Tensor vector
B	Source vector at the right hand side of an algebraic equation
c_d	Drag coefficient
c_p	Specific heat capacity ($Jkg^{-1}K^{-1}$)
C	Cell centre
C_c	Cunningham correction factor
C_f	Face centre
C_μ	Adjustable turbulent constant
CPU	Central processing unit
CFD	Computational fluid dynamics
Cu	Copper
CuO	Copper oxide
d	Particle diameter (m) Dimension of tensor
D	Cylinder diameter (m)
e	Internal energy of fluid element (J)
E	Young's modulus (GPa)
EG	Ethylene glycol
F	Acting force (N) Particle mass fraction
F_C	Contact force (N)
F_D	Drag force (N)
F_{LS}	Saffman lift force (N)
F_{LM}	Magnus lift force (N)
F_P	Acting force between particles (N)
F_W	Acting force between particle and wall (N)

g	Gravity acceleration (ms^{-2})
g_b	Fixed gradient on boundary
g_k	Boussinesq body force ($kgm^{-2}s^{-2}$)
G	Gravity force (N)
Gr	Grashof number
h	Heat transfer coefficient ($Wm^{-2}K^{-1}$)
h_{nl}	Nanolayer thickness (m)
H	Height (m)
H_2	Hydrogen
HNO_3	Nitric acid
H_2O	Water
i_x	Unit base vector
i_y	Unit base vector
i_z	Unit base vector
I	Inertia (kgm^2) RANS turbulent kinetic energy (m^2s^{-2})
k	Thermal conductivity ($Wm^{-1}K^{-1}$) RANS turbulent kinetic energy (m^2s^{-2}) Spring stiffness for particles collision calculation (N/m)
l	Mean free path of a phonon (m)
L	Length (m)
m	Mass (kg)
M_s	Extra term in momentum equation, presenting the effects from dispersed solid phase
n	Particle number Particle size spread factor
n_{ij}	Unit direction vector pointing from one particle to the other
N	Number of assemble members for Reynolds averaging
NaH_2PO_2	Sodium hypophosphite
NaH_2PO_3	Sodium dihydrogen phosphite
Nu	Nusselt number
p	Pressure ($kgm^{-1}s^{-2}$)
$\tilde{p}$	Kinematic pressure (m^2s^{-2})
p^*	Pressure combining with gravity ($kgm^{-1}s^{-2}$)
pH	Measurement of acidity or alkalinity level
Pr	Prandtl number
q	Heat flux (Ws^{-2})
$\dot{q}$	Volumetric heat addition rate per unit mass (W/m)
r	Particle size (m) Rank of tensor

rpm	Revolutions per minute
R	RANS Reynolds stress tensor Radius (m)
Ra	Rayleigh number
Re	Reynolds number
s	Tensor scalar
S	Finite control volume surface area (m^2)
S_e	Source term for energy
S_f	Face area vector
S_m	Source term for momentum
S_p	Implicit coefficient for force calculation on particle
S_U	Explicit contribution for force calculation on particle
SDS	Sodium dodecylsulfate
SDBS	Dodecylbenzenesulfonate
SiO_2	Silica
t	Time (s)
T	Temperature (K) Transpose operation Torque (Nm) Total number of nanoparticles
T^*	Dimensionless temperature
ΔT	Temperature difference (K)
T_0	Reference temperature (K)
T_C	Temperature of cooling wall (K)
T_H	Temperature of heating wall (K)
T_m	Melting temperature (K)
TiO_2	Titanium dioxide
T_{Ref}	Reference temperature (K)
u	Velocity component in x direction (ms^{-1})
$\bar{u}$	Averaged velocity (ms^{-1})
u'	Fluctuation velocity (ms^{-1})
u^+	Dimensionless velocity in wall function
U	Velocity (ms^{-1})
U_{max}	Maximum velocity (ms^{-1})
U_0	Settlement velocity (ms^{-1})
U_y	Vertical velocity (ms^{-1})
U_y^*	Dimensionless vertical velocity
v	Velocity component in y direction (ms^{-1})
V	Finite control volume (m^3)
δV	Volume of cell (m^3)
w	Velocity component in z direction (ms^{-1})
x	Column vector of dependent variable

	Direction component of x
$\bar{x}$	Brownian motion range (m)
X	Horizontal position (m)
X^*	Dimensionless length
y	Direction component of y Distance from wall (m)
y^+	Dimensionless wall distance in wall function
Y	Vertical position (m)
z	Direction component of z
Greek symbols	
α	Thermal diffusivity (m^2s^{-1}) Critical damping ratio
α_U	Relax coefficient for velocity
α_P	Relax coefficient for pressure
β	Thermal expansion coefficient (K^{-1})
γ	Gruneisen parameter Viscous damping constant ($Nm^{-1}s$)
δ	Grid expansion ratio Particles contacting overlap (m)
ε	Turbulent dissipation rate (m^2s^{-3})
κ	Von Karman constant
κ_B	Boltzmann constant ($m^2kgs^{-2}K^{-1}$)
λ	Secondary viscosity coefficient ($kgm^{-1}s^{-1}$) Particle mean free path (m)
μ	Dynamic viscosity ($kgm^{-1}s^{-1}$)
ν	Kinetic viscosity (m^2s^{-1}) Poisson ratio
ρ	Density (kgm^{-3})
π	Pi, 3.1415926
τ	Stress (shear) force (N)
ϕ	Nanofluid volume fraction
ϕ_f	Face field
ϕ_b	Fixed value on boundary
ϕ_g	Face flux
ω	Rotation velocity ($revs^{-1}$)
Subscripts	
a	Additional
eff	Effective
f	Fluid
i	Component indicator Particle index
j	Component indicator Particle index

k	Component indicator Kinematic
m	Mixture
n	Nanoparticle
nf	Nanofluid
nl	Nanolayer
N	Neighbour
P	Point
s	Solid
t	Turbulent
Units	
kg	Kilogram
K	Kelvin
m	Metre
N	Newton
s	Second
W	Watt

1. INTRODUCTION

1.1 Background

1.1.1 Nanofluid and applications

Nanofluid is a sort of solid-liquid composite material consisting of nanometre sized solid particles, fibres, rods or tubes suspended in different basefluids (Fig. 1.1), providing a promising technical selection for enhancing heat transfer performance due to its anomalously high thermal conductivity (Chol, 1995). Because of Brownian motion and interaction of nanoparticles, nanofluids represent improved stability compared to those conventional fluids containing micrometre- or millimetre-sized solid particles (Buongiorno et al., 2009). Therefore, by using nanofluid instead of traditional pure liquid as coolant, the sizes of heat transfer systems can be reduced while high heat transfer efficiency still can be obtained (Buongiorno et al., 2009, Jang and Choi, 2006). In recent years, nanofluids began to be used in many practical engineering applications such as transportation, solar device and electronics cooling, etc (Yu et al., 2007).

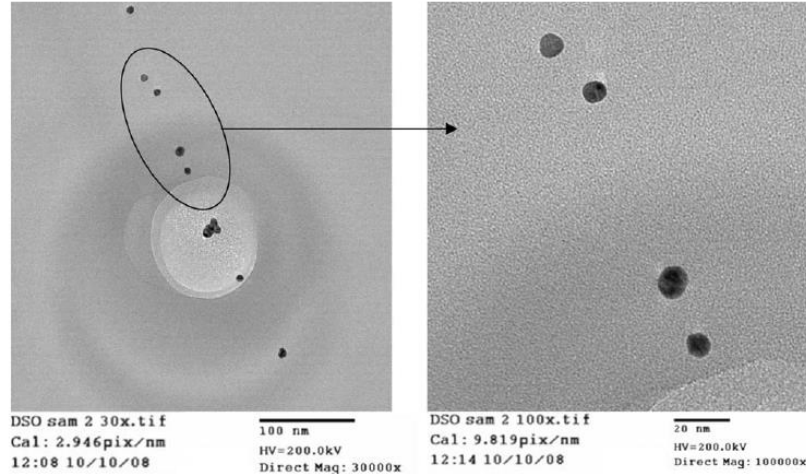


Fig. 1.1 TEM images of Au/de-ionized water nanofluid (Buongiorno et al., 2009)

In transportation, ethylene glycol (EG) and water mixture is the most commonly used automotive coolant for engine systems (Fig. 1.2). Because such mixture usually has a low freezing point, it is suitable for applications in cold weather without freezing issues. However, due to the poor thermal conductivity of EG ($k \approx 0.258 \text{ W/mK}$), how to enhance the heat transfer performance of EG/water mixture is always an interesting topic. Today, adding nanoparticles to engine coolant has potential to improve engine cooling rates (Liu et al., 2005). Such improvement

can be applied to remove engine heat with a smaller sized cooling system, which would result in smaller radiators. This concept is believed to be beneficial to enhance vehicle performance and fuel economy considerably (Kole and Dey, 2010).

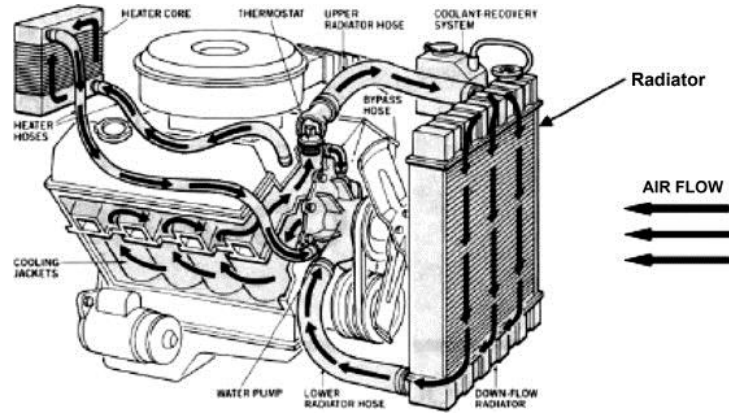


Fig. 1.2 Typical radiator of a vehicle engine (Saidur et al., 2011)

In solar device (Fig. 1.3), the application of nanofluid in collectors and water heaters is potential to enhance the heat transfer performance of working fluid. In an investigation of solar collector working principles which was performed by Tyagi et al. (2009), it was reported that adding low volume fraction of nanoparticles to working fluid could remarkably increase the solar collector working efficiency. They attributed the efficiency enhancement to the increase in attenuation of sunlight passing through the collector due to nanoparticles addition. The similar benefits were also reported by some other researchers recently (Otanicar et al., 2010, Mahian et al., 2013).

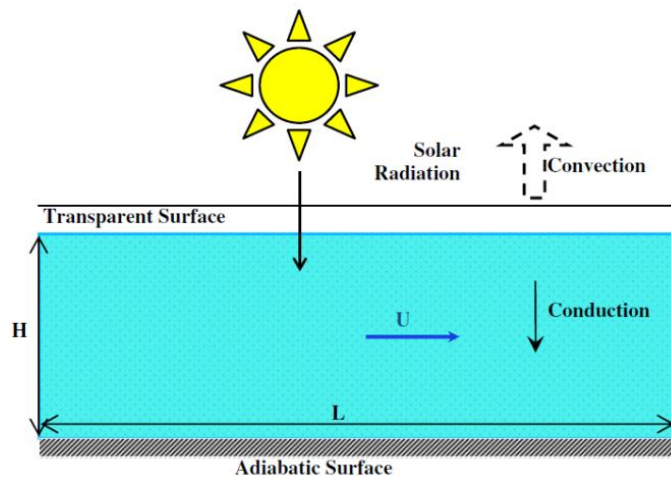


Fig. 1.3 Schematic nanofluid-based direct absorption solar collector (Tyagi et al., 2009)

In electronics cooling systems, improvement of coolant thermal performance has always been

a great technical challenge. Taking computer central processing unit (CPU) as an example, because the system reliability reduces by nearly 50% when CPU temperature increases every 10°C (Moore et al., 2004). It is estimated that a high performance CPU would dissipate power in a range of 100~300 W/cm^2 by 2018 (Pradeep and Ashokreddy, 2012). To maintain and control CPU temperature at a tolerable level, nanofluid is believed to be a better coolant than air and pure liquids (Paisarn and Somchai, 2011). To a CPU with power of 80 W, Tsai et al. (2004) reported that the temperature could be reduced from 40.9°C to 24.3°C by adding 1% volume fraction gold nanoparticles to pure water coolant. In 2013, Reserator 3, the first commercial CPU liquid cooler using nanofluid was launched by Zalman Tech. Co (<http://www.zalman.com>). Compared to other types of cooling device, this system might be the best one in future to reduce the temperature of high-performance CPUs.

1.1.2 Numerical investigations of nanofluid flow and heat transfer

The apparatuses needed for nanofluid experimental investigations are usually expensive. Particularly to some micro-sized channels and pipes, it is nearly impossible to set up appropriate experimental devices for every specific practical problem (Wang and Mujumdar, 2008b). Due to above reasons, numerical investigations are increasingly more popular for predictions of nanofluid flow and heat transfer. In this area, compared to molecule dynamics approach (Nie et al., 2004, Sarkar and Selvam, 2007, Mohebbi, 2012, Cui et al., 2015) and Lattice-Boltzmann approach (Xuan and Yao, 2005, Nematì et al., 2010, Ashorynejad et al., 2013), CFD is applied more widely in nanofluid numerical simulations (Wang and Mujumdar, 2008a).

In previous CFD investigations, single-phase approach was employed very often, in which nanofluid is assumed to be stable with homogenous and uniform properties (Kamýar et al., 2012). However, in the past decade, some experiments indicated that the assumption ‘nanofluid is stable and homogeneous’ could be invalid in some cases. More specifically, if no appropriate dispersion treatments and stabilizers were applied after nanofluid preparation (or manufacture), nanoparticles sedimentation could be observed in a short period after preparation (Wen et al., 2009, Witharana et al., 2012, Drzazga et al., 2012). Because the nanoparticles sedimentation layer has higher volume fraction/concentration, it will induce the situation that nanofluid properties are not uniform anymore in the whole computational region. This could lead to different numerical

results of nanofluid flow and heat transfer performance.

Therefore, in those extreme nanofluid cases without stabilizing treatments, it is necessary to consider the factor of possible nanoparticles sedimentation during nanofluid CFD simulations. Regarding this consideration, a numerical work attempting to couple nanofluid flow, heat transfer and nanoparticles sedimentation is presented in this thesis. Based on present work, CFD simulations for nanofluid flow and heat transfer are believed to be more practical and reliable for those cases in which nanoparticles sedimentation would happen soon after nanofluid preparation.

1.1.3 OpenFOAM

Compared to popular commercial CFD software such as ANSYS FLUENT, CFX and Star CCM+ etc (Glatzel et al., 2008), OpenFOAM is believed to be a better option to develop ‘customised’ numerical solvers due to its feature of ‘open’. OpenFOAM, which presents ‘Open Field Operation and Manipulation’, is an open source CFD software package started by Dr. Hrvoje Jasak and currently maintained by OpenCFD Ltd (Jasak, 2009). OpenFOAM has an extensive range of features to solve complex fluid flows involving chemical reactions, turbulence and heat transfer. It also includes tools for meshing (such as snappyHexMesh for complex CAD geometries) and post-processing respectively. Furthermore in OpenFOAM, simulations are possible to run in parallel and users are able to take full advantage of computer hardware at their disposal (Jasak et al., 2013).

By being ‘open’, OpenFOAM offers users great freedom to customise and extend existing functionalities. It follows a highly modular code design in which collections of functionalities (e.g. numerical methods and physical models) are compiled into their own shared libraries, to which executable applications can be linked and correlative functions can be called simply. At moment, OpenFOAM includes more than 80 solvers for various specific problems in engineering mechanics (such as flow, heat transfer and combustion, etc) and over 170 utility applications for pre- and post-processing tasks (such as meshing and data visualisation, etc) (Andersson, 2011). In this thesis, several solvers originally provided by OpenFOAM 2.3.1 will be mentioned (Tab. 1.1), and some of these solvers such as ‘buoyantBoussinesqPimpleFoam’, ‘icoUncoupledKinematicParcelFoam’ and ‘driftFluxFoam’ will be used as the basis for new

solvers development.

Tab. 1.1 Some basic solvers officially provided by OpenFOAM 2.3.1

Solver name	Suitable for
icoFoam	Transient solver for incompressible, laminar flow of Newtonian fluids
pisoFoam	Transient solver for incompressible flow
pimpleFoam	Large time-step transient solver for incompressible, flow using PIMPLE (merged PISO-SIMPLE) algorithm
buoyantBoussinesqPimpleFoam	Transient solver for buoyant, turbulent flow of incompressible fluids
icoUncoupledKinematicParcelFoam	Transient solver for the passive transport of a single kinematic particle cloud
driftFluxFoam	Transient solver for two incompressible fluids using Mixture approach with drift-flux approximation for relative motion of the phases.

1.2 Purposes and significance of this study

This project aims to investigate the relationships between nanofluid flow, heat transfer and nanoparticles sedimentation by appropriate CFD approaches. Under OpenFOAM frame, several multi-phase solvers are developed for nanofluid simulations in both Eulerian-Lagrangian and Eulerian-Mixture ways. After necessary validations, the two approaches are discussed and a better option is selected to carry out following CFD simulations for some typical cases in nanofluid heat transfer investigation.

For above purposes, the main tasks of this thesis can be summarised as follows:

1. To review and summarise recent investigations in nanofluid manufacture, heat transfer performance and CFD simulations.
2. To analyse the critical CFD implementations in OpenFOAM as the basis for new solvers development in this project.
3. To develop new solvers which can couple nanofluid flow, heat transfer and nanoparticles sedimentation in both Eulerian-Lagrangian and Eulerian-Mixture ways.
4. To choose an appropriate solver for the investigations of relationships between nanofluid flow, heat transfer and nanoparticles sedimentation in several typical cases.

1.3 The main contributions of this study

In previous CFD investigations, nanofluid is usually assumed to be a stable suspension with uniform properties. However, this is believed not significantly practical to some extreme cases in which nanoparticles sedimentation can be observed soon after preparation. This project aims to investigate the relationships between nanofluid flow, heat transfer and nanoparticles sedimentation by applying newly developed OpenFOAM solvers in multi-phase ways. The contributions of this study mainly include:

1. For the first time, nanoparticles sedimentation is coupled to CFD simulations of nanofluid flow and heat transfer.
2. Under OpenFOAM frame, several new solvers combining nanoparticles sedimentation and nanofluid flow and heat transfer are developed in both Eulerian-Lagrangian and Eulerian-Mixture ways.
3. A complete open source approach using free tools Gmsh, OpenFOAM and paraFoam is developed and tested for nanofluid flow simulations in both two- and three-dimensional cases.

1.4 Outline

In this thesis, the main contents can be summarised as follows:

Chapter 2	Retrospect previous studies in nanofluid manufacture, thermal conductivity, heat transfer performance and CFD simulations. Address the problems in previous nanofluid CFD investigations.
Chapter 3	Illustrate basic principles for fluid flow, heat transfer, main features of OpenFOAM and the treatments of Navier–Stokes (N–S) equations in OpenFOAM. Introduce previous typical nanofluid CFD simulations under different frames.
Chapter 4	Develop a new OpenFOAM solver in Eulerian-Lagrangian way to combine nanoparticles motion to nanofluid flow and heat transfer. Validate the newly developed solver by previous published investigations.
Chapter 5	Develop a new OpenFOAM solver in Eulerian-Mixture way to consider nanoparticles sedimentation in nanofluid flow and heat transfer simulation. Validate the newly developed solver by previous published investigations.
Chapter 6	Apply appropriate solvers to disclose the relationships between nanofluid heat transfer and nanoparticles sedimentation.
Chapter 7	Summarise the whole project and draw up appropriate conclusions. Give recommendations for future developments of the newly developed solvers.

2. LITERATURE REVIEW

In the past two decades, many investigations in nanofluid have been published (Das et al., 2006). In this chapter, a literature review is carried out to retrospect previous nanofluid research. Following what, the significance and necessity of present work are presented. However, it should be noticed that this literature review is not aiming to cover every corner of nanofluid-related research. Instead, it only focuses on the topics such as nanofluid preparation, nanofluid thermal conductivity enhancement, nanofluid heat transfer of natural and forced convections, nanofluid CFD simulations and previous OpenFOAM applications.

2.1 Nanofluid preparation

2.1.1 One-step method

One-step method was developed by Akoh et al. (1978) to produce dry nanoparticles from fluids and was named as ‘vacuum evaporation onto a running oil substrate (VEROS) technique’. After necessary developments and modifications, this method was used to produce nanofluids directly. In nanofluid preparation, ‘one-step’ means nanoparticles are dispersed into basefluids directly in the course of preparation (Komarneni et al., 1997). In one-step method, because nanoparticles are added into basefluids directly at the same time when they are produced, the storage and transportation for nanoparticles are not necessary in the whole manufacturing procedure. Because nanoparticles will not be oxidized by air during the whole process, one-step method is suitable for those applications requiring pure metal nanoparticles strictly, particularly in those experiments aiming to validate theoretical studies.

Based on VEROS technique, Choi and Eastman (2001) invented a direct method to produce and disperse nanoparticles into fluid. In their way, substance is heated to a sufficient high temperature for dispersion in a vacuum environment while passing a thin film of fluid near the heated substance. This method was registered as a patent in America in 2001. Even to date, this system is still considered as one of the most important and significant methods in nanofluids preparation.

To tackle the problem of nanoparticles aggregation in conventional methods, Chang et al.

(2005) combined arc spray nanoparticle synthesis system (ASNSS) with ultrasonic vibration and rotating electrode to prepare suspensions of TiO₂, CuO and Cu (Lo et al., 2005) nanoparticles. ASNSS mainly comprises a heating system, an ultrasonic system, a pressure control system and a temperature control system. This method is believed to be effective to avoid particles aggregation. By ASNSS, it was reported that the nanoparticles in suspension are distributed more uniformly. Furthermore, being able to control particles size easily is another advantage of this method.

In order to synthesise non-agglomerated and stable suspended silver nanofluids, another novel one-step method was proposed by Sudhan and Meenakshi (2011). In their method, chemical reaction $\text{NaH}_2\text{PO}_2 + 3\text{H}_2\text{O} + \text{AgNO}_3 \rightarrow \text{NaH}_2\text{PO}_3 + \text{HNO}_3 + \text{H}_2 + \text{Ag}^-$ was applied to obtain silver nanofluid directly. As a developed VEROS technique, this method combines preparations of nanoparticles and nanofluid together. Hence the process of drying, storage, transportation and re-dispersion of Ag nanoparticles can be avoided and the production cost can be reduced ultimately. Furthermore, because this process can be finished just after a chemical reaction, it is believed to be advantageous with high yield of production but less time consumption.

Chen and Wen (2011) also proposed a novel ultrasonic-aided one-step method for the fabrication of gold nanofluids (GNP). In their method, the 'Citrate Reduction' method (Cooper & Stevenson, 1951) and 'Brust-Schiffrin' method (Brust et al., 1994) were used to control the size of GNPs. Furthermore, the ultra-sonication was used to control the particle morphology. In their experiments, both spherical- and plate-shaped GNPs with a scale range of 10~300nm were synthesized and the ultra-sonication was demonstrated as a powerful tool in controlling the size and shape of GNPs.

From above operations for nanofluid preparation, it can be found that one-step method has many advantages such as fabricating nanofluids with high purity, high stability and small average nanoparticle size, etc. However, to make one-step method more popular in practical engineering, further attempts are still needed on the topics such as how to produce different nanofluids with greater range of volume fraction and quantity, especially for those basefluids with high vapour pressure (Buongiorno et al., 2009).

2.1.2 Two-step method and anti-sedimentation treatments

2.1.2.1 Two-step method

Two-step method means nanoparticles are prepared or purchased independently in the form of dry powders firstly and added into basefluids as the second step. Two-step method usually needs some external treatments such as ultrasonic bath and surfactant additions to ensure nanofluids are stable enough after preparation. Comparatively, two-step method is used more widely than one-step method because almost all kinds of the nanoparticles with different shapes and sizes can be purchased from professional companies (Drzazga et al., 2012), such as Sigma-Aldrich Co. Ltd (Kim et al., 2007c), Carbon Nanotechnology Inc. (Douroumis et al., 2007), Nanophase Co. Ltd (Das et al., 2003b) and Applied Nanoworks INC (Kim et al., 2007d), etc. Furthermore, compared to one-step method, two-step method is easier to operate with simpler devices.

However, after two-step method preparation, nanoparticles sedimentation would happen very soon if without appropriate stabilizing treatments. Generally, several ways such as UV–Vis spectrophotometer, zeta potential, sediment photograph capturing, TEM and SEM, light scattering, three omega and sedimentation balance methods can be applied to measure nanofluid stability. Therefore, the rate or percentage of sedimentation will be identified by analyzing the gathered data (Ghadimi et al., 2011).

Wen et al. (2009) reported that the stability of nanofluids formulated without stabilizers would change rapidly with time. In their experiment for 2.5wt% Al_2O_3 /water nanofluid, Al_2O_3 nanoparticles became ‘completely separated’ after five hours (Fig. 2.1). Witharana et al. (2012) reported an experiment of 0.5wt% Al_2O_3 /water nanofluid stability without dispersing treatment, in which Al_2O_3 particles sedimentation layer was observed after only 30 minutes of preparation (Fig. 2.2). Drzazga et al. (2012) also reported that sedimentation layer can be observed in 1% CuO /water and 0.5% Al_2O_3 /water nanofluids after one day of preparation if no appropriate dispersing treatments were applied.

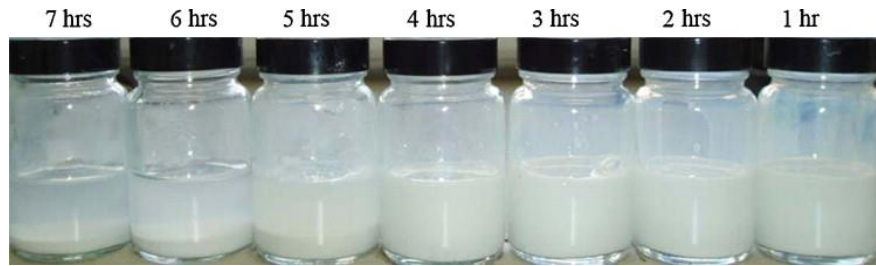


Fig. 2.1 Stability of Al_2O_3 /water nanofluid (without any stabilizers) changing with time (Wen et al., 2009)



Fig. 2.2 A gravity settling experiment of 0.5 wt% Al_2O_3 /water (Witharana et al., 2012)

In order to ensure nanofluids are stable for an even longer period, how to diminish nanoparticles agglomeration and prevent nanoparticles sedimentation are the two prime challenges. To deal with such problems, some typical operations can be found in previous publications (Ghadimi et al., 2011). Based on what, it can be summarised that adding surfactant and physical agitation are the two most commonly applied options to make nanoparticles dispersed well in basefluid.

2.1.2.2 Adding surfactant to keep nanofluid stable

Adding surfactant (or dispersant) to basefluid was reported to be effective to avoid nanoparticles sedimentation for a longer period (Mukherjee and Paria, 2013). The reason is that the hydrophobic surfaces of nanoparticles can be modified by surfactant to become more hydrophilic and vice versa for non-aqueous liquids. Then a repulsion force between suspended particles is triggered by zeta potential absolute value (Ghadimi et al., 2011). This repulsion force prevents nanoparticles from forming bigger agglomerations (which will deposit to vessel bottom quickly).

Previously, those stabilizers which have been used very often are sodium dodecylsulfate (SDS) (Hwang et al., 2007), dodecylbenzenesulfonate (SDBS) (Wang and Zhu, 2009), salt and oleic acid (Hwang et al., 2008, Yu et al., 2010). When additional substances are necessary to keep nanofluids

stable, choosing the right stabilizers is the key point. Moreover, enough stabilizer should be applied as inadequate surfactant cannot make a sufficient coating to ‘persuade’ electrostatic repulsion and compensate van der Waals attractions (Jiang et al., 2003).

Although nanofluids with high stability can be obtained by adding stabilizers, however, it also has some ‘side-effects’ on heat transfer characteristic and rheology of nanofluids (Das et al., 2003a, Wen and Ding, 2005a, Wang and Mujumdar, 2007), especially when nanofluid temperature is high in some experiments of pool boiling (Wen et al., 2011). More specifically, when nanofluid temperature is high, the bonding between surfactant and nanoparticles could be damaged. The temperature threshold was reported as $T = 60^{\circ}\text{C}$ (Assael et al., 2005, Murshed et al., 2008a). Thus when nanofluid temperature is greater than $T = 60^{\circ}\text{C}$, additional stabilizers would lose positive effects considerably and sedimentation of nanoparticles will occur (Wang and Mujumdar, 2007). Because nanofluid is usually applied as coolant for high temperature devices, it is actually difficult to obtain super stable nanofluids by adding stabilizers in practical applications.

2.1.2.3 Performing physical agitations to disperse nanoparticles

In order to obtain ‘pure’ nanofluid without the issues caused by high temperature in practical applications, physical agitation is usually preferred rather than adding stabilizers (Witharana et al., 2013). Physical agitation uses mechanical forces such as stirring and ultrasonic treatments to break down particles agglomeration which is caused by Van Der Waals force and make nanoparticles suspend in basefluid for a longer period. Because physical operation is easy to operate and does not induce any other impurities to nanofluid, it becomes increasingly prevail in recent years. Tab. 2.1 shows some typical nanofluid preparations reported in the past twelve years. From which it can be found that physical vibration is the most popular operation to keep nanofluids stable.

Tab. 2.1 Summary of some nanofluids prepared with physical agitation

References	nanofluids	Operations	Average particle size	Volume fraction	Stability
(Putra et al., 2003)	$\text{Al}_2\text{O}_3/\text{water}$	Ultrasonic vibration	131.2nm	1~4%	Sedimentation in the 4% sample by six hours
(Putra et al., 2003)	CuO/water	Ultrasonic vibration	87.3nm		
(Wen and Ding, 2005b)	$\text{TiO}_2/\text{distilled water}$	Ultrasonication and high-shear homogenizer	34nm	0.012%	Agglomeration occurs after a few hours

References	nanofluids	Operations	Average particle size	Volume fraction	Stability
(Kim et al., 2007a)	Al ₂ O ₃ /distilled water	Ultrasonic excitation	47nm	10 ⁻⁵ %~10 ⁻¹ %	Irregular clusters just after preparations
(Kim et al., 2007a)	TiO ₂ /distilled water	Ultrasonic excitation	23nm		
(Wen et al., 2011)	Al ₂ O ₃ /deionized water	Ultrasonication	20~150nm	0.001%, 0.01%, 0.1%	Stable for a couple of hours
(Kouloulis et al., 2016)	Al ₂ O ₃ /water	Ultrasonication and vortex mixer	50nm	0.06%	Visible sedimentation layer by 3.5 days

Putra et al. (2003) used a four-hour ultrasonic vibration to obtain stable Al₂O₃/water and CuO/water nanofluids (volume fractions $\phi = 1\%$, 2% and 4%). After their operation, no sedimentation was observed in the following six hours in 1% and 2% samples but minor sedimentations can be observed in 4% sample. Thereafter, minor sedimentation was observed in 4% suspension but none in 1% and 2% suspensions. Building on that, in order to obtain the best suspensions, they used freshly vibrated nanofluid in each experiment and the experimental time of 1.5 to 2 hours did not bring out any sedimentation.

Wen and Ding (2005b) used ultrasonication and high-shear homogenizer to break the agglomerate for stable nanofluids after adding TiO₂ powders to distilled water. In their work, ultrasonication of the vessels which contain TiO₂/water nanofluids lasted for about 15 minutes, and the homogenizer kept working for 30~180 minutes with a speed up to 24000rpm and shear rate up to 40000s⁻¹. Furthermore, the pH value in suspension was kept at a low level (pH = 3) to avoid the iso-electrical point (pH = 6.5) of TiO₂ particles. Besides, they used the suspensions from the upper part of vessel for following works to eliminate the observed small sediments. By above methods, their dispersion was found to be very stable in a couple of weeks without visually observable sedimentation. However, it was also mentioned that ‘a very small amount of sediments was still found at the bottom of the container after a few hours’.

Kim et al. (2007a) dispersed dry Al₂O₃ and TiO₂ nanoparticles into distilled water with a three-hour ultrasonic excitation before their experiments. Although it was satisfying that TEM images showed the mean sizes of Al₂O₃ and TiO₂ nanoparticles were 47nm and 23nm respectively, however, electrostatic repulsion between nanoparticles was doubted not strong enough because aggregations and sedimentations actually occurred in both Al₂O₃/water and TiO₂/water nanofluids just after preparations.

In order to obtain stable Al₂O₃/water nanofluids with different concentrations, Wen et al.

(2011) dispersed dry Al_2O_3 nanopowder into de-ionized water with a two-hour ultrasonic agitation. The nanofluids were found not permanently stable but could last for a couple of hours without visible sedimentations. Moreover, in order to make the nanofluids for their following experiments are even more stable, they carried out the agitation just before the experiments.

Kouloulis et al. (2016) performed ultrasonication bath and vortex mixer for five hours when they prepared 0.06% Al_2O_3 /water nanofluid for natural convection experiments. According to their visualization study, although no visible change in the nanofluid stability took place after 24 hours of preparation, however, the suspension became less bright by the middle of the first week, indicating significant nanoparticle deposition. During the period between the first half week and up to the second week, the nanofluid appeared qualitatively unchanged in terms of stability, while in the third week concentration stratification layers were visible.

Despite sonication treatment makes nanoparticles dispersed well with less sedimentations, but it does not mean longer sonication operation will provide better suspensions definitely. To determine the optimum duration of sonication operation, Kwak and Kim (2005) varied the durations from one hour to thirty hours and measured the average size of CuO particles. By comparison, they concluded that the time period of nine hours is the optimum to obtain the most stable nanofluids. In the sample with nine hours sonication, the average particle size remained same (60nm) regardless of particle volume fractions after a hundred days.

From above works, it can be found that although physical agitation is popular to keep nanofluids stable, however, the effect usually does not last a long period. The reason is supposed to be nanoparticles are powerless to resist Van Der Waals attractive force (which induces agglomeration) and gravity force (which induces sedimentation). Although it was reported that nanofluid could be stable for up to a hundred days with an appropriate physical treatment in (Kwak and Kim, 2005), but it is difficult to find similar reports from other researchers. In other words, during nanofluid applications, no matter in engineering or research, nanoparticles sedimentation is always an issue which cannot be ignored. This is one of the original motivations of this project.

2.2 Nanofluid thermal conductivity enhancement

2.2.1 Mechanisms

Thermal conductivity k is a measurement of material's ability to conduct heat. It presents the quantity of heat can pass in unit time through a plate of particular area and thickness when its opposite faces differ in temperature by one Kelvin (Incropera, 2011). In recent years, there is no doubt that adding nanoparticles can enhance the thermal conductivity of basefluid dramatically. Although no one has clarified firmly what are the reasons for this phenomenon, however, three possible reasons are widely accepted around the world for the anomalous enhancement of nanofluid thermal conductivity, they are (Kebllinski et al., 2002, Machrafi and Lebon, 2016, Pang et al., 2014):

- Brownian motion of nanoparticles
- Liquid layering at particle-liquid interface
- Nanoparticles clustering effects

2.2.1.1 Brownian motion of nanoparticles

Brownian motion of nanoparticles at molecular and nanoscale level is considered as the primary mechanism for the enhancement of nanofluids thermal conductivity (Jang and Choi, 2004). Brownian motion has two contributions to the thermal conductivity enhancement, one is direct effect due to the motion of nanoparticles that transports the heat, while the other one is indirect effect due to the convection of liquid molecules caused by moving nanoparticles. The most important evidence for this hypothesis is that nanofluids thermal conductivity increases with increasing temperature (Mintsa et al., 2009).

According to Stokes-Einstein equation $\bar{x}^2 = \frac{\kappa_B T}{3\pi r \mu} t$ (which predicts small particle's Brownian motion range $\bar{x}$ in fluid), Brownian motion does not depend on Boltzmann constant κ_B only, but also influenced by some other factors such as consideration time t , temperature T , particle size r and viscosity μ of environmental fluid (Edward, 1970). It means that even to the same nanofluid, nanoparticles Brownian motion may have different contributions to nanofluid thermal conductivity enhancement due to different temperatures. This hypothesis was validated

by experimental investigations in (Das et al., 2003b, Kim et al., 2007b) by showing that the nanofluid thermal conductivity enhancements increased noticeably with increasing temperatures.

2.2.1.2 Liquid layering at particle-liquid interface

On the interface between solid and liquid, liquid molecules are known to form more ordered layer structures than those in bulk liquid. Xue (2003) believed these ordered solid-like structures play an important role to enhance nanofluid thermal conductivity. Furthermore, he also inferred such an organized solid-like structure on solid particle surface might be a governing factor in heat conduction from solid to adjacent liquid.

Yu and Choi (2003) investigated the connection between nanolayer thickness and nanofluid thermal properties. Based on their study, they proposed a new model for nanofluid thermal conductivity prediction. However, before they modified Maxwell equation for the new prediction model, the expression of effective volume fraction was changed first as $\phi_e = \frac{4}{3}\pi(r + h_{nl})^3 n = \frac{4}{3}\pi n r^3 (1 + h_{nl}/r)^3 = \phi(1 + \beta)^3$ to consider layer thickness effect, in which n is particle number per volume unit and $\beta = h_{nl}/r$ is the ratio of nanolayer thickness to original particle radius. In above expression, it can be found that nanofluid effective volume fraction ϕ_e increases with β . According to classical Maxwell equation, greater ϕ will enlarge nanofluid thermal conductivity. Based on the same idea, Yu and Choi (2004) also developed Hamilton-Crosser model to predict nanofluid thermal conductivity. However, these models still need to be validated by more experimental measurements.

2.2.1.3 Nanoparticles clustering effects

Well dispersed nanoparticles are considered to be better for nanofluid thermal conductivity enhancement, but nanoparticle clusters are also considered to be a possible mechanism of thermal conductivity enhancement in some literatures (Buongiorno et al., 2009, Eastman et al., 2004). The theoretical basis is that nanoparticle clustering is possible to enhance effective volume fraction ϕ_e of nanofluid because of liquid filling into the space between the clustered nanoparticles.

The schematic tendency between excess thermal conductivity k and nanoparticles cluster packing fraction ϕ is shown in Fig. 2.3, in which number i, ii, iii and iv indicate that clustering

are closely packed, simple cubic arrangement, loosely packed and separated by layers thin enough, respectively. According to the analysis reported by Koblinski et al. (2002), it is believed that the excess thermal conductivity k increases with decreasing packing fraction.

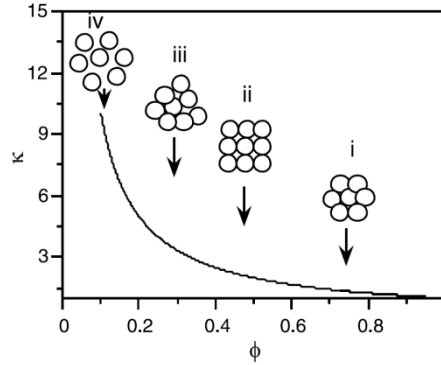


Fig. 2.3 Relationship between thermal conductivity k and nanoparticles cluster packing fraction ϕ (Koblinski et al., 2002)

2.2.2 Measurements of nanofluid thermal conductivity

To measure nanofluid thermal conductivity, transient hot-wire method and temperature oscillation technique are the two most important methods. Based on transient hot-wire theory for liquid thermal conductivity measurement (Healy et al., 1976), Nagasaka and Nagashima (1981) developed a device to measure thermal conductivity of electrically conducting liquids. Due to the short operation time of transient hot-wire method ($t < 5s$), the influence of natural convection caused by temperature gradient is possible to be avoided. This is the reason why hot-wire method is applied so widely today in nanofluid thermal conductivity measurement.

Besides transient hot-wire method, temperature oscillation technique proposed by Roetzel et al. (1990) is another popular method. Because temperature oscillation technique is purely thermal and the electrical components of the apparatus are away from test sample, the measuring operation is believed to have no influence to the ion movements at all. For this reason, the accuracy of measurement can be guaranteed (Wang and Mujumdar, 2007).

By means of above two methods, researchers have measured thermal conductivities of many different nanofluids in recent years (Lee et al., 1999, Murshed et al., 2005, Hwang et al., 2006, Xie et al., 2002, Kang et al., 2006, Liu et al., 2006). Buongiorno et al. (2009) had a comprehensive summarise for previous benchmark studies on the thermal conductivity of nanofluids, from which it can be found that Al_2O_3 , CuO , TiO_2 and SiO_2 are used very often by researchers in the past

two decades, and there is no doubt that adding nanoparticles can enhance basefluid thermal conductivity.

By previous publications, nanofluid thermal conductivity can be found to increase with increasing volume fraction, increasing temperature and decreasing particle size. It is actually a good support to those possible nanofluid thermal conductivity enhancement mechanisms summarised in Section 2.2.1. However, it also can be found that, due to the noticeably different experimental data, it is not easy to give a comprehensive regression model for nanofluids thermal conductivity predictions. For this reason, the numerical simulations in this PhD project will use nanofluid experimental data instead of applying previous prediction models. This will be helpful to obtain even more reliable simulation results.

2.3 Nanofluid heat transfer performance

2.3.1 Nanofluid natural convection

Natural convection is one of heat transport mechanisms, in which the fluid motion is generated by density differences due to temperature gradients (Kakac et al., 1985). In natural convection, the fluid surrounding a heat source receives heat and becomes less dense and rises, subsequently the surrounding cooler fluid moves to replace it. This cooler fluid is then heated and the process continues to form convection current. Recently, using nanofluid instead of pure liquid for natural convection is becoming one of the most interesting topics (Nsofor, 2008). However, deterioration of heat transfer performance in nanofluid natural convection was usually reported in previous experimental studies (Haddad et al., 2012).

Putra et al. (2003) found notably heat transfer deterioration in horizontal cylinders (Fig. 2.4) filled with Al_2O_3 /water and CuO /water nanofluids ($\phi=1\%$ and 4%). This phenomenon could be found in all their three testing devices with different ratios of cylinder length and diameter ($L/D = 0.5, 1.0, 1.5$). Based on such observation, they concluded that adding nanoparticles to pure fluid could lead to worse natural heat convection. They ascribed the possible reasons to particle-fluid slip and nanoparticles sedimentation. However, they also admitted that the reason of this deterioration is still not clear yet.

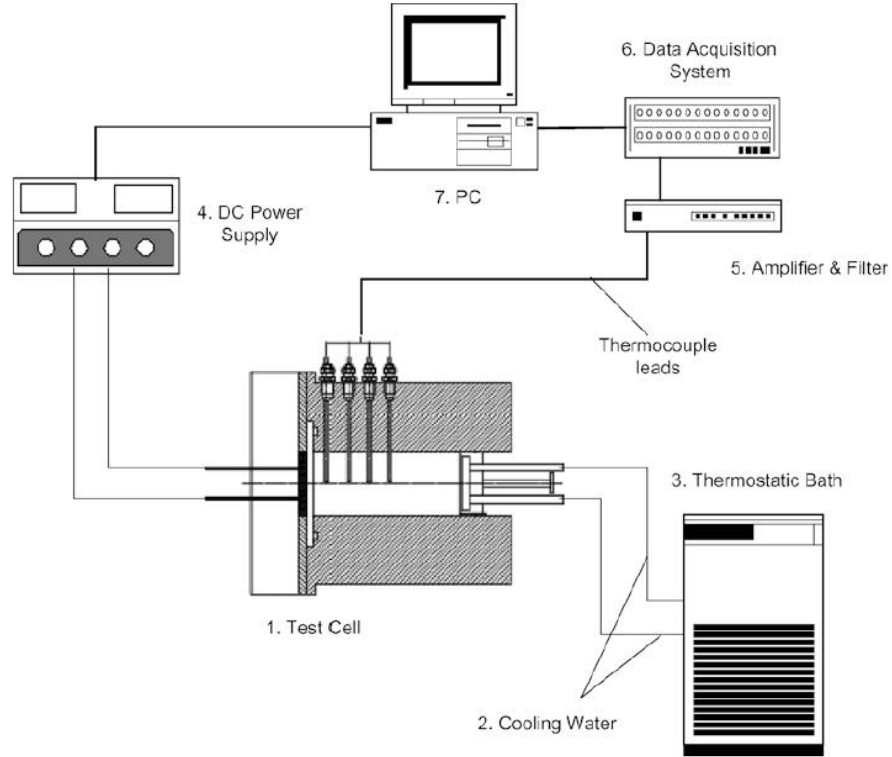


Fig. 2.4 Schematic experimental apparatus in (Putra et al., 2003)

Similar observation was also reported by Wen and Ding (2005b). In their experiments, 0.19%~0.57% TiO_2 /water nanofluids between heating and cooling discs were found to have lower natural convective heat transfer coefficient than pure water. Furthermore, they also reported that this deterioration increased with volume fraction. They supposed that the convection induced by particles concentration difference could be a possible reason. Although Ho et al. (2010) reported up to 18% natural convective heat transfer enhancement of 0.1% Al_2O_3 /water nanofluid in a similar device and a correlation $\overline{Nu}_{nf} = C Ra_{nf}^n (Pr_{nf,h}/Pr_{nf})^m (\beta_{nf,h}/\beta_{nf})^p$ was used to present such effect. However, degradation was still found when volume fraction was greater than 2%.

Li and Peterson (2009) reported natural heat convection deterioration of 0.5%~6% Al_2O_3 /DI water nanofluids in a cavity enclosed by two copper bars and a rubber O-ring. They inferred the reason could be the nanoparticles' Brownian motion smoothen the temperature gradient and lead to the delay of the natural convection. Moreover, higher viscosity of nanofluids also could induce such an effect. Ni et al. (2011) reported deteriorated natural convection after their experiments for 1.08% Al_2O_3 /water nanofluid in a Rayleigh-Bénard configuration. They used a correlation $Nu =$

$0.115Ra^{0.306}$ to fit their experimental data of pure water and a part of experimental data of Al_2O_3 /water nanofluid and suggested that the significant decrease might be caused by mass diffusion of nanoparticles.

Nnanna (2007) reported that the presence of Al_2O_3 nanoparticles did not impede the water free heat convection when volume fraction was in the range of 0.2%~2%. However, the heat convection declined due to increased kinematic viscosity since volume fraction was larger than 2%. Based on their analysis, they proposed a correlation $Nu = 16.4e^{-4 \times 10^{-7} Ra \phi} e^{-11 \phi}$ for such effect when $10^5 \leq \phi Ra e^{-11 \phi} \leq 10^6$. Hu et al. (2014) carried out an experimental investigation of TiO_2 /water nanofluids natural convection in a square enclosure (Fig. 2.5). Their results indicated that natural convection heat transfer of TiO_2 /water nanofluids is more sensitive to viscosity rather than thermal conductivity. In other word, adding nanoparticles will induce worse natural heat transfer performance of basefluid because viscosity of basefluid is increased considerably. Compared to heat transfer enhancement due to increased thermal conductivity, increased viscosity is playing the dominant role to deteriorate natural heat transfer performance.

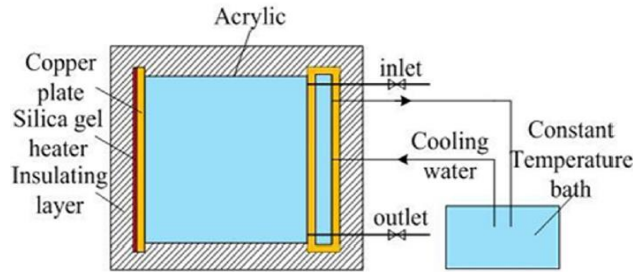


Fig. 2.5 Schematic experimental apparatus reported in (Hu et al., 2014)

Based on above experimental investigations, it can be concluded that natural convection heat transfer performance of fluid will be worse after adding nanoparticles. For this phenomenon, the possible slip between nanoparticles and basefluid is supposed to be the primary reason. Apart from that, the possible nanoparticles sedimentation is supposed to be another possible reason. Moreover, nanofluid heat transfer due to natural convection is considered more sensitive to viscosity rather than thermal conductivity. Therefore, in CFD simulations for nanofluid heat transfer, it is critical to substitute practical and reliable nanofluid properties instead of those values obtained from prediction models.

2.3.2 Nanofluid forced convection

Forced convection is a transport mechanism in which fluid motion is generated by external source, such as a pump, fan and suction device, etc (Incropera, 2011). Generally, fluid flow can be classified as laminar and turbulent by Reynolds threshold $Re \approx 2300$ (White and Corfield, 2006). Thus the forced nanofluid convection heat transfer is always investigated in laminar flow and turbulent flow respectively. Due to limitations of experimental setup, most investigations of nanofluid forced convection heat transfer are carried out in a horizontal tube system (Fig. 2.6) as reported in (Wen and Ding, 2004).

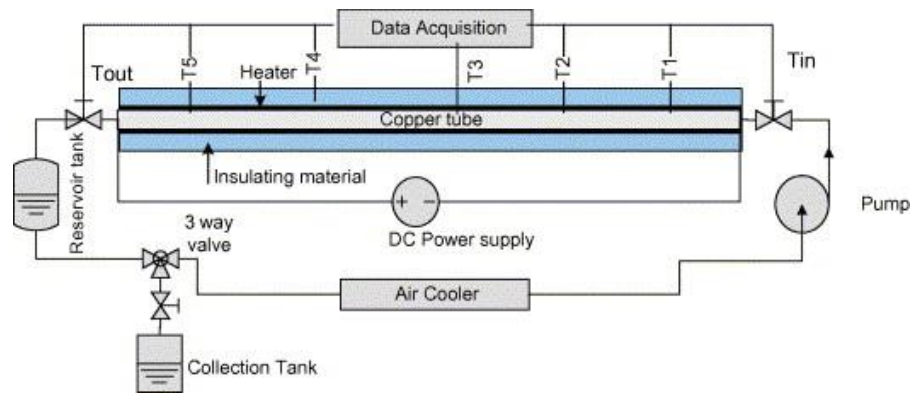


Fig. 2.6 Experimental setup reported in (Wen and Ding, 2004)

Yang et al. (2005) studied the convection heat transfer performance of 2wt% graphite/oil nanofluids in laminar flow through a horizontal tube heat exchanger. Their experimental results showed that nanoparticles increased heat transfer coefficient of the fluid system in laminar flow with increasing volume fraction and Reynolds number. Such relationship was summarised as $Nu = 0.4328(1 + 11.285\phi^{0.754}Pe^{0.218})Re^{0.333}Pr^{0.4}$. They also concluded that the type of nanoparticles, particle loading, basefluid chemistry, and process temperature are the most important factors to achieve higher nanofluids heat transfer coefficients.

Murshed et al. (2008b) also reported an experimental investigation into force convective heat transfer of TiO₂/water nanofluids (ϕ is up to 0.8%) flowing through a cylindrical mini-channel under laminar flow and constant wall heat flux conditions. Their results showed that nanofluids exhibit a considerably enhanced convective heat transfer coefficient than basefluid. Moreover, the heat transfer coefficient also increased with volumetric loading of nanoparticles. Such effect can

be roughly measured by the well know Shah's correlation $Nu = 1.953 \left(Re Pr \frac{D}{x} \right)^{1/3}$. For the enhanced heat transfer coefficients, they supposed the reasons could be the enhanced effective thermal conductivity of nanofluid and the acceleration of energy exchange process in basefluid due to nanoparticles random movement. Besides, the migration of nanoparticles in basefluid due to shear action, viscosity gradient and Brownian motion on the tube cross section was also considered as a possible reason.

He et al. (2007) measured the forced convective heat transfer coefficient h of TiO_2 /water ($\phi = 0.2\%$, 0.6% and 1.1%) in turbulent condition ($Re=5900$). Their results showed that the convective heat transfer coefficient of TiO_2 increased with increasing nanoparticle concentration in turbulent flow conditions. They used Gnielinski equation (Gnielinski, 1976) as the comparison basis for their experimental data. Compared to the experimental investigation reported in (Murshed et al., 2008b), it can be found that the effect of particle concentration on nanofluid heat transfer coefficient h is more considerable in turbulent flow than laminar flow.

Torii (2009) also used a similar experimental apparatus to investigate convective heat transfer performance of diamond/water, Al_2O_3 /water and CuO /water nanofluids ($\phi = 0.1\%$, 1% and 5%) in turbulent flow ($Re \approx 6000$). Taking the results for Al_2O_3 /water as example, their experiments showed that the Nusselt number increased with increasing volume fraction and Reynolds number. They also used Gnielinski equation as the comparison basis for their experimental data. In their study, enhancement of nanofluid heat transfer was considered affected by several factors such as concentration, aggregation and Zeta potential.

By above experimental studies on nanofluids forced convection in both laminar and turbulent conditions, nanofluid heat transfer performance is found to increase with increasing volume fraction and Reynolds number. Furthermore, by comparisons, it also can be found that nanofluid heat transfer enhancement is more considerable in turbulent flow than that in laminar flow. Apart from above general conclusions, it is also concluded that the nanoparticles migration in basefluids due to shear action, viscosity gradient and Brownian motion could be an important reason to affect nanofluid heat transfer performance.

2.4 CFD simulations for nanofluid flow and heat transfer

2.4.1 Different CFD approaches

To gain a better understanding of nanofluid flow and heat transfer performance, many numerical investigations have been carried out in the past two decades (Kamylar et al., 2012). Although molecule dynamics approach (Nie et al., 2004, Sarkar and Selvam, 2007, Lu and Fan, 2008, Mohebbi, 2012, Cui et al., 2015) and Lattice-Boltzmann approach (Xuan and Yao, 2005, Nemati et al., 2010, Ashorynejad et al., 2013, Lai and Yang, 2011) have been applied and satisfactory results were reported very often, however, ‘classical’ CFD methods are still playing a dominant role in this area.

In conventional CFD simulations, nanofluid is usually assumed to be a stable and homogeneous single-phase fluid but with different uniform properties (i.e. density, thermal conductivity and viscosity, etc) as its basefluid. After predicting such properties, nanofluid flow and heat transfer performance can be investigated numerically in the same way as that for pure fluid. This method is recognised as ‘single-phase’ approach (Kamylar et al., 2012).

Since Ding and Wen (2005) reported that the particles migration in nanofluid suspension could be one of the most important reasons to influence nanofluid heat transfer performance, researchers began to apply multi-phase approaches for nanofluid flow and heat transfer simulation. These investigations can be sorted into four categories (approaches) in terms of the different basic ideas. They are Eulerian-Lagrangian approach, Eulerian-Eulerian approach, Mixture approach and Volume of Fluid (VOF) approach.

More generally, all above approaches can be summarised as single-phase method and multi-phase method. Because multi-phase method can be performed in completely different ways, it can be sorted further into two categories in terms of basic theoretical frames: Lagrangian-based approach and Eulerian-based approach. Therefore, in this section, previous typical CFD works on nanofluid flow and heat transfer are reviewed from three aspects, i.e. single-phase approach, Eulerian-Lagrangian approach and Eulerian approaches.

2.4.2 Single-phase approach

As mentioned before, single-phase approach assumes that nanofluid is a stable and homogeneous single-phase fluid but with different properties as its basefluid. For nanofluid properties such as thermal conductivity and viscosity, prediction models are always employed before the simulations according to several factors such as volume fraction, temperature and nanoparticle size (Kaltah et al., 2012). With above assumption, nanofluid flow and heat transfer performance can be investigated numerically in the same way as that for pure fluid. Based on this idea, CFD simulation can be applied easily for both nanofluid natural and forced convection problems.

For nanofluid natural convection, Khanafer et al. (2003) found water heat transfer rate in a two-dimensional enclosure could be substantially increased by adding more copper nanoparticles (ϕ is up to 20%). Oztop and Abu-Nada (2008) found natural heat transfer enhancement by using Al_2O_3 /water, TiO_2 /water and Cu/water nanofluids ($\phi=0\sim 20\%$) in two-dimensional rectangular enclosures with different aspect ratios. In their study, the enhancement was more pronounced at low aspect ratio than that at high aspect ratio. Aminossadati and Ghasemi (2009) reported that adding Cu, Ag, Al_2O_3 and TiO_2 nanoparticles (ϕ is up to 20%) could improve cooling performance of pure water in a bottom-heated two-dimensional enclosure, especially when Rayleigh number was low. Oueslati and Bennacer (2011) found nanofluid natural heat convection enhancement in a two-dimensional cavity when the volume fraction of Al_2O_3 , TiO_2 and Cu nanoparticles was lower than 5%. Ternik and Rudolf (2012) indicated that average Nusselt number was an increasing function of nanofluid volume fraction after they examined the heat transfer enhancement of water-based Au, Al_2O_3 , Cu and TiO_2 nanofluids (ϕ is up to 10%) in a two-dimensional cavity.

In above investigations, Nusselt number along heating wall is preferred to measure nanofluid heat transfer performance. Generally, by above single-phase CFD simulations, it can be concluded that adding nanoparticles to pure liquids can enhance natural convection heat transfer performance of basefluid. Furthermore, this enhancement increases with increasing Rayleigh number and volume fraction of nanoparticle phase. Compared to previous experimental investigations in which deteriorations of nanofluid natural convection heat transfer are usually

reported, it is not difficult to find that arguments still exist in this area. For the controversial conclusions, the possible slip between nanoparticles and basefluid is supposed to be one of the primary reasons.

For nanofluid forced convection, Raisee et al. (2006) investigated the hydrodynamic and thermal characteristics of laminar Al_2O_3 /water and Al_2O_3 /EG nanofluids ($\phi=1\sim10\%$) flowing through two-dimensional and axis-symmetric passages ($Re = 100$ and 250). It was found that the addition of nanoparticles increased both heat transfer coefficient and shear stress on the wall. Comparatively, they considered that the nanoparticles have more effects on the wall shear stress rather than heat transfer coefficient. Abu-Nada (2008) obtained the distributions of Nusselt number of five different nanofluids (Cu, Ag, Al_2O_3 , CuO, and TiO_2 /water with ϕ up to 0.2%) at the top and bottom walls in a channel with back-facing step. It was found that nanoparticles with high thermal conductivity (such as Ag and Cu) had more enhancements on Nusselt number outside the recirculation zones ($Re = 200$ and 400). Bajestan et al. (2010) investigated heat transfer of laminar Al_2O_3 /water nanofluid ($\phi = 0.2\sim6\%$) flow in a bent pipe and a straight one numerically ($Re = 833$). Their simulations showed that both the nanoparticles and curvature effects enhanced heat transfer performance but also increased pressure drop. Mehrez et al. (2013) investigated mixed convection of nanofluids (Cu, Al_2O_3 , CuO and TiO_2 /water with ϕ up to 10%) flow in an open cavity heated from below by uniform temperature ($Re = 100\sim500$). It was found that the heat transfer increased with the increase of Reynolds number and volume fraction of nanoparticles. The selection of these parameters was reported important to obtain maximum enhancement of heat transfer.

In above investigations of nanofluid forced convection, it can be found that local heat transfer coefficient and Nusselt number along heating surface are always applied to measure nanofluid forced convection performance. Generally, by above single-phase CFD simulations, it can be concluded that nanofluid forced convection heat transfer increases with nanoparticles volume fraction and Reynolds number. This conforms to previous experimental investigations. However, after adding nanoparticles to basefluid, the increased viscosity also plays a role to impede nanofluid flow. In some of previous investigations, the relative motion between nanoparticles and basefluid was considered as a primary reason for the noticeable difference between numerical and

experimental results, but, due to the nature of single-phase CFD simulation, it is actually impossible to consider the interactions between nanoparticles and basefluid.

2.4.3 Eulerian-Lagrangian approach

If nanofluid was treated as a real two-phase flow, it will be considered to have two parts: basefluid and nanoparticles. The most straightforward approach is to solve governing equations for basefluid first, then the information of fluid flow is used to predict nanoparticles motion based on the second Newton's law (4.1). In momentum and energy equations for basefluid, extra terms S_m and S_e are added to governing equations (3.15) and (3.21) for the impacts between fluid and solid phases in terms of momentum and energy, respectively. However, regarding the forces acting on nanoparticles, different presentations can be found from previous publications.

For nanofluids under Eulerian-Lagrangian frame, the work reported by Bianco et al. (2009) is the most typical one. In which they reported a comparative work between single-phase method and multi-phase method. They carried out the investigation by FLUENT and Eulerian-Lagrangian approach was applied to track nanoparticles motion (Stokes number $Stk \approx 0.01$). In their works, only drag force was considered on nanoparticles. After the simulations for 1% and 4% Al_2O_3 /water nanofluids flowing through a horizontal tube, they concluded that the Lagrangian approach predicted very similar results as the single-phase method did. However, the information of how many particles were employed to present nanoparticles in their investigation was not presented. Actually, it is impossible to track all the real nanoparticles in Lagrangian approach due to the huge particle amount (i.e. about 5.7×10^{20} nanoparticles with diameter of $d = 100nm$ in a $1cm^3$ volume).

Pallares and Grau (2010) presented a CFD simulation of 3% Al_2O_3 /water nanofluid natural convection in a two-dimensional square cavity by Eulerian-Lagrangian approach ($Stk \approx 10^{-6}$ and 10^{-4}). They assumed that nanoparticles distribution is perfectly uniform in basefluid and only drag force from basefluid was considered. In their simulation, 40000 and 400000 numerical particles were applied to present real nanoparticles in different cases respectively. Compared to pure water, a heat transfer rate reduction of about 1% was reported in their conclusion.

Tahir and Mital (2012) reported a CFD study of 1% and 4% Al_2O_3 /water nanofluids flow and heat transfer in a circular channel by Eulerian-Lagrangian approach. With the help of FLUENT, they treated basefluid as a continuous medium and the laminar flow field was solved by N-S equations. Being different with other previous works, Brownian force, thermophoretic force and Saffman lift force were considered together for the first time in CFD investigations of nanofluid. The nanoparticles were tracked individually under Lagrangian frame and their trajectories were determined using particle force balance. Using this approach, a good match was obtained between their numerical results and the experimental results reported by other literatures. However, they also did not mention how many Lagrangian particles were applied in their simulations.

Moraveji and Esmaeili (2012) published a comparison between single-phase approach and Lagrangian approach for forced convection simulation of 1wt% and 4wt% Al_2O_3 /water nanofluids in a long tube with uniform heating at wall boundary ($Stk \approx 0.01$). In this work, only drag force was considered for nanoparticles. It was reported that heat transfer was enhanced by increasing the concentration of nanoparticles in nanofluid and Reynolds number. However, their simulations showed that the results predicted by the two approaches 'are quite similar'. This conclusion is similar to that in (Bianco et al., 2009).

In summary, it can be found that when nanofluid is simulated under Eulerian-Lagrangian frame, it is a truly practical way because nanoparticles can be treated as real particles suspended in basefluid. However, in most previous CFD works under Eulerian-Lagrangian frame, drag force is the only force type to count in for nanoparticles. It is reasonable to argue that this may ignore many details when particle motion features are considered. Moreover, in a typical Eulerian-Lagrangian approach, the focus is mainly on the interactions between continuous flow and discrete particles, but the interactions among discrete particles are not considered, this may lead to information missing for nanofluid investigations.

2.4.4 Eulerian approaches

Under Eulerian frame, Eulerian-Eulerian approach, VOF approach and Mixture approach have been applied for nanofluid CFD simulations. No matter which approach is selected, a common assumption always can be found: nanoparticles phase is a pseudo fluid. For each

numerical cell within computational region, basefluid and the ‘fluid’ of nanoparticles may exist together. With this consideration, all the three approaches under Eulerian frame have to meet a criteria expressed by $\phi_f + \phi_n = 1$, i.e. the sum of basefluid volume fraction ϕ_f and nanoparticle volume fraction ϕ_n is 100%.

With such a consideration, Eulerian-Eulerian approach solves two sets of governing equations, i.e. (3.15) and (3.21) for the flow and heat transfer of both basefluid and nanoparticle phases. In the two sets of equations for basefluid and nanoparticle phases respectively, additional terms S_m and S_e are still needed for momentum and energy exchanges between the two phases (Akbari et al., 2012). However by this approach, the most remarkable advantage is the required computational resource mainly depends on cells number, rather than particles number as that under Lagrangian frame.

VOF approach solves a single set of momentum equation and energy equation for the two phases and tracks their volume fractions all over the computational region (Moraveji and Ardehali, 2013). Nanofluid properties can be calculated according to the volume fractions of nanoparticle phase in every computational cell. In this approach, velocity U , pressure p and temperature T are shared by basefluid and nanoparticle phases. However, it should be noticed that VOF approach is based on a consideration that there is no interpenetration between phases. To each computational cell, it should have volume fraction $\phi_n = 0$ or $\phi_n = 1$. If $0 < \phi_n < 1$ is the case, the information of the interface between the two phases must be tracked (Hirt and Nichols, 1981).

Mixture approach can model multi-phase flow and heat transfer behaviours by solving one set of continuity, momentum, energy and volume fraction equations for the mixture (More details will be given in Chapter 5). In this procedure, several algebraic expressions for the relative velocities between different phases are needed. Thus in Mixture approach, it is possible to define some parameters what are particularly for nanoparticles sedimentation (based on experimental investigations). The difference between Mixture model and VOF model is that the interface between the two phases is not necessary to track. This nature of Mixture model saves lots of computational resource requirement when numerical mesh is fine.

(Behzadmehr et al., 2007) is believed to be the first publication using multi-phase method for

nanofluid CFD simulation. In their investigation, 1% Cu/water nanofluid in a circular tube was assumed to be a mixture of basefluid phase and nanoparticle phase, and Mixture approach (a simplified Eulerian-Eulerian method) was employed. In their study, the governing equations were solved for a mixture, whose properties were calculated based on the properties of both basefluid and nanoparticles. However, the difference between this method and the single-phase approach is, the slip velocity between basefluid and nanoparticle phase is considered and estimated according to mixture flow velocity and nanoparticle features (such as density and size). By their comparison of Nusselt number between the results provided by single-phase approach, Mixture approach and previous experimental investigation, it was found that Mixture approach gives better agreement to experimental investigation. Lotfi et al. (2010) also carried out a comparison between single-phase approach and Mixture approach for 1% Al_2O_3 /water nanofluid in horizontal tubes. They obtained a very similar conclusion that the Mixture model provided better results than single-phase approach.

Akbari et al. (2011) carried out a comparative investigation between single-phase and different multi-phase CFD simulations. They applied all the three approaches under Eulerian frame, i.e. VOF, Mixture and Eulerian-Eulerian models to laminar forced convection of 0.6%, 1% and 1.6% Al_2O_3 /water nanofluids in a horizontal tube heated by uniform heat flux. Furthermore, they also applied temperature-dependent basefluid density and viscosity to make the simulation is more practical. They reported that the three multi-phase models predicted better results than single-phase approach. However, if compared the results predicted by the three multi-phase models, no noticeable difference could be found.

Moraveji and Ardehali (2013) also reported a comparison between the three two-phase models and single-phase model for laminar forced convection of 0.5wt%, 1wt% and 6wt% Al_2O_3 /water nanofluids in a mini-channel heat sink. In this study the three multi-phase models showed very similar results. By their comparison, multi-phase approaches were believed to be better than single-phase approach, and Mixture model was suggested to be the best choice to achieve the balance between more accurate results and less CPU usage.

Garoosi et al. (2014) carried out a CFD study for natural and mixed convection heat transfer of Al_2O_3 /water nanofluids (up to $\phi = 5\%$) in a laterally-heated square cavity. In their

simulations, standard N–S equations were solved for nanofluid mixture, but an extra term was added in energy equation for the possible energy flux due to nanoparticles Brownian motion and thermophoretic effects. Their results indicated that there is an optimal volume fraction of nanoparticles at each Rayleigh and Richardson numbers for the maximum heat transfer rate. It was also observed that with low Rayleigh and high Richardson numbers, the particle distribution was ‘fairly non-uniform’.

By above investigations and correlative comparisons, treating nanofluid as multi-phase under Eulerian frame is believed to be more practical than Eulerian-Lagrangian approach in terms of computational effort. This is because when the dispersed particles are assumed to be a pseudo continuous phase, only another one set of governing equations is needed. In terms of further benefits in computational efficiency, Mixture approach is believed to be better than Eulerian-Eulerian approach and VOF approach.

2.5 Conclusions

A literature review is carried out in this chapter, in which some topics such as nanofluid preparation, thermal conductivity enhancement, natural and forced convections heat transfer and nanofluid CFD simulations. By this review, some conclusions can be drawn as follows:

1. For nanofluid preparation, two-step method is more popular than one-step method due to its good natures such as easy to operate and less apparatus are needed. However, nanoparticles sedimentation is a big problem if no appropriate dispersion treatments or stabilizers are applied after preparation.
2. According to previous experimental investigations, nanofluid has better thermal conductivity than the corresponding basefluid, but the mechanisms are not completely clear yet. When nanofluid properties (particularly thermal conductivity and viscosity) are needed for CFD simulation, it could be more practical to apply trustable experimental data instead of predicting models.
3. In experimental investigations, nanofluid heat transfer deterioration is usually observed in natural convection while enhancement can be observed in forced convection. For the unexpected phenomenon in nanofluid natural convection, the slip between nanoparticles

and basefluid is considered as one of the most possible reasons.

4. Single-phase CFD simulation is comparatively easy to carry out and very popular for present nanofluid numerical investigations. However with this approach, it is impossible to consider the possible slip between nanoparticles and basefluid.
5. Among multi-phase CFD approaches, Lagrangian method is considered as the most straightforward one, but simulations under Eulerian frame are believed to be more efficient with less required computational resource. Furthermore, Mixture approach under Eulerian frame is believed to be the best option to achieve even more benefits in computational effort.

3. CFD BASICS OF FLUID FLOW AND HEAT TRANSFER

3.1 Introduction

As mentioned in Chapter1 and Chapter2, a multi-phase CFD investigation is going to be carried out in this work to combine nanofluid flow, heat transfer and nanoparticles sedimentation via OpenFOAM. For this, some necessary CFD basics such as governing equations, main features of OpenFOAM and treatment of N-S equations in OpenFOAM are presented in this chapter. This chapter is the theoretical basis to build up OpenFOAM ‘platforms’ for the following nanofluid simulations in multi-phase way.

3.2 Governing equations for CFD simulation

In CFD simulations for any sorts of fluid, conservation rules are always the most important basis to describe fluid behaviour. For the simulations of single-phase fluid (such as pure water and air) or uniform mixture (such as nanofluid with single-phase assumption under Eulerian frame), conservation equations usually consist of a set of continuity equation, momentum equation and energy equation. While for the simulations under Lagrangian frame, the three governing equations are also employed to describe flow and heat transfer of the primary continuous phase.

3.2.1 Continuity equation

For continuity equation, the physical principle ‘mass is conserved’ is applied. According to this theory, a control volume of arbitrary shape of finite size, which is fixed in space needs to be considered. Fluid is supposed to move through the control volume and across the control surface as shown in Fig. 3.1 (Anderson Jr, 2009). This is the physical basis for all the following equations and derivations.

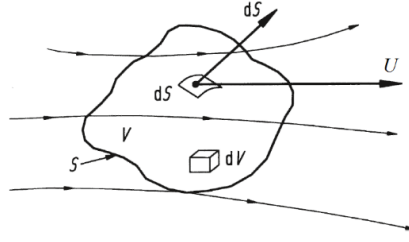


Fig. 3.1 Finite control volume in space (Anderson Jr, 2009)

At a point on the control surface in Fig. 3.1, flow velocity vector is U , vector volume elemental surface area is dS and dV is elemental volume inside the finite control volume V . To a hexahedron control volume with above considerations, if the velocities in direction x , y and z are presented as u , v and w , respectively, the total net mass flow out of the control volume through surface S can be expressed as:

$$\iint_S \rho U \cdot dS = \left[\frac{\partial(\rho u)}{\partial x} dx + \frac{\partial(\rho v)}{\partial y} dy + \frac{\partial(\rho w)}{\partial z} dz \right] dxdydz \quad (3.1)$$

The time rate of mass increase inside the control volume is:

$$\frac{\partial}{\partial t} \iiint_V \rho dV = \frac{\partial \rho}{\partial t} (dxdydz) \quad (3.2)$$

According to mass conservative law, the absolute values of the right hand side of (3.1) and (3.2) should be exactly same. Thus, continuity equation can be written in a conservation form (3.3) or a non-conservation form (3.4):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho U) = 0 \quad (3.3)$$

$$\frac{D\rho}{Dt} + \rho \nabla \cdot U = 0 \quad (3.4)$$

3.2.2 Momentum equations

Momentum equations for fluid in x , y and z directions can be derived by applying Newton's second law on an infinitesimal moving fluid element. To an infinitesimally small moving fluid element (Anderson and Wendt, 1995), if considered those components in x direction only, Newton's second law to the moving fluid element can be expressed as (3.5).

$$F_x = ma_x \quad (3.5)$$

where the acting force F_x on the fluid element consists of body forces (e.g. gravitational, electric and magnetic forces acting on element body) and surface forces (e.g. pressure, shear and normal

stresses acting on element surfaces).

If only gravity is considered for body force, then the total force F_x acting on the fluid element is:

$$F_x = \left(-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho g_x\right) dx dy dz \quad (3.6)$$

where τ is for shear stress at different faces in different directions.

Eventually, with the consideration that numerical cells volume do not change, (3.5) becomes to (3.7):

$$\rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho g_x \quad (3.7)$$

To a fluid element, (3.7) is usually referred to as momentum equation in x direction. In similar way, the momentum equations in y and z directions can be expressed as (3.8) and (3.9), respectively:

$$\rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho g_y \quad (3.8)$$

$$\rho \frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho g_z \quad (3.9)$$

Momentum equations (3.7)~(3.9) are obtained directly from an application of fundamental physical principles to an infinitesimal moving fluid element, they are non-conservation form N-S equations.

N-S equations also can be obtained in a conservation form:

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u U) = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho g_x \quad (3.10)$$

$$\frac{\partial(\rho v)}{\partial t} + \nabla \cdot (\rho v U) = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho g_y \quad (3.11)$$

$$\frac{\partial(\rho w)}{\partial t} + \nabla \cdot (\rho w U) = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho g_z \quad (3.12)$$

According to Stokes estimations for surface forces (Stokes, 1851), N-S equation (3.10) can be written as:

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u U) = -\frac{\partial p}{\partial x} - \frac{2}{3}\mu \frac{\partial}{\partial x} (\nabla \cdot U) + \mu \nabla^2 u + \mu \frac{\partial}{\partial x} (\nabla \cdot U) + \rho g_x \quad (3.13)$$

If fluid density does not change with time, (3.13) becomes to:

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u U) = -\frac{\partial p}{\partial x} + \mu \nabla^2 u + \rho g_x \quad (3.14)$$

With similar considerations for the variables in directions y and z , the momentum equation can be written in a general form:

$$\frac{\partial(\rho U)}{\partial t} + \nabla \cdot (\rho U U) = -\nabla p + \mu \nabla^2 U + \rho g \quad (3.15)$$

To incompressible fluid, momentum equation becomes to:

$$\frac{\partial U}{\partial t} + \nabla \cdot (U U) = -\frac{1}{\rho} \nabla p + \nu \nabla^2 U + g \quad (3.16)$$

where ν is kinetic viscosity.

In this work, the fluid density ρ may be replaced by a kinematic density ρ_k due to temperature variation (Boussinesq, 1903). Such density change is the reason for fluid natural convection, and (3.17) is the numerical basis to simulate this phenomenon according to Boussinesq assumption.

$$\rho_k = 1 - \beta(T - T_0) \quad (3.17)$$

where T_0 is reference temperature, and β is fluid thermal expansion coefficient.

3.2.3 Energy equation

For energy equation, another physical principle ‘energy is conserved’ is applied. This theory is also the first thermodynamics law (Cengel et al., 1998). When the basic idea of energy conservation is applied to flow model of a fluid element moving with the flow, the first thermodynamics law can be expressed in a comprehensive non-conservation form to include the considerations such as net flux of heat into the fluid element and work done on the element, etc:

$$\rho \frac{D}{Dt} (e + U^2/2) = \rho \dot{q} + \frac{\partial}{\partial x} (k \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y} (k \frac{\partial T}{\partial y}) + \frac{\partial}{\partial z} (k \frac{\partial T}{\partial z}) - \left[\frac{\partial(u p)}{\partial x} + \frac{\partial(v p)}{\partial y} + \frac{\partial(w p)}{\partial z} \right] + \frac{\partial(u \tau_{xx})}{\partial x} + \frac{\partial(u \tau_{yx})}{\partial y} + \frac{\partial(u \tau_{zx})}{\partial z} + \frac{\partial(v \tau_{xy})}{\partial x} + \frac{\partial(v \tau_{yy})}{\partial y} + \frac{\partial(v \tau_{zy})}{\partial z} + \frac{\partial(w \tau_{xz})}{\partial x} + \frac{\partial(w \tau_{yz})}{\partial y} + \frac{\partial(w \tau_{zz})}{\partial z} + \rho g \cdot U \quad (3.18)$$

where $\dot{q}$ is volumetric heat addition rate per unit mass, e is internal energy and T is temperature.

To cast (3.18) to a form dealing with internal energy e only, an equation (3.19) can be

obtained as:

$$\rho \frac{De}{Dt} = \rho \dot{q} + \frac{\partial}{\partial x} (k \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y} (k \frac{\partial T}{\partial y}) + \frac{\partial}{\partial z} (k \frac{\partial T}{\partial z}) - p (\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}) + \tau_{xx} \frac{\partial u}{\partial x} + \tau_{yx} \frac{\partial u}{\partial y} + \tau_{zx} \frac{\partial u}{\partial z} + \tau_{xy} \frac{\partial v}{\partial x} + \tau_{yy} \frac{\partial v}{\partial y} + \tau_{zy} \frac{\partial v}{\partial z} + \tau_{xz} \frac{\partial w}{\partial x} + \tau_{yz} \frac{\partial w}{\partial y} + \tau_{zz} \frac{\partial w}{\partial z} \quad (3.19)$$

If used completely flow field variables, the energy equation in conservation form can be written as:

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\rho e U) = \rho \dot{q} + \frac{\partial}{\partial x} (k \frac{\partial T}{\partial x}) + \frac{\partial}{\partial y} (k \frac{\partial T}{\partial y}) + \frac{\partial}{\partial z} (k \frac{\partial T}{\partial z}) - p (\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}) + \lambda (\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z})^2 + \mu \left[2(\frac{\partial u}{\partial x})^2 + 2(\frac{\partial v}{\partial y})^2 + 2(\frac{\partial w}{\partial z})^2 + (\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})^2 + (\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x})^2 + (\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y})^2 \right] \quad (3.20)$$

In most cases, the terms representing work done by pressure and viscous forces can be removed for incompressible fluid (Ferziger and Perić, 2002). Furthermore, in CFD simulations for fluid heat transfer, the contribution of volumetric heating of the element, i.e. the term $\rho \dot{q}$ in (3.20) is usually ignored. Thus, (3.20) becomes to:

$$\frac{\partial(\rho e)}{\partial t} + \nabla \cdot (\rho e U) = \nabla \cdot (k \nabla T) \quad (3.21)$$

To incompressible fluid, due to $e = c_p T$, the simplified conservation form of energy equation (3.21) can be written as:

$$\frac{\partial T}{\partial t} + \nabla \cdot (T U) = \nabla \cdot \left(\frac{k}{\rho c_p} \nabla T \right) \quad (3.22)$$

where c_p is specific heat capacity.

By Prandtl number $Pr = \frac{\mu c_p}{k}$, (3.22) can be written as:

$$\frac{\partial T}{\partial t} + \nabla \cdot (T U) = \nabla \cdot \left(\frac{\nu}{Pr} \nabla T \right) \quad (3.23)$$

3.3 Main features of OpenFOAM for CFD

OpenFOAM is not popular in CFD simulations for nanofluids yet. However, it is becoming an important tool for normal fluid flow and heat transfer analysis (Chen et al., 2014a). Due to the ‘open’ features, OpenFOAM definitely has a great potential to play a much more important role in nanofluid CFD simulations. In OpenFOAM, the concepts of tensor and domain discretisation

are the two most important natures to ensure a reliable CFD simulation.

3.3.1 Tensors

Tensors are geometric objects belonging to a particular space and obeying some certain mathematical rules (Aris, 2012). They are used to describe linear relations between vectors, scalars, and other tensors. Specifically, tensors are represented by a set of component values relating to a set of unit base vectors (Guide, 2011). In OpenFOAM, the unit base vectors i_x , i_y and i_z are aligned with the right-handed rectangular Cartesian coordinate axes x , y and z , respectively (Fig. 3.2).

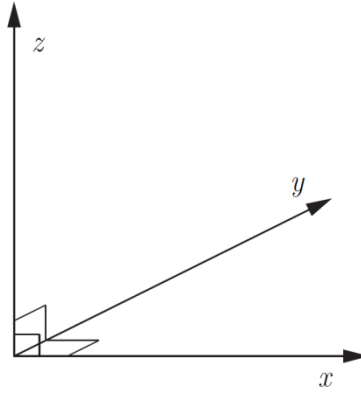


Fig. 3.2 Right handed Cartesian coordinate system

Basically, there are two attributes for a tensor: dimension d and rank r . In OpenFOAM, dimension d is set as three, it offers tensor rank r a range of $0 \sim 2$. When $r = 0$, a ‘scalar’ can be represented by a single real number, e.g. pressure p , temperature T and volume fraction ϕ etc in present work. When $r = 1$, a ‘vector’ can be represented by parameters of both magnitude and direction. In present work, velocity U and force F are presented by vectors. In component form, vector $a = (a_1, a_2, a_3)$ is related to a set of Cartesian axes x , y and z , respectively. When $r = 2$, a ‘tensor’ can be expressed in array notation as:

$$T_{ij} = \begin{pmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{pmatrix} \quad (3.24)$$

where a certain component T_{ij} is represented by indices i and j . The components for $i = j$ are referred to as diagonal components, while those for $i \neq j$ are referred to as off-diagonal components.

OpenFOAM contains a C++ class library for tensor mathematics. The basic tensor classes in OpenFOAM are recognised as ‘scalar’, ‘vector’ and ‘tensor’, the values of which can be accessed by functions as shown in Tab. 3.1.

Tab. 3.1 Basic tensor classes in OpenFOAM

Rank	Basic class	Access functions
0	scalar	~
1	vector	x(), y(), z()
2	tensor	xx(), xy(), xz()

Before performing algebraic operations, all variables must have appropriate units to ensure the calculation is physically meaningful. In OpenFOAM, it is one of the most primary rules that the values at left and right hand sides of an equation must have exactly same dimensional units. For this, it is encouraged that each variable should be given an appropriate dimensional unit in an array. For instance, ‘ T [0 0 0 1 0 0 0] 300’ indicates that the temperature T is defined as $T = 300K$. The rules for unit definition can be found in Tab. 3.2, the first four of which are needed in present work.

Tab. 3.2 Seven units in OpenFOAM

Sequence in array	Physical meaning in OpenFOAM	Unit
1	Mass	kilogram
2	Length	metre
3	Time	second
4	Temperature	Kelvin
5	Quantity	Moles
6	Current	Ampere
7	Luminous intensity	Candela

3.3.2 Discretisation of computational domain and boundary conditions

Discretisation can be performed by approximating a problem into discrete quantities. This can be operated for solution domain, time domain and algebraic equations. More specifically, space domain can be discretised into computational mesh on which the PDEs are subsequently discretized, while time (if required) can be broken into a set of time steps (Fig. 3.3). In OpenFOAM, time step could depend on Courant number, which is calculated during the simulation (ÇEr et al., 2014).

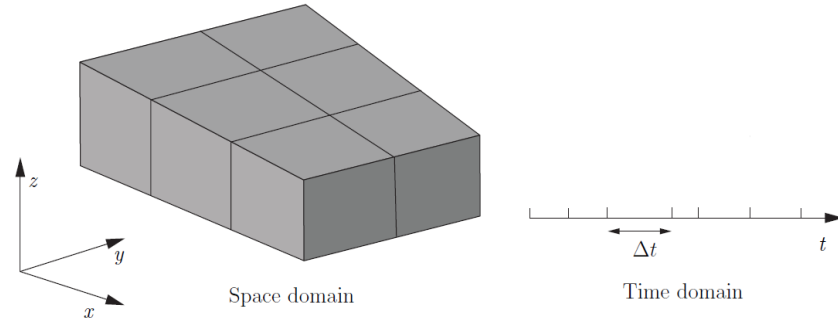


Fig. 3.3 Discretisation of solution and time domains (Guide, 2011)

Particularly, space discretisation requires the subdivision of computational domain into a number of cells (control volumes). In OpenFOAM, there is no limitation on the number of faces bounding each cell, nor any restriction on the alignment of each face. The mesh based on such space discretisation is often referred to as ‘arbitrarily unstructured’ (Fig. 3.4). However in present work, structured mesh is applied for all the cases due to its advantages of high efficiency and simpler data structure (Jacquotte and Coussement, 1992).

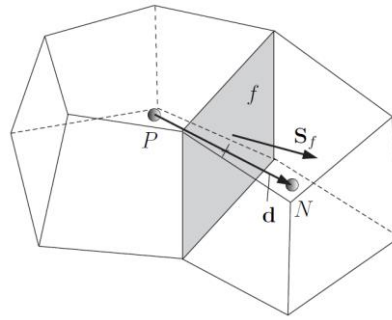


Fig. 3.4 Key parameters in finite volume discretisation (Guide, 2011)

As the most basic mesh class in OpenFOAM, ‘*polyMesh*’ is constructed by the minimum information required to define a mesh geometry. For this, four lists, i.e. ‘*points*’, ‘*faces*’, ‘*boundary*’ and ‘*cellZones*’ should be defined first in a case folder (Tab. 3.3).

Tab. 3.3 Four lists in OpenFOAM to describe mesh

List name	Content	Location
<i>points</i>	A list of cell vertex point coordinate vectors.	<i>Case/constant/polyMesh</i>
<i>faces</i>	A list of face numbers and cell vertex points.	<i>Case/constant/polyMesh</i>
<i>boundary</i>	A list of patches with information such as names, groups and start face.	<i>Case/constant/polyMesh</i>
<i>cellZones</i>	A list of cell numbers.	<i>Case/constant/polyMesh</i>

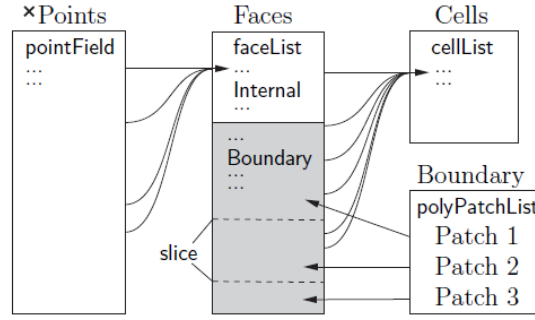


Fig. 3.5 Schematic mesh description in OpenFOAM

In OpenFOAM, '*fvMesh*' class is created through discretisation of geometric field and used to store additional data needed for following calculations (Tab. 3.4). '*fvMesh*' class supports standard algebraic matrix operations such as addition, subtraction and multiplication. These data stored in '*fvMesh*' class are the basis of all the other calculations for PDEs and new variables. For instance, face area S_f in Fig. 3.4 is necessary to estimate the heat flux across a cell face.

Tab. 3.4 Stored data in '*fvMesh*' class

Class	Description	Symbol
<i>volScalarField</i>	Cell volumes	V
<i>surfaceVectorField</i>	Face area vectors	S_f
<i>surfaceScalarField</i>	Face area magnitudes	$ S_f $
<i>volVectorField</i>	Cell centres	C
<i>surfaceVectorField</i>	Face centres	C_f
<i>surfaceScalarField</i>	Face motion fluxes	ϕ_g

Discretisation of N–S equations converts PDEs into a set of algebraic equations for all cells in computational domain. The algebraic equation for a cell as in Fig. 3.4 is commonly expressed in a matrix form as:

$$[A][x] = [B] \quad (3.25)$$

where A is a square matrix storing the coefficients of a set of algebraic equations, x is column vector of dependent variable and B is source vector.

Each term in a PDE is represented individually in OpenFOAM code by the classes of static functions '*finiteVolumeMethod*' and '*finiteVolumeCalculus*', which are abbreviated as '*fvM*' and '*fvC*', respectively. '*fvM*' and '*fvC*' contain static functions representing differential operators such as ' ∇ ', ' $\nabla \cdot$ ' and ' ∇^2 '. More specifically, functions of '*fvM*' calculate implicit derivatives of and return an *fvMatrix*<Type>, while functions of '*fvC*' calculate explicit derivatives and other explicit

calculations, returning a *geometricField<Type>*. Further details can be found in (Guide, 2011).

Finite volume discretisation of each term in momentum equation (3.15) is generated by integrating the term over a cell volume V as shown in Fig. 3.4. Most spatial derivative terms are then converted to integrals over the cell surface S around the volume using Gauss's theorem (Simmons, 1985):

$$\int_V \nabla \star \phi dV = \int_S dS \star \phi \quad (3.26)$$

where S is surface area and the information is stored in '*surfaceVectorField*'. ϕ presents any tensor fields and the star notation ' $\star$ ' is used to present any tensor products (i.e. gradient ' $\nabla\phi$ ' and divergence ' $\nabla \cdot \phi$ ' in present work).

Volume and surface integrations are then linearised using appropriate schemes provided by OpenFOAM. Again, taking a typical cell in Fig. 3.4 as example, time derivative term $\frac{\partial(\rho\phi)}{\partial t}$, gradient term ' $\nabla\phi$ ', divergence term ' $\nabla \cdot \phi$ ', convection term ' $\nabla \cdot (\rho U\phi)$ ' and Laplacian term ' $\nabla^2\phi$ ' will be presented in this section.

The time derivative term $\frac{\partial(\rho\phi)}{\partial t}$ can be integrated over a control volume with either Euler implicit scheme or backward differencing (Fezoui and Stoufflet, 1989). However, '*Euler*' scheme is the default setup in most OpenFOAM tutorial cases:

$$\frac{\partial}{\partial t} \int_V \rho\phi dV = \frac{(\rho_P\phi_P V)^n - (\rho_P\phi_P V)^0}{\Delta t} \quad (3.27)$$

where $\phi^n = \phi(t + \Delta t)$ is the value at current time step, $\phi^0 = \phi(t)$ is stored as the value at previous time step and subscript P denotes the cell centre point.

The gradient term ' $\nabla\phi$ ' can be evaluated in a variety of ways, in which '*Gauss linear*' is the default setup in most OpenFOAM tutorial cases:

$$\int_V \nabla\phi dV = \int_S dS\phi = \sum_f S_f\phi_f \quad (3.28)$$

where ϕ_f is face field. The schemes for estimation of ϕ_f can be controlled by file '*fvSchemes*', which is located in folder '*system*' of the case in consideration.

In OpenFOAM, central differencing, upwind differencing and blended differencing are the

three predefined schemes for differencing operation (Tab. 3.5).

Tab. 3.5 Differencing schemes for face flux estimation in OpenFOAM

Differencing scheme	Expression	Note
Central differencing (CD)	$\phi_f = f_x \phi_P + (1 - f_x) \phi_N$	$f_x = \overline{fN}/\overline{PN}$, where $\overline{fN}$ is the distance between f and cell centre N , while $\overline{PN}$ is the distance between cell centres P and N .
Upwind differencing (UD)	$\phi_f = \begin{cases} \phi_P, & F \geq 0 \\ \phi_N, & F < 0 \end{cases}$	F indicates flow direction.
Blended differencing	$\phi_f = (1 - \gamma)(\phi_f)_{UD} + \gamma(\phi_f)_{CD}$	γ is blending coefficient.

The divergence term ' $\nabla \cdot \phi$ ' can be integrated over a control volume and linearised as:

$$\int_V \nabla \cdot \phi dV = \int_S dS \cdot \phi = \sum_f S_f \cdot \phi_f \quad (3.29)$$

The convection term ' $\nabla \cdot (\rho U \phi)$ ' can be integrated over a control volume and linearised as:

$$\int_V \nabla \cdot (\rho U \phi) dV = \int_S dS \cdot (\rho U \phi) = \sum_f S_f \cdot (\rho U)_f \phi_f = \sum_f F \phi_f \quad (3.30)$$

The Laplacian term ' $\nabla^2 \phi$ ' can be integrated over a control volume and linearised as:

$$\int_V \nabla \cdot (\Gamma \nabla \phi) dV = \int_S dS \cdot (\Gamma \nabla \phi) = \sum_f \Gamma_f S_f \cdot (\nabla \phi)_f \quad (3.31)$$

where Γ is the diffusivity for quantity ϕ (Ferziger and Perić, 2002).

By length vector d between the cell centre of interest P and the centre of a neighbouring cell N , the term $S_f \cdot (\nabla \phi)_f$, which is orthogonal to the face plane S_f can be defined as (3.32). In the case of non-orthogonal meshes, an additional explicit term will be introduced (which is evaluated by interpolating cell centre gradients).

$$S_f \cdot (\nabla \phi)_f = |S_f| \frac{\phi_N - \phi_P}{|d|} \quad (3.32)$$

To ensure the solutions for PDEs are unique, continuous problem requires information about solutions at boundaries. For those control volumes next to boundaries in FVM, the boundary conditions require that the equations should be modified somewhat. In OpenFOAM, it is necessary to specify boundary conditions on all boundary faces. The boundary type could be either Dirichlet or Neumann as shown in Tab. 3.6 (Chung, 2010):

Tab. 3.6 Two boundary types in OpenFOAM

Boundary type	Information to provide
Dirichlet	Value of the dependent variable on the boundary.
Neumann	Gradient of the variable normal to the boundary.

For fixed value boundary, real values ϕ_b can be substituted directly to where the values are required. However, for fixed gradient boundary, g_b is a specification on inner product of the gradient and the unit normal to computational field boundary. It can be estimated by (3.33). Furthermore, gradient g_b can be substituted directly as (3.34) in those cases where the discretisation requires face gradient. But if the discretisation required value ϕ_f on a boundary face, an interpolation should be carried out as (3.35).

$$g_b = \left(\frac{S}{|S|} \cdot \nabla \phi \right)_f \quad (3.33)$$

$$S_f \cdot (\nabla \phi)_f = |S_f| g_b \quad (3.34)$$

$$\phi_f = \phi_p + d \cdot (\nabla \phi)_f = \phi_p + |d| g_b \quad (3.35)$$

3.4 Treatment of N–S equations in OpenFOAM

To ensure governing equations are able to be solved by computer program, the computational region can be divided into a finite number of nodes or volumes. When the concept of finite volume is employed, it is referred to as finite volume method (FVM) (Ferziger and Perić, 2002). OpenFOAM programming is based on FVM.

After the computational region has been divided into small elements, the equations system can be discretised to form matrices. For discretisation of N–S equations, continuity equation (3.4) and momentum equation (3.16) for incompressible fluid can be simply rewritten in general forms as (Jasak, 1996)

$$\nabla \cdot U = 0 \quad (3.36)$$

and

$$\frac{\partial U}{\partial t} + \nabla \cdot (UU) - \nu \nabla^2 U = -\nabla p^* \quad (3.37)$$

where $p^* = \frac{p}{\rho} + g$ is the pressure term combining with gravity g (this term may need

corrections with Boussinesq assumption when buoyancy force is needed, the details of which are given in (3.17)).

Equation (3.37) is non-linear because of the convection term $\nabla \cdot (UU)$. After the computational region has been divided into a number of control volumes, the convection term $\nabla \cdot (UU)$ can be described as (Jasak, 1996):

$$\nabla \cdot (UU) = a_P U_P + \sum_N a_N U_N \quad (3.38)$$

where coefficients a_P and a_N are functions of velocity U (cells P and N can be seen in Fig. 3.4).

Because continuity equation (3.36) and momentum equation (3.37) should be solved together during CFD simulations, this procedure will result in an even larger non-linear system. Comparatively, linearisation of the terms is preferred rather than using a solver particularly for non-linear systems. This means that a solved velocity U satisfying continuity equation (3.36) will be used to calculate coefficients a_P and a_N .

3.4.1 Pressure equation

According to (3.38), a semi-discretised form of momentum equation (3.39) can be used for pressure-velocity coupling. The pressure gradient term ∇p^* is not discretised at this stage, for the reason that it needs to be left as the unknown in pressure equation later (Rhie and Chow, 1983).

$$a_P U_P = \mathbf{H}(\mathbf{U}) - \nabla p^* \quad (3.39)$$

Term $\mathbf{H}(\mathbf{U})$ at the right hand side of (3.39) has two parts:

- ‘transport’ part, including the matrix coefficients for all neighbours multiplied by corresponding velocities (i.e. the first term at the right hand side of (3.40)).
- ‘source’ part, including the transient term and all the other source terms apart from pressure gradient (i.e. the second term at the right hand side of (3.40)).

$$\mathbf{H}(\mathbf{U}) = -\sum_N a_N U_N + \frac{U^0}{\Delta t} \quad (3.40)$$

According to (3.39), velocity U_P for cell P can be expressed as:

$$U_P = \frac{\mathbf{H}(\mathbf{U})}{a_P} - \frac{\nabla p^*}{a_P} \quad (3.41)$$

And those velocities on cell faces can be expressed as the interpolation of (3.41):

$$U_f = \left[\frac{\mathbf{H}(\mathbf{U})}{a_P} \right]_f - \left(\frac{1}{a_P} \right)_f (\nabla p^*)_f \quad (3.42)$$

Face velocity U_f can be used to calculate those fluxes across cell faces, such as momentum flux and heat flux.

According to the definition of divergence, continuity equation (3.36) can be discretised as:

$$\nabla \cdot \mathbf{U} = \sum_f S \cdot U_f = 0 \quad (3.43)$$

Substituting (3.42) to (3.43), it can be obtained that:

$$\sum_f S \cdot \left(\frac{1}{a_P} \right)_f (\nabla p^*)_f - \sum_f S \cdot \left[\frac{\mathbf{H}(\mathbf{U})}{a_P} \right]_f = 0 \quad (3.44)$$

Thus the pressure equation is:

$$\nabla \cdot \left(\frac{1}{a_P} \nabla p^* \right) = \sum_f S \cdot \left[\frac{\mathbf{H}(\mathbf{U})}{a_P} \right]_f \quad (3.45)$$

Then the final form of discretised incompressible N-S equations system can be written as:

$$a_P U_P = \mathbf{H}(\mathbf{U}) - \sum_f S (p^*)_f \quad (3.46)$$

The pressure equation for programming can be written as:

$$\sum_f S \cdot \left(\frac{1}{a_P} \right)_f (\nabla p^*)_f = \sum_f S \cdot \left[\frac{\mathbf{H}(\mathbf{U})}{a_P} \right]_f \quad (3.47)$$

And the face flux is:

$$F = S \cdot U_f = S \cdot \left\{ \left[\frac{\mathbf{H}(\mathbf{U})}{a_P} \right]_f - \left(\frac{1}{a_P} \right)_f (\nabla p^*)_f \right\} \quad (3.48)$$

According to above derivations, it can be concluded that when pressure equation (3.45) is satisfied, face flux F in (3.48) is also guaranteed to be conservative. In OpenFOAM, implementations of (3.47) can be found in file '*pEqn.H*' of a solver as the core of pressure equation.

3.4.2 SIMPLE, PISO and PIMPLE algorithms

Compared to simultaneous algorithms (Caretto et al., 1972, Vanka, 1986) which solve

complete equations system simultaneously over the whole domain, the segregated approach used by OpenFOAM is to solve equations in sequence. With this idea, semi-implicit method for pressure linked equations (SIMPLE) algorithm (Patankar, 1980) is usually employed for steady flow simulations, while pressure implicit with split operator (PISO) and PIMPLE (PISO+SIMPLE) algorithms (Issa et al., 1991) are usually employed for transient flow simulations.

Briefly, SIMPLE method is designed to take advantage of following facts (Jasak, 1996):

- Velocity field approximation is obtained by solving momentum equation. Pressure gradient term is calculated using the pressure distribution from previous iteration (or an initial guess). Equation (3.39) is under-relaxed by coefficient α_U in an implicit manner:

$$\frac{a_p}{\alpha_U} U_p^n = \mathbf{H}(\mathbf{U}) - \nabla p^* + \frac{1-\alpha_U}{\alpha_U} a_p U_p^0 \quad (3.49)$$

- Pressure equation is formulated and solved for the new pressure distribution.
- A new set of conservative fluxes is calculated using (3.48). It is accurate enough to obtain an approximation of the pressure field and recalculate the $\mathbf{H}(\mathbf{U})$ coefficients with the new set of conservative fluxes. Pressure solution p^* is therefore under-relaxed by factor α_p (pressure under-relaxation), the solution of pressure equation p^{*s} and the pressure field used in momentum predictor p^{*0} .

$$p^* = p^{*0} + \alpha_p(p^{*s} - p^{*0}) \quad (3.50)$$

For velocity and pressure under-relaxation factors, Peric (1985) suggested that $\alpha_U = 0.8$ and $\alpha_p = 0.2$ should be used for velocity and pressure respectively.

And PISO algorithm, which is an extension of the SIMPLE algorithm, can be described as working in following manners (Jasak, 1996):

- Momentum equation (3.46) is solved first. Because the exact pressure gradient source term is not known at this stage, thus the pressure field from previous time-step is used instead. The solution gives an approximation of the new velocity field U_p . This stage is referred to as momentum predictor.
- Using the predicted velocities, operator $\mathbf{H}(\mathbf{U})$ can be assembled according to (3.40) and

the pressure equation can be formulated by (3.45). The solution of pressure Laplacian equation gives the first estimation of new pressure field p^* . This step is referred to as pressure solution.

- Expression (3.48) gives a set of conservative fluxes consistent with the latest pressure field. The velocity field should also be corrected as a consequence of the new pressure distribution. Velocity correction is done in an explicit manner using (3.41).

However, the velocity correction actually consists of two parts: the correction due to the change in pressure gradient term $\frac{\nabla p^*}{a_p}$ and the transported influence of corrections of neighbour cell velocities term $\frac{H(U)}{a_p}$. It is necessary to correct term $H(U)$ to formulate the new pressure equation and repeat the procedure instead of doing explicit correction for term $\frac{\nabla p^*}{a_p}$ only. In other words, PISO loop consists of an implicit momentum predictor and an explicit velocity correction. The loop is repeated until a pre-determined tolerance is reached. In OpenFOAM, the values for tolerance are defined in file '*fvSolution*', which is located in folder '*system*' of the case in consideration.

According to SIMPLE and PISO algorithms discussed above, it can be summarised that SIMPLE algorithm is essentially a guess-and-correct procedure for pressure calculation, while PISO algorithm involves one predictor step and two corrector steps. PISO may be considered as an extension of SIMPLE, but with a further corrector step to enhance it. Based on the above two approaches, a 'PISO+SIMPLE' method can be applied to couple velocity and pressure for final solutions. This is the origin of the name PIMPLE (Chen et al., 2014b), i.e. a combination of PISO and SIMPLE algorithms. However, PIMPLE is still a PISO scheme, but with outer correctors which correct pressure value before the next time-step. It should be noticed that when the value of parameter '*nOuterIterations*' in OpenFOAM is defined as 1, PIMPLE becomes to PISO (Di Stefano, 2014). In this project, PIMPLE algorithm is applied for the solvers under both Eulerian-Lagrangian and Eulerian frames.

3.5 Conclusions

In this chapter, some CFD basics such as governing equations for mass, momentum and

energy are discussed. The main features of OpenFOAM for CFD are also introduced. Based on what, the treatment of N–S equations in OpenFOAM is also provided as the basis for solver developments in the following chapters. By the CFD basics introduced in this chapter, it can be found that, although no works have been published so far to develop any numerical solvers particularly for nanofluid simulations, however, it is actually possible to do so based on previous matured theoretical basis in CFD.

4. EULERIAN-LAGRANGIAN APPROACH FOR NANOFLUID SIMULATION WITH OpenFOAM

4.1 Introduction

Lagrangian frame has been applied to nanofluid simulation since (Bianco et al., 2009). Technically, Eulerian-Lagrangian approach is believed to be more practical than others for nanofluid simulations because it is possible to treat nanofluid as real two-phase mixture consisting of dispersed nanoparticles and continuous basefluid. However, because OpenFOAM does not have any solvers coupling particles motion and continuous fluid flow simulations to each other, no works applying OpenFOAM to nanofluid with Eulerian-Lagrangian approach have been published yet.

In this chapter, a new OpenFOAM solver is developed under Eulerian-Lagrangian frame. For this, some functions provided by solver ‘buoyantBoussinesqPimpleFoam’ and solver ‘icoUncoupledKinematicParcelFoam’ are going to be combined in an appropriate way. More specifically, the basic strategy of solving continuous fluid flow and heat transfer in solver ‘buoyantBoussinesqPimpleFoam’ and the basic function of particles tracking in solver ‘icoUncoupledKinematicParcelFoam’ are combined. In this project, this newly developed solver is named as ‘nanofluidELFoam’. Solver ‘nanofluidELFoam’ is expected to be able to carry out the simulations of fluid flow, heat transfer and particles tracking at same time (Tab. 4.1).

Tab. 4.1 Features comparison for the three solvers

Solver	Fluid flow	Fluid heat transfer	Particles tracking
buoyantBoussinesqPimpleFoam	Yes	Yes	No
icoUncoupledKinematicParcelFoam	No	No	Yes
nanofluidELFoam	Yes		

Furthermore, regarding the issues of collision between particles as well as the collision between particles and walls, ‘nanofluidELFoam’ will provide a platform, on which such collisions can be switched on/off to achieve different coupling methods, such as ‘one-way’ coupling, ‘two-way’ coupling and ‘four-way’ coupling (Elghobashi, 1994).

4.2 Discrete element method (DEM) in OpenFOAM

4.2.1 Basic idea

Motion and discrete element method (Glatzel et al.) (Glatzel et al.) was proposed by P.A. Cundall in 1979 to simulate motions of large number of discrete objects and particles (Cundall and Strack, 1979). With this method, the state of each particle is calculated using classical mechanics. DEM has been proved as a powerful approach to investigate the mixing process of particles as it can provide precise description of both particle-particle and fluid-particle interactions (Xu and Yu, 1997).

In a typical DEM approach, the motion of each individual particle is tracked based on the second Newton's law (4.1), while the translational and rotational motions of particle i can be expressed by (4.2) (Ren et al., 2012).

$$m_i \frac{dU_{pi}}{dt} = m_i g + F_i \quad (4.1)$$

$$I_i \frac{d\omega_i}{dt} = \sum_{j=1}^{n_i} (T_{tij} + T_{nij}) \quad (4.2)$$

where m_i , U_{pi} , I_i and ω_i are mass, velocity, inertia and rotational velocity of particle i , respectively. F_i is a combination of forces, possibly including drag force $F_{D,i}$, contact force $F_{C,i}$, Saffman lift force $F_{S,i}$ and Magnus lift force $F_{M,i}$, etc (He et al., 2009). Torques T_{tij} and T_{nij} are generated by tangential forces and rolling friction respectively.

DEM was introduced to OpenFOAM since the version 2.0.0 (which was released in 2011). Based on this idea, 'icoUncoupledKinematicParcelFoam' is a transient solver for passive transport of a single kinematic particle cloud. In recent years, the solver has been validated as a powerful tool to investigate the collisions amongst different particles, as well as the collisions between particles and solid walls by the basic process shown in Fig. 4.1 (Yt Feng et al., 2015, Vaish, 2014, Breinlinger et al., 2015).

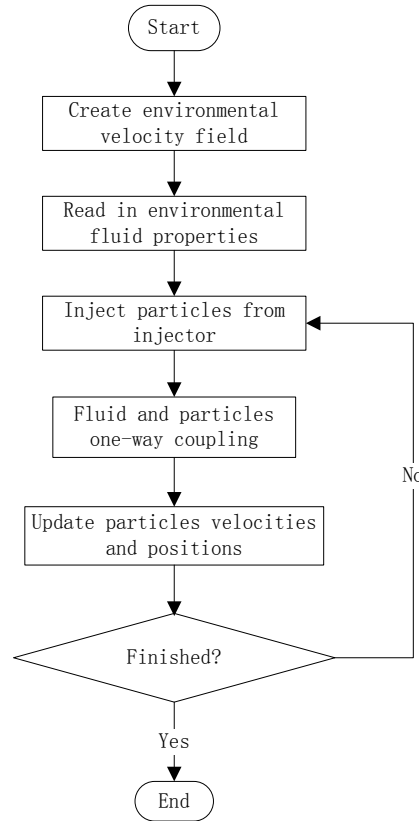


Fig. 4.1 Flow chart of solver ‘icoUncoupledKinematicParcelFoam’

From above DEM governing equations and the implementation flow chart in OpenFOAM, it can be seen that DEM could be a good tool to treat CFD of nanofluids in a Lagrangian way. By which, not only nanoparticles can be tracked reasonably as in previous publications, but the particle-particle and particle-wall collisions also can be considered. This is potential to make the Lagrangian CFD approach for nanofluid is even more reliable. However, before being able to use OpenFOAM for nanofluid investigation in a Lagrangian way, two problems must be solved.

Firstly, for particles motion, a pre-defined velocity field is given before simulation to provide environmental forces from continuous phase. In other words, there are no governing for continuous phase and the fluid field cannot be updated with time. When the continuous fluid and particles are coupled in original solver ‘icoUncoupledKinematicParcelFoam’, only the forces from fluid to particles can be considered. Secondly, in most previous references such as (Bianco et al., 2009), (Pallares and Grau, 2010) and (Moraveji and Esmaeili, 2012), etc, only drag force was considered. However, to ensure the simulations are more reasonable, more force models for thermophoretic force, Brownian force, Saffman force, pressure gradient force and virtual mass, etc should be added (He et al., 2009).

4.2.2 Parcels

When DEM approach is applied, the mass flow rate could be converted into the number of particles injected during per unit time. However, it is often prohibitive to track that number of particles in a simulation. Therefore, the model will track a number of ‘parcels’, and each parcel is representative of a fraction of the total mass flow released during a time step. With such an idea, mass and size of the numerical parcel are critical for forces estimations.

In terms of parcel mass, it is calculated as a combination of all the single particles in this parcel. Considering a parcel containing k single particles, the parcel mass m_p is calculated as in (4.3). Assuming that the parcel is still spherical, then the parcel size d_p can be determined by (4.4), i.e. it is a sphere whose volume is the mass of the entire parcel m_p divided by solid phase (particle) density ρ_p .

$$m_p = \sum_{i=1}^k m_i \quad (4.3)$$

$$d_p = \sqrt[3]{\frac{6m_p}{\pi\rho_p}} \quad (4.4)$$

By parcel concept in DEM approach, the requirement of computational resources can be considerably reduced, but the mass fraction of the discrete phase still can be retained. However, it should be noticed that the parcel concept is a numerical solution to overcome the problem of computational resource. It is not an effective way to consider nanoparticle cluster effects. Therefore a numerical parcel is still considered as solid rather than porous. In present work under Eulerian-Lagrangian frame, the forces acting on nanoparticles are considered to act on a parcel of nanoparticles. When the parcel concept is employed in the following coding works, parcel mass m_p is used instead of a single nanoparticle mass m_i , and parcel size d_p is used instead of a single nanoparticle size d_i .

The parcel concept is also employed by OpenFOAM (Garg et al., 2012). If a numerical parcel is to present k real particles, then $n = N/k$ parcels (the value of N/k will be approximated to be integral) need to be generated/injected for the system to present N real particles. In OpenFOAM 2.3.1, nine injection models (such as ‘CellZoneInjection’, ‘PatchInjection’, ‘ConeInjection’ and ‘ConeNozzleInjection’ etc) can be employed to inject numerical parcels into the computational region. In which, model ‘ConeInjection’ can be used to inject parcels at inlet

boundary for forced flow in the cases with inlet boundaries, while a self-defined parcels position file can be used via model ‘manualInjection’ for the cases without inlet boundaries.

When parcels are generated at $t = 0s$, parcel size distribution can be treated in the same way as that for a single particle. In OpenFOAM 2.3.1, seven particle size models (such as ‘exponential’, ‘fixedValue’, ‘RosinRammler’ and ‘general’ etc) can be found in a folder ‘*distributionModels*’ which is located in folder category ‘src’. Among these models, ‘fixedValue’ is the most commonly used one, in which only one particle size value is needed to define. However, model ‘RosinRammler’ is actually a more practical option because it was reported that nanoparticles size usually conforms to a sort of mathematical distribution (e.g. normal distribution), but not same as the value given by fabrication manual (Anoop et al., 2009).

Rosin-Rammler distribution is frequently used to describe the particle size distribution in practical engineering (4.5). The function is particularly suited to be applied for those particles generated by grinding, milling and crushing operations (Brown and Wohletz, 1995).

$$F(x, n, d) = 1 - e^{-x/d^n} \quad (4.5)$$

where F is the mass fraction of particles with diameter smaller than x , d is the mean particle diameter and n is a factor to set the spread of particle size. The value of x varies in a given range of particles size (restricted by ‘maxVaule’ and ‘minValue’ in (4.8)).

In OpenFOAM 2.3.1, the real particle size d_{real} is calculated by:

$$d_{real} = minValue + d \times \left\{ -\ln \left[1 - randomNumber \times (1 - e^{-\left(\frac{maxVaule - minValue}{d}\right)^n}) \right] \right\}^{1/n} \quad (4.6)$$

where the values of ‘maxVaule’, ‘minValue’, d and n can be defined in file ‘*kinematicCloudProperties*’ of the case in consideration. However, when parcel size is needed, d_p in (4.4) will be used instead of a single particle size d .

4.3 Forces consideration for nanoparticles

4.3.1 Surface and body forces

In OpenFOAM 2.3.1, the force vectors acting on a particle can be divided to two types according to their calculation methods: implicit and explicit as in (4.7). Taking solver ‘icoUncoupledKinematicParcelFoam’ as an example, governing equations (4.1) is employed to

control particles motion and (4.7) is used to summarise the total force acting on a certain particle i , i.e. (4.7) is for the term F_i in (4.1). However, it should be noticed that, although a single spherical particle is used in this section to present both the surface and body force estimations, parcel mass m_p and size d_p will be used for forces calculation instead of m_i and d_i for single particles when the parcel concept is employed.

$$F = S_p(U_f - U_p) + S_u \quad (4.7)$$

where scalar S_p presents the implicit coefficient, while vector S_u presents the explicit contribution.

Drag force

Because nanoparticles are normally ultra small and the real shape is rather difficult to describe, it is practical to assume that nanoparticle is spherical in 3-D space (Kebblinski et al., 2002). With this consideration, drag force $F_{D,i}$ acting on the i th particle can be calculated in a general form:

$$F_{D,i} = \frac{1}{2} c_d Re \rho_f (U_f - U_p)^2 A_p \quad (4.8)$$

where c_d is drag coefficient, Re is particle Reynolds number, the product of $c_d Re$ can be calculated by (4.9), ρ_f is basefluid density, U_f is basefluid velocity, U_p is particle velocity and A_p is particle projection area in moving direction.

$$c_d Re = \begin{cases} 0.424 Re, & Re > 1000 \\ 24 \left[1 + \frac{Re^{2/3}}{6} \right], & Re \leq 1000 \end{cases} \quad (4.9)$$

With the calculation $A_p = \frac{\pi d_p^2}{4}$ for particle projection area, if multiplied by particle mass m_p and divided by particle mass $\frac{\pi \rho_p d_p^3}{6}$, (4.8) becomes to:

$$F_{D,i} = \frac{3 c_d Re \mu_f m_p (U_f - U_p)}{4 \rho_p d_p^2} \quad (4.10)$$

The value of coefficient $\frac{3 c_d Re \mu_f m_p}{4 \rho_p d_p^2}$ will be collected to container scalar S_p as in (4.11) and eventually summed up together with other force sources.

$$S_p = \frac{3 c_d Re \mu_f m_p}{4 \rho_p d_p^2} \quad (4.11)$$

Buoyancy force and gravity

Buoyancy force is usually considered together with gravity force. The combination of the two forces can be calculated as:

$$F_{BG,i} = m_p g (1 - \rho_f / \rho_p) \quad (4.12)$$

where ρ_f and ρ_p present densities of fluid and particle respectively.

The result of $F_{BG,i}$ calculated by (4.12) is added to container vector S_u in (4.7) and eventually calculated together with other explicit force sources.

Thermophoretic force

The diffusion of particles due to a tangential temperature gradient on the particle surface generates a thermophoretic force. To a single particle i , the thermophoretic force $F_{T,i}$ can be estimated by (McNab and Meisen, 1973):

$$F_{T,i} = -D_T \frac{1}{m_p T} \nabla T \quad (4.13)$$

where ∇T is the temperature difference between the cell with this particle and the neighbouring cells, while D_T is the thermophoretic coefficient defined as:

$$D_T = \frac{6\pi d_p \mu_f^2 C_s (k_f/k_p + C_t Kn)}{\rho_f (1 + 3C_m Kn) (1 + 2k_f/k_p + 2C_t Kn)} \quad (4.14)$$

where $C_s = 1.146$, $C_t = 1.147$ and $C_m = 2.18$, while Kn is Knudsen number, which is defined by $Kn = 2\lambda/d_p$ (Talbot et al., 1980). λ is mean free path of the fluid molecules and a reference value $\lambda = 2.5 \times 10^{-10}m$ can be used.

The result of $F_{T,i}$ calculated by (4.13) is added to container vector S_u in (4.7) and eventually calculated together with other explicit force sources.

Brownian force

Ultra small particles suspended in fluid are likely to have random motions resulting from their collisions with the fast-moving atoms or molecules of the liquid. This is known as Brownian motion. Correspondingly, the force leads to such a motion is called Brownian force and can be estimated as (Li and Ahmadi, 1992):

$$F_{B,i} = \zeta_i \sqrt{\frac{\pi S_i}{\Delta t}} \quad (4.15)$$

where ζ_i is the unit-variance-independent Gaussian random number with zero-mean.

The components of the Brownian force are modelled as a Gaussian white noise process with spectral intensity S_i :

$$S_i = S_0 \delta_{ij} \quad (4.16)$$

where δ_{ij} is the Kronecker delta, and S_0 is defined as:

$$S_0 = \frac{216 \nu k_B T}{\pi^2 \rho_f d_p^5 (\rho_p / \rho_f)^2 C_c} \quad (4.17)$$

where k_B is Boltzmann constant and Stokes-Cunningham slip correction C_c is given as:

$$C_c = 1 + \frac{2\lambda}{d_p} (1.257 + 0.4e^{-\frac{1.1d_p}{2\lambda}}) \quad (4.18)$$

The result of $F_{B,i}$ calculated by (4.15) is added to container vector S_u in (4.7) and eventually calculated together with other explicit force sources.

Saffman force

Small particles in a shear field experience a lift force perpendicular to the direction of flow, the expression for such inertia shear lift was first obtained by Saffman (Saffman, 1965):

$$F_{S,i} = 1.61 (\mu_f \rho_f)^{0.25} d_p^2 (\nabla U_f)^{0.5} (U_f - U_p) \quad (4.19)$$

The value of coefficient $1.61 (\mu_f \rho_f)^{0.25} d_p^2 (\nabla U_f)^{0.5}$ will be collected to container scalar S_p as in (4.11) and eventually summed up together with other force sources.

Pressure gradient force

The pressure gradient force results from the local fluid pressure gradient around the particle and is defined as (Monaghan, 1992):

$$F_{P,i} = \frac{\rho_f}{\rho_p} U_p \nabla U_f \quad (4.20)$$

The result of $F_{P,i}$ calculated by (4.20) is added to container vector S_u in (4.7) and eventually calculated together with other explicit force sources.

Virtual mass

When a particle accelerates in fluid, an inertia is added because the particle must move some volume of surrounding fluid as it moves through it. For simplicity this can be modelled as some volume of fluid moving with the object, though in reality ‘all’ the fluid will be accelerated. This additional virtual mass force $F_{V,i}$ can be calculated as (Drew et al., 1979):

$$F_{V,i} = \frac{1}{2} \rho_f U_p \left(\frac{\partial U_f}{\partial t} - \frac{\partial U_p}{\partial t} \right) \quad (4.21)$$

The result of $F_{V,i}$ calculated by (4.21) is added to container vector S_u in (4.7) and eventually calculated together with other explicit force sources.

Magnus force

If a particle is moving in fluid with rotation, the Magnus effect which is generated by pressure gradient may need to be considered. Magnus force $F_{M,i}$ can be estimated by (Zhong et al., 2006):

$$F_{M,i} = \frac{1}{8} \pi d_p^3 \rho_f \omega_i (U_f - U_p) \quad (4.22)$$

where the particle angular speed ω_i can be obtained after solving (4.1).

The value of coefficient $\frac{1}{8} \pi d_p^3 \rho_f \omega_i$ will be collected to container scalar S_p as in (4.11) and eventually summed up together with other force sources.

4.3.2 Particle-particle and particle-wall contacting forces

In previous nanofluid investigations using Eulerian-Lagrangian approach, the impacts between particles were ignored (Bianco et al., 2009, Tahir and Mital, 2012). However, the impacts between particles are assumed to be one of the most possible reasons to enhance nanofluid thermal conductivity and heat transfer performance (Xuan and Li, 2000). Furthermore, due to nanoparticles Brownian motion and mutual collisions, nanofluid can be observed more stable than normal multi-phase solutions. Therefore in practice, it is better to consider nanoparticle collisions for nanofluid simulations with Eulerian-Lagrangian approach.

Generally, the methods of treating particle collisions can be sorted into two categories: soft sphere and hard sphere approaches (Mitarai and Nakanishi, 2002). Comparatively, the hard sphere approach is more suited for collision dominated dilute flows, while soft sphere approach is better for contact dominated dense flows (Kempe and Fröhlich, 2012). In OpenFOAM 2.3.1, soft sphere

approach is employed for particles collision and linear contact model is applied.

With above considerations, if there are two sphere particles i and j with diameters d_i and d_j , respectively (Fig. 4.2), the overlap δ of them can be written as:

$$\delta = \frac{d_i + d_j}{2} - (r_i - r_j) \cdot \mathbf{n}_{ij} \quad (4.23)$$

where $\mathbf{n}_{ij} = (r_i - r_j)/|r_i - r_j|$ is the unit direction vector pointing from particle i to j .

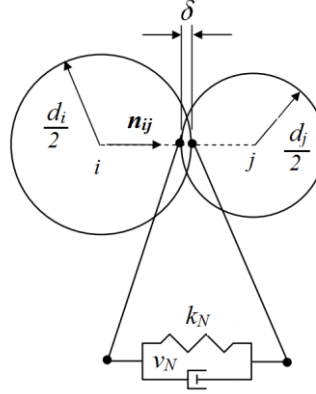


Fig. 4.2 Two particles contacting with overlap δ

The acting force F_P due to collision between particles i and j can be decomposed into a normal one and a tangential one as:

$$F_{P,ij} = F_{nij} + F_{tij} \quad (4.24)$$

Normal force F_{nij} can be calculated by Hertz's law as shown in (4.25) (Hertz, 1882). This linear spring dashpot model allows to consider particle contact as a damped harmonic oscillator. It is employed by OpenFOAM 2.3.1 for calculation of both normal and tangential forces between particles. The calculation program is predefined and stored in file '*PairSpringSliderDashpot.C*', which is located in folder category '*src*'.

$$F_{nij} = k_n \delta_n^b + \gamma_n v_n \quad (4.25)$$

where k_n is normal spring stiffness, δ_n is overlap in normal direction of particle surface, b is collision constant and the default value is set as $b = 1.5$ in most tutorial cases (defined in '*kinematicCloudProperties*' file of the case in consideration), γ_n is normal viscous damping constant and $v_n = -(v_i - v_j) \cdot \mathbf{n}_{ij}$ is the relative velocity between particles i and j in normal direction.

Particle spring stiffness k_n is obtained by (4.26) (Antypov and Elliott, 2011):

$$k_n = \frac{4}{3} \sqrt{\frac{d_i d_j}{2(d_i + d_j)}} \frac{E}{2(1 - \nu_p^2)} \quad (4.26)$$

where E is Young's modulus, while ν_p is Poisson's ratio of the particle.

Viscous damping constant γ_n is given by (Chen, 2009):

$$\gamma_n = \alpha \sqrt{k_n \frac{m_i m_j}{(m_i + m_j)}} \quad (4.27)$$

where α is critical damping ratio and default as $\alpha = 0.12$ in most OpenFOAM cases.

To tangential force F_{tij} , it can be calculated in a similar way as shown in (4.25):

$$F_{tij} = k_t \delta_t + \gamma_t v_t \quad (4.28)$$

where δ_t is overlap in tangential direction, tangential viscous damping constant is $\gamma_t = \gamma_n$, v_t is sliding velocity between particles i and j , and the tangential particle spring stiffness k_t is defined as:

$$k_t = 8 \sqrt{\frac{d_i d_j}{2(d_i + d_j)}} \delta_n \frac{E}{(1 - \nu_p^2)(2 - \nu_p)} \quad (4.29)$$

In OpenFOAM 2.3.1, the relative motion between two contacting particles are estimated first to decide whether there is slip or not. For this, sliding friction F_s is calculated as a threshold:

$$F_s = \mu_p F_{nij} \quad (4.30)$$

where μ_p depends on smooth level of particle surface and can be defined in file '*kinematicCloudProperties*' of the case in consideration.

If $|F_{tij}| > |F_s|$, it means relative slip exists between the two particles, then the tangential sliding friction is $F_{tij} = -F_s$. Otherwise, tangential force F_{tij} between the two particles is calculated by (4.28). The program of above decision-making procedure can be found in file '*PairSpringSliderDashpot.C*', which is located in folder category '*src*'.

Similar to the calculation for collision force between particles, the collision force between particles and boundaries can be calculated in the same way. Again, if decomposed the collision force $F_{w,i}$ into normal and tangential directions, it should be a combination of normal force F_{nwi} and tangential force F_{twi} as:

$$F_{w,i} = F_{nwi} + F_{twi} \quad (4.31)$$

Normal force component F_{nwi} can be calculated by Hertz's law:

$$F_{nwi} = k_{nw} \delta_{nw}^b + \gamma_{nw} v_{nw} \quad (4.32)$$

where δ_{nw} is the overlap between particle and boundary, v_{nw} is relative velocity of particle to boundary. The particle spring stiffness k_{nw} is defined as:

$$k_{nw} = \frac{4}{3} \sqrt{\frac{d_i}{2}} \frac{E}{2(1-\nu_p^2)} \quad (4.33)$$

And the viscous damping constant γ_n is given by:

$$\gamma_{nw} = \alpha \sqrt{k_{nw} m_i} \quad (4.34)$$

To tangential force component F_{twi} , it can be calculated similarly as (4.28):

$$F_{twi} = k_{tw} \delta_{tw} + \gamma_{tw} v_{tw} \quad (4.35)$$

where the tangential particle spring stiffness k_{tw} is:

$$k_{tw} = 8 \sqrt{\frac{d_i}{2}} \delta_{nw} \frac{E}{(1-\nu_p^2)(2-\nu_p)} \quad (4.36)$$

For tangential force F_{twi} , it also needs to estimate whether slide happens between particle and boundary by a threshold F_{sw} , which can be calculated by:

$$F_{sw} = \mu_w F_{nwi} \quad (4.37)$$

where μ_w depends on smooth level of boundary and can be defined in file '*kinematicCloudProperties*' of the case in consideration.

If $|F_{twi}| > |F_{sw}|$, then slide exists between particle and wall, the tangential force is considered as $F_{twi} = -F_{sw}$. Otherwise, F_{twi} is calculated by (4.35). The program of this decision-making procedure can be found in file '*PairSpringSliderDashpot.C*', which is located in folder category '*src*'.

4.3.3 Order-of-magnitude analysis-the significance of DEM for nanofluid

In previous Eulerian-Lagrangian approaches for nanofluid simulations, only the interactions between nanoparticles and fluids were considered, such as the typical investigation reported by (Bianco et al., 2009). Building on that, some researchers such as He et al. (2009) and Mahdavianesh et al. (2013) considered more force models. In which, Brownian force, Saffman force, virtual mass, pressure gradient force and thermophoretic forces, etc were included for more

convincible simulations. Although most of the present available force models were established for relatively large particles and it is still not clear currently how these expressions are corrected for nanoparticles, the previous modelling results were reported to agree well with experimental data (He et al., 2009). Therefore, in the present works using Eulerian-Lagrangian approach, the force models in Section 4.3.1 are still employed and the manner suggested by (He et al., 2009) is followed.

With DEM, the particle-particle and particle-wall interactions also will be considered. The necessity of such considerations can be demonstrated by a case study, i.e. an order-of-magnitude analysis. For this, the order of particles contacting force due to collision is compared to that of particle gravity force (with buoyancy), which is one of the most important forces in nanoparticles sedimentation study. Assuming that there are two identical spherical Al_2O_3 nanoparticles ($d_i = d_j = 50nm$) are hitting each other with a relative velocity $v_{ij} = 1 \times 10^{-4}m/s$. Young's modulus and Poisson's ratio are used as $E = 400 GPa$ and $\nu_p = 0.25$ for the Al_2O_3 nanoparticles (Auerkari, 1996). When the two particles just contact, the overlap δ_n in (4.25) is $\delta_n = 0$, then the normal contact force at this moment is:

$$F_{nij} = 0.12 \sqrt{\frac{4}{3} \sqrt{\frac{d_i d_j}{2(d_i + d_j)}} \frac{E}{2(1 - \nu_p^2)} \frac{\frac{4}{3}\pi \left(\frac{d_i}{2}\right)^3 \frac{4}{3}\pi \left(\frac{d_j}{2}\right)^3}{\frac{4}{3}\pi \left(\frac{d_i}{2}\right)^3 + \frac{4}{3}\pi \left(\frac{d_j}{2}\right)^3}} v_{ij} = 3.7 \times 10^{-13} N$$

The gravity force with the consideration of buoyancy is:

$$F_{BG,i} = \frac{4}{3}\pi \left(\frac{d_i}{2}\right)^3 \rho_p g \left(1 - \frac{\rho_f}{\rho_p}\right) = 2.1 \times 10^{-18} N$$

From above forces estimations, it can be found that when the contacting force between two contacting nanoparticles is considered in DEM approach could be much greater than the gravity force in a normal Lagrangian consideration. Indicating that DEM could make the nanofluids simulations using Eulerian-Lagrangian approach more convincible, particularly in those cases when nanoparticles may hit others or walls with a noticeable speed.

4.4 Development of ‘nanofluidELFoam’

In this section, a procedure to develop the new OpenFOAM solver ‘nanofluidELFoam’ is introduced. By ‘nanofluidELFoam’, fluid flow, heat transfer and particles motion will be able to influence each other. For this purpose, some functions in original solvers ‘buoyantBoussinesqPimpleFoam’ and ‘icoUncoupledKinematicParcelFoam’ will be combined together.

In the new solver ‘nanofluidELFoam’, the simulation at each time step can be divided into two parts. Firstly, nanofluid is assumed to be a pseudo single-phase mixture to finish fluid simulation, in which continuous governing equations are solved to obtain nanofluid flow and heat transfer performance in real time. At this step, nanofluid non-uniform volume fraction is estimated according to the particles distribution in the whole computational region. Nanofluid properties can be obtained and updated according to the information such as nanofluid volume fraction, temperature distribution and nanoparticles size features, etc. Secondly, nanofluid is treated as a real two-phase suspension. Based on the details such as environmental flow and collisions between particles (or particles and boundaries), nanoparticles will be accelerated or decelerated to new positions. These new positions are recorded to estimate nanofluid volume fraction and get ready to update nanofluid properties for next time step. Generally, the whole procedure for a single time step simulation can be summarised as in Fig. 4.3.

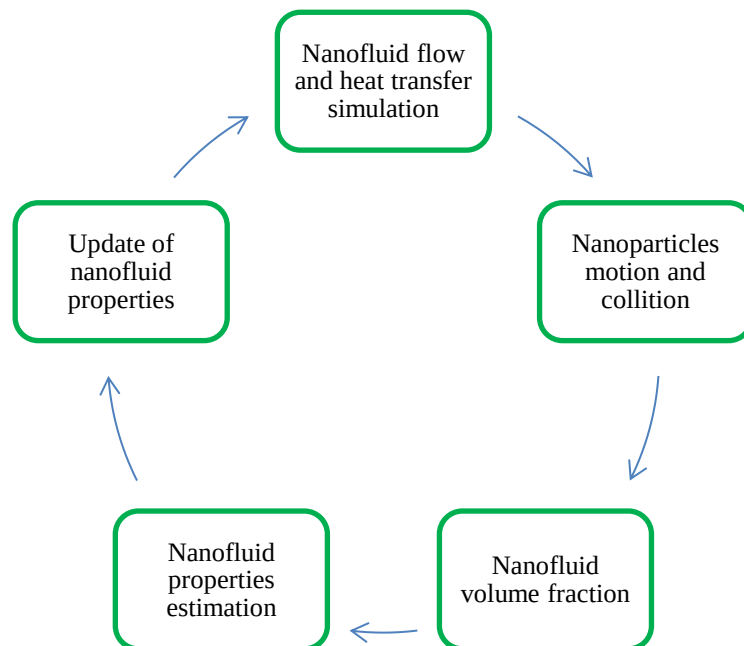


Fig. 4.3 Basic idea of solver ‘nanofluidELFoam’

In terms of nanofluid properties estimation in this thesis, nanofluid thermal conductivity k_{nf} and viscosity μ_{nf} are collected from the experimental investigations reported in (Das et al., 2003b) with regression analysis rather than being obtained from estimation models. More specifically, parameters k_{nf} and viscosity μ_{nf} for Al_2O_3 /water nanofluids in the following numerical works are considered to be related to volume fraction, basefluid properties and temperature, which are the three most important parameters for nanofluid properties estimation (given that the solid phase is Al_2O_3). They are presented by:

$$k_{nf} = k_f[0.0027(T - 273) + 0.972](2.44\phi + 1) \quad (4.38)$$

$$\mu_{nf} = \mu_f[0.0002(T - 273)^2 - 0.0305(T - 273) + 1.52](9.23\phi + 1) \quad (4.39)$$

Nanofluid density ρ_{nf} and heat capacity $c_{p_{nf}}$ are estimated by average model (Khanafar et al., 2003, Oztop and Abu-Nada, 2008):

$$\rho_{nf} = (1 - \phi)\rho_f + \phi\rho_s \quad (4.40)$$

$$\rho_{nf}c_{p_{nf}} = (1 - \phi)\rho_fc_{p_f} + \phi\rho_sc_{p_s} \quad (4.41)$$

To enable above estimations in the whole computational region, nanofluid volume fraction for each numerical cell should be calculated beforehand. For this, a function is designed as:

$$\phi_i = \frac{R_i}{N} \frac{V}{V_i} \phi \quad (4.42)$$

where R_i is real particles number in cell i , N is the total number of nanoparticles in computational region, V_i is volume of cell i and V is total volume of the whole computational region. ϕ is the given nanofluid volume fraction if assumed that nanofluid is homogeneous.

From (4.42), it is easy to see that $\phi_i = 0$ can be assured if there is no particle in a certain cell i , indicating that the physical properties of this cell are the same as the basefluid. Furthermore, it also can be proved that $\sum_{i=1}^n \phi_i V_i$ equals to ϕV , indicating that the sum of nanoparticles effect in each Eulerian cell equals to the situation that the nanofluid is homogenous with volume fraction ϕ .

Furthermore in this approach, a subroutine is necessary to decide whether a simulation using

above strategy is trustable. For instance, if an extreme case existed, in which the cells number n_{cell} is much larger than numerical particles number n_p (as shown in Fig. 4.4). It means many cells (maybe) have not even one particle and they will have the same physical properties as the basefluid, while those cells with only a few particles would have obvious different properties from their neighbour cells. This will lead to un-trustable results for sure.

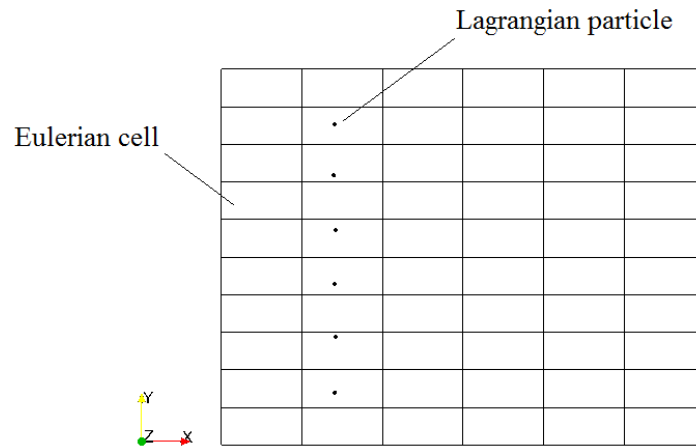


Fig. 4.4 An extreme case with very few numerical particles in computational region

To combine the functions of solving fluid flow and heat transfer performance in original solver ‘buoyantBoussinesqPimpleFoam’ and the functions of particles tracking in original solver ‘icoUncoupledKinematicParcelFoam’, the best option is to maintain the main structure of one basic solver and develop the new solver based on it. Comparatively, adding particles tracking function to original solver ‘buoyantBoussinesqPimpleFoam’ is a better option due to the programming work is slightly easier. For this, one of the most important steps is to add a program section:

```
argList::addOption
(
    "cloudName",
    "name",
    "specify alternative cloud name. default is 'kinematicCloud'"
);
laminarTransport.correct();
mu = laminarTransport.nu()*rhoInfValue;
kinematicCloud.evolve();
```

to solver ‘buoyantBoussinesqPimpleFoam’. All the information of particles motion and tracking are included in the module ‘kinematicCloud.evolve()’. In original source file

'buoyantBoussinesqPimpleFoam.C', the newly added coding section should be located after the N-S equations and energy equation have been solved, but before the final statements of solved results.

So far, the impacts of continuous fluid can be coupled to small particles, however, the impacts of particles to environmental fluid are not considered yet. To make the new solver is available for 'four-way' coupling (Tab. 4.2), a source term, i.e. the product of (4.5) should be added to the original N-S equation (3.15) to present the collective effects from solid particles to continuous fluid. Furthermore, there is a suggestion that the four-way coupling should be performed when volume fraction ϕ of the secondary phase is greater than 0.001 (Crowe, 2005). Regarding nanofluid, four-way coupling is believed to be the most practical method because most nanofluid volume fractions are greater than $\phi = 0.001$.

Tab. 4.2 Interaction considerations in different coupling approaches

Coupling approach	Considerations
one-way	fluid→particle
two-way	fluid↔particle
four-way	fluid↔particle particle↔particle

4.5 Solver test

To test the newly developed solver 'nanofluidELFoam', a numerical work published by (Pallares and Grau, 2010) is repeated. In this case, a two-dimensional square cavity filling with $\phi = 0.03$ $\text{Al}_2\text{O}_3/\text{water}$ nanofluid is considered. The length and height of this cavity geometry are both $L = 0.1\text{m}$, with temperature difference $\Delta T = 1\text{K}$ between the heating and cooling walls at left and right sides respectively (Fig. 4.5a), i.e. $T_H = 300.5\text{K}$, $T_H = 299.5\text{K}$ and $T_{Ref} = 300\text{K}$. The temperature driven flow is laminar and Rayleigh number is $Ra = 1.4 \times 10^7$. For this test, uniform mesh strategy 50×50 is selected for continuous fluid simulations as well as the Eulerian 'frame' to record particles distribution (Fig. 4.5b).

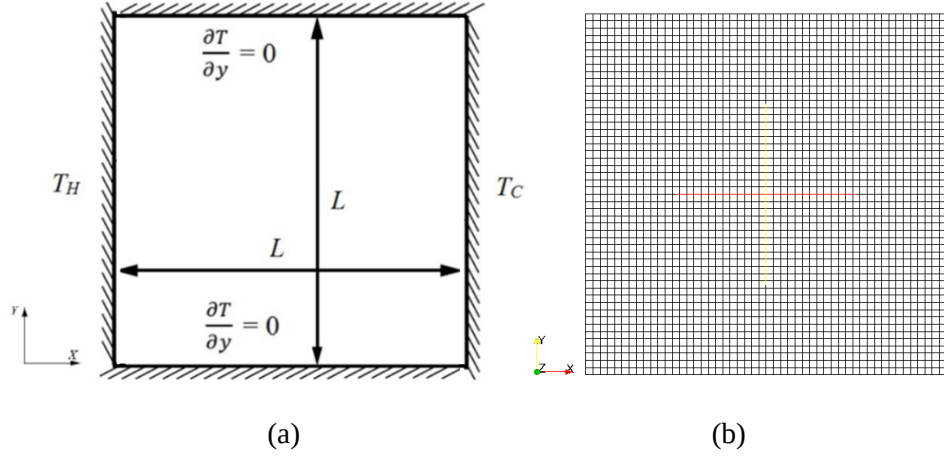


Fig. 4.5 The geometry and uniform mesh strategy with 50×50 cells

According to (Pallares and Grau, 2010), $n = 40000$ numerical particles are generated and distributed randomly in the 2500 Eulerian cells at $t = 0s$ to present practical nanoparticles (Fig. 4.6). However, building on the basic setup in (Pallares and Grau, 2010), drag force F_D , buoyancy and gravity force F_{BG} , thermophoretic force F_T , Brownian force F_B , Saffman force F_S , pressure gradient force F_P , virtual mass induced force F_V and Magnus force F_M are also considered for each numerical particle. The estimation approaches and the values for those necessary constants were introduced in Section 4.3.1. In terms of contacting forces, the estimation approaches and the values for those necessary constants were introduced in Section 4.3.2. Young's modulus and Poisson's ratio are still used as $E = 400 \text{ GPa}$ and $\nu_p = 0.25$, respectively for calculation (Auerkari, 1996).

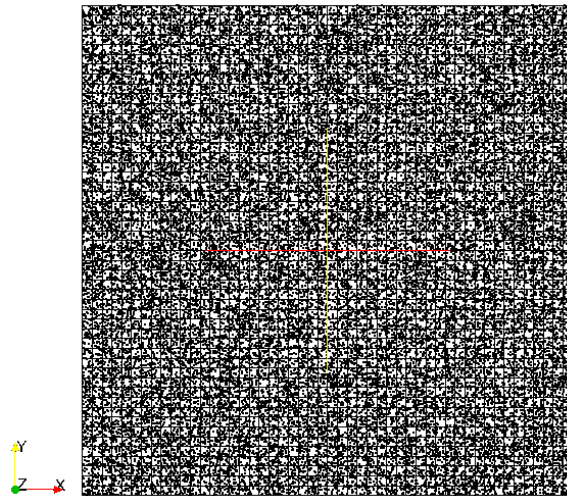


Fig. 4.6 The initialized 40000 particles in cavity at $t = 0s$

Properties of water and Al_2O_3 particles can be found in Tab. 4.3 (Pallares and Grau, 2010).

Tab. 4.3 Water fluid and Al_2O_3 particle properties

	ρ (kg/m^3)	k (W/mK)	C_p (J/kg K)	ν (m^2/s)	β ($1/\text{K}$)
Water	997	0.605	4179	$8.9\text{e-}7$	$2.75\text{e-}4$
Al_2O_3	3600	46	765	$\sim$	$6.3\text{e-}6$

To ensure the particles motions can be reasonably captured, time step in simulation is controlled as $\Delta t = 10^{-5}\text{s}$. The simulation is carried out to $t = 600\text{s}$ (stable results can be found at about $t = 400\text{s}$ in this case). With the help of paraView 4.0.1 (Henderson et al., 2004), temperature driven flow can be found in this two-dimensional cavity due to the temperature difference between heating and cooling walls, indicating that the Boussinesq assumption is functional in solver ‘nanofluidELFoam’. However, due to the further surface, body and contacting forces considered in this solver, the temperature features (Fig. 4.7a) are found notably different from the results predicted by original solver ‘buoyantBoussinesqPimpleFoam’ using single phase approach (Fig. 4.7b). This deviation was also reported in (Pallares and Grau, 2010) but with no further discussions.

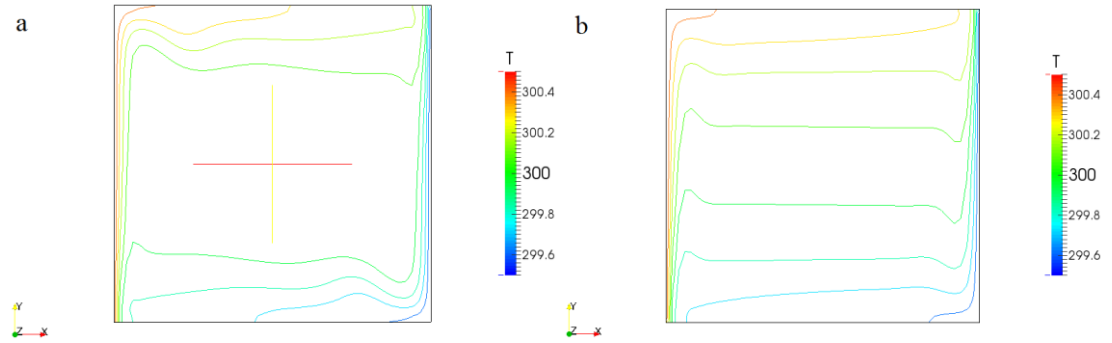


Fig. 4.7 Temperature features with (a) and without (b) colliding particles

Actually, if looked at the particles in computational region, it can be observed that the distribution of 40000 particles are considerably changed to a strange manner by the temperature driven flow and the further considered forces in the newly developed solver (Fig. 4.8). However, if a statistics was carried out to measure how many particles in each cell at $t = 600\text{s}$, it can be found that the distribution still roughly conforms to Poisson distribution curve (Fig. 4.9) (Haight and Haight, 1967). This observation is same as the conclusion in (Pallares and Grau, 2010). As discussed in (He et al., 2009), the formats of most of forces acting on particles given in Section 4.3.1 are established for relatively large particles and may not be applicable to nanoparticles due to for example the rarefaction and it is not clear currently how these expressions are corrected for

nanoparticles. Further work is still needed on this aspect.

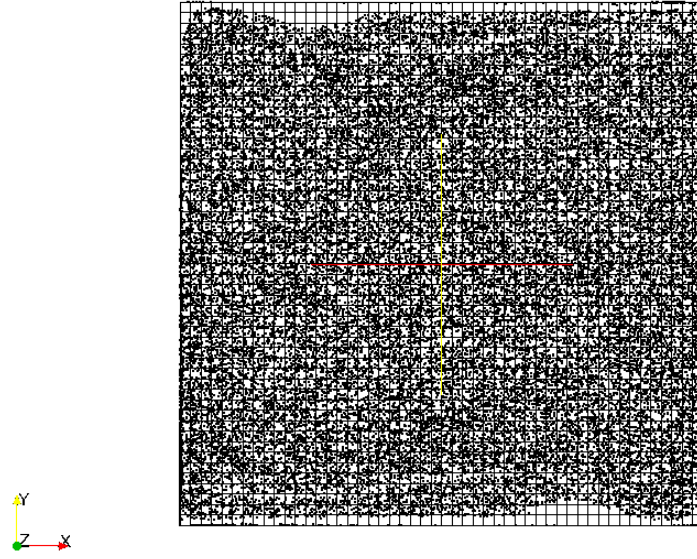


Fig. 4.8 Particles distribution is changed by temperature driven flow

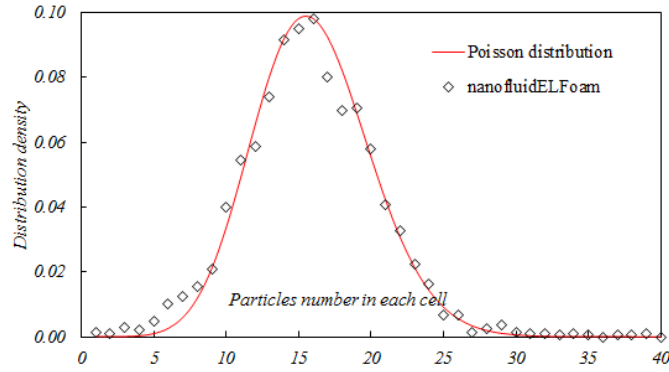


Fig. 4.9 The comparison between theoretical Poisson distribution and particles distribution predicted by 'nanofluidELFoam'

Assuming the present forces models are reliable (He et al., 2009), another phenomenon which was not mentioned in (Pallares and Grau, 2010) also can be found in Fig. 4.8, that is some Eulerian cells apparently have no numerical particles in the regions close to upper and bottom walls due to the non-uniform particles distribution features in Fig. 4.8, while particles in some cells near heating and cooling walls are considerably concentrated because the continuous fluid flow velocities in such areas are comparatively small. Due to this, those cells with more particles will definitely have greater volume fractions ϕ_i according to (4.42) (as shown in Fig. 4.10). Consequently, when nanofluid properties are estimated based on regional nanoparticle concentrations, non-uniform features over the whole computational region can be obtained as in

Fig. 4.11. In (Pallares and Grau, 2010), such information was ignored.

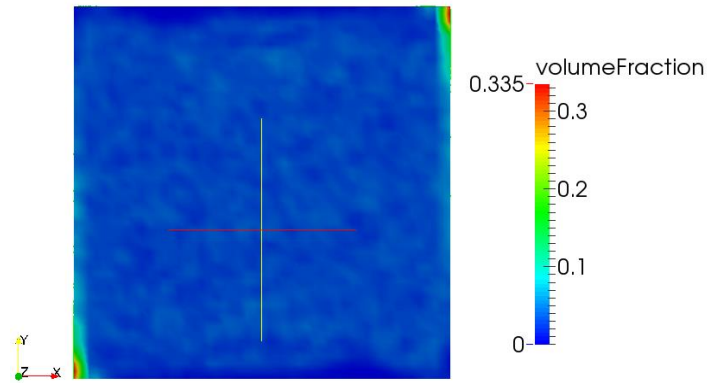


Fig. 4.10 Non-uniform particles distribution

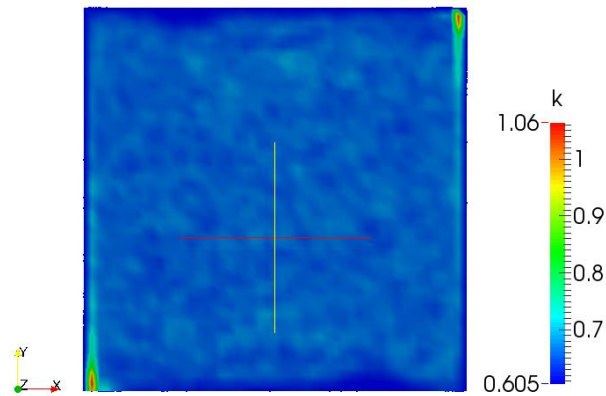


Fig. 4.11 Non-uniform nanofluid thermal conductivity due to non-uniform particles distribution

By this test, it can be verified that the newly developed solver ‘nanofluidELFoam’ works fairly well to combine fluid flow, heat transfer and nanoparticles tracking together. During the simulation, non-uniform nanofluid properties can be estimated according to non-uniform particles distribution. But the problem in this test is, the particles number is $n = 40000$ only. On a computer with configuration of Intel (R) Core (TM) i5-4690 CPU @ 3.50GHz (2 CPUs parallel running), about 10 hours are needed to finish the simulation to $t = 600s$. When particles number is increased to $n = 400000$, the current computer seems cannot afford the simulation during a reasonable period. In present simulations, the parcel concept is applied to avoid tracking all those real nanoparticles in this case. According to (4.3) and (4.4), if a numerical particle (parcel) number is used as $n = 40000$, it means one numerical parcel should contain nearly two billion real nanoparticles. This will induce an unreliable simulation very likely. Therefore, although solver ‘nanofluidELFoam’ is believed to be theoretically practical, however in practice, it is not recommended when computational resource is limited.

4.6 Conclusions

In this chapter, some key points of DEM simulation for nanoparticles tracking are discussed, such as parcel treatment, forces estimation and particles initialization, etc. Building on what, a new OpenFOAM solver ‘nanofluidELFoam’ is developed to combine the functions of simulating continuous fluid flow and heat transfer (‘buoyantBoussinesqPimpleFoam’) and the function of tracking dispersed particles (‘icoUncoupledKinematicParcelFoam’).

Based on a two-dimensional cavity testing case, it can be found that the newly developed solver ‘nanofluidELFoam’ is able to perform nanofluid simulation under Eulerian-Lagrangian frame. By tracking $n = 40000$ Al_2O_3 nanoparticles in water, it can be observed that nanofluid volume fraction is not uniform over the whole computational region due to non-uniform distribution of nanoparticles. This phenomenon will consequently induce non-uniform nanofluid properties, which will have strong impacts on nanofluid flow and heat transfer performance.

In terms of computational efficiency, however, the required computational time is found to be a problem when nanoparticles number is increased from $n = 40000$ to $n = 400000$ and collision mechanisms are still considered. Actually, even when the nanoparticles number has been increased to $n = 400000$, it is still not big enough to present real nanoparticles in a computational region with non-uniform grid strategies. Moreover, when a numerical parcel contains too many nanoparticles, the reliability of the simulation is becoming to be doubtful. Therefore, Eulerian-Lagrangian solver ‘nanofluidELFoam’ is not recommended when computational resource is limited.

5. EULERIAN-MIXTURE APPROACH FOR NANOFLUID SIMULATION WITH OpenFOAM

5.1 Introduction

By previous test for the Eulerian-Lagrangian solver ‘nanofluidELFoam’, the issue of huge computational resource requirement still exists when too many nanoparticles need to be tracked. Moreover, when mesh is refined in near wall regions, even a huge particle number still cannot ensure all Eulerian cells have computational particles. Comparatively, CFD simulations under Eulerian frame seems to be a better option for most nanofluid cases at present.

As discussed in Section 2.4, Mixture approach under Eulerian frame is believed to be the best choice to achieve the balance between CFD simulation accuracy and computational effort. In this chapter, the basic idea of Eulerian-Mixture approach is discussed. Based on what, a new OpenFOAM solver ‘nanofluidMixtureFoam’ is developed and tested for the coupling of nanofluid flow, heat transfer and nanoparticles sedimentation. More specifically, for ‘nanofluidMixtureFoam’, the basic ideas of solving continuous fluid flow and heat transfer in ‘buoyantBoussinesqPimpleFoam’ and the basic idea of simulating dispersed phase settling in ‘driftFluxFoam’ are combined (Tab. 5.1).

Tab. 5.1 Solvers features

Solver	Fluid flow	Fluid heat transfer	Particles sedimentation consideration
buoyantBoussinesqPimpleFoam	Yes	Yes	No
driftFluxFoam	Yes	No	Yes
nanofluidMixtureFoam	Yes		

5.2 Mixture implementations in OpenFOAM

5.2.1 Mixture continuity equation

By assuming multi-phase flow as a pseudo multi-phase mixture, only one set of governing equations is needed in solver ‘driftFluxFoam’. They are continuity equation and momentum equation for mixture and a continuity equation for dispersed phase. Theoretically, continuity and momentum equations can be derived from Eulerian-Eulerian model (Brennan, 2001). Considering

a two-phase flow, one phase is continuous basefluid and the other phase consists of dispersed solid particles. A continuity equation is required for each of the two phases:

$$\frac{\partial(\phi_f \rho_f)}{\partial t} + \nabla \cdot (\phi_f \rho_f U_f) = 0 \quad (5.1)$$

$$\frac{\partial(\phi_s \rho_s)}{\partial t} + \nabla \cdot (\phi_s \rho_s U_s) = 0 \quad (5.2)$$

where ρ_f and ρ_s are the densities of continuous phase and dispersed solid phase, respectively. ϕ_f and ϕ_s are the volume fractions of continuous phase and dispersed solid phase, respectively. U_f and U_s are the velocities of continuous phase and dispersed solid phase, respectively.

If added (5.1) to (5.2), the result can be written as:

$$\frac{\partial(\phi_f \rho_f + \phi_s \rho_s)}{\partial t} + \nabla \cdot (\phi_f \rho_f U_f + \phi_s \rho_s U_s) = 0 \quad (5.3)$$

For the two-phase mixture, key properties and flow features can be estimated using (Ishii and Grolmes, 1975):

$$\rho_m = \phi_f \rho_f + \phi_s \rho_s \quad (5.4)$$

$$U_f = U_{fm} + U_m \quad (5.5)$$

$$U_s = U_{sm} + U_m \quad (5.6)$$

$$\phi_f \rho_f U_{fm} + \phi_s \rho_s U_{sm} = 0 \quad (5.7)$$

where U_{fm} and U_{sm} are relative velocities of continuous phase and dispersed solid phase to the mixture, respectively. U_m is the velocity of the mixture.

Then the contents in the second bracket of (5.3) can be rewritten as:

$$\phi_f \rho_f U_f + \phi_s \rho_s U_s = \rho_m U_m \quad (5.8)$$

Therefore, (5.3), the continuity equation for the two phases can be written in a very similar form as that for a normal single phase flow:

$$\frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m U_m) = 0 \quad (5.9)$$

In solver ‘driftFluxFoam’, (5.9) is not used directly in any header files. However, it will be

used implicitly in file ‘*pEqn.H*’ for pressure-velocity correction. This procedure was introduced in Section 3.3.

5.2.2 Mixture momentum equation

Momentum equations for continuous and dispersed solid phases can be given as:

$$\frac{\partial \phi_f \rho_f U_f}{\partial t} + \nabla \cdot (\phi_f \rho_f U_f U_f) = -\nabla(\phi_f p_f) + \nabla \cdot [\phi_f (\tau_f + \tau_f^t)] + \phi_f \rho_f g \quad (5.10)$$

$$\frac{\partial \phi_s \rho_s U_s}{\partial t} + \nabla \cdot (\phi_s \rho_s U_s U_s) = -\nabla(\phi_s p_s) + \nabla \cdot [\phi_s (\tau_s + \tau_s^t)] + \phi_s \rho_s g \quad (5.11)$$

Adding (5.10) to (5.11), the result can be obtained as:

$$\frac{\partial (\sum \phi_k \rho_k U_k)}{\partial t} + \nabla \cdot (\sum \phi_k \rho_k U_k U_k) = -\nabla(\sum \phi_k p_k) + \nabla \cdot [\sum \phi_k (\tau_k + \tau_k^t)] + \sum \phi_k \rho_k g \quad (5.12)$$

where k presents the k th phase in mixture, it could be $k = f$ for continuous fluid phase (basefluid) and $k = s$ for dispersed solid phase (nanoparticles).

Eventually, the momentum equation (5.12) for mixture can be given as:

$$\frac{\partial \rho_m U_m}{\partial t} + \nabla \cdot (\rho_m U_m U_m) = -\nabla p_m + \nabla \cdot [\tau_m + \tau_m^t - \sum \phi_k \rho_k U_{km} U_{km}] + \rho_m g \quad (5.13)$$

According to (5.7), the relative velocity U_{fm} between fluid (continuous phase) and mixture can be expressed as:

$$U_{fm} = -\frac{\phi_s \rho_s}{\phi_f \rho_f} U_{sm} \quad (5.14)$$

With this consideration, the relative velocity between solid particle (dispersed phase) and mixture, U_{sm} should be defined before solving the mixture momentum equation (5.13).

For original solver ‘*driftFluxFoam*’, two models are officially provided for the calculation of U_{sm} , named as ‘*simple*’ and ‘*general*’, respectively. They are predefined and compiled in folder ‘*relativeVelocityModel*’. In OpenFOAM tutorials, the ‘*simple*’ one is usually set as default. This model was proposed by (Vesilind, 1968) and can be expressed as:

$$U_{sm} = \frac{\rho_f}{\rho_m} U_0 10^{-A \phi_s} \quad (5.15)$$

where U_0 and A are settlement velocity and settlement coefficient, respectively.

The ‘general’ one was proposed by (Takács et al., 1991) and can be expressed as:

$$U_{sm} = \frac{\rho_f}{\rho_m} U_0 [e^{-A \max(\phi_s - \phi_r, 0)} - e^{-A_1 \max(\phi_s - \phi_r, 0)}] \quad (5.16)$$

where A and A_1 are settling parameters, while ϕ_r is residual volume fraction (a reference parameter). If (5.16) was applied, the sedimentation phenomenon will stop once the volume fraction of mixture reduces to a given threshold value ϕ_r .

According to the experimental observations reported in (Wen et al., 2009), the ‘simple’ model (5.15) is believed to be more practical for sedimentation simulations of those nanofluids without stabilising treatments. Thus this model will be employed in following simulations. However in some cases, the ‘general’ model presented in (5.16) could be more practical because a certain amount of nanoparticles maybe suspend in basefluid for a rather long period due to Brownian motion and mutual collisions.

5.2.3 Continuity equation for the dispersed phase

The continuity equation for the dispersed phase has been given by (5.2), but together with (5.6), it can be rewritten as:

$$\frac{\partial(\phi_s \rho_s)}{\partial t} + \nabla \cdot [\phi_s \rho_s (U_m + U_{sm})] = 0 \quad (5.17)$$

After opening the bracket of dispersed phase velocity term $(U_m + U_{sm})$, the continuity equation for the dispersed phase can be expanded as:

$$\frac{\partial(\phi_s \rho_s)}{\partial t} + \nabla \cdot (\phi_s \rho_s U_m) + \nabla \cdot (\phi_s \rho_s U_{sm}) = 0 \quad (5.18)$$

In (5.18), mixture velocity U_m can be solved by momentum equation (5.13), while U_{sm} can be estimated by (5.15) or (5.16). Thus ϕ_s , the volume fraction of dispersed phase (solid particle), is the only unknown variable in (5.18). In solver ‘driftFluxFoam’, continuity equation for the dispersed phase is defined in header file ‘*alphaEqn.H*’.

After going through the governing equations of Mixture model, it can be inferred that the basic idea of Mixture approach should be suitable for nanofluid CFD simulation with consideration of nanoparticles sedimentation. The reasons can be summarised as follows:

1. Compared to Eulerian-Lagrangian approach, Mixture approach is not to track every single

particle in computational region. Therefore it is possible to deal with the cases in which too many computational cells are needed.

2. Compared to standard Eulerian-Eulerian approach, Mixture approach is not to solve two sets of governing equations for continuous phase and dispersed phase separately. This reduces lots of computational effort.
3. Compared to VOF approach, Mixture approach is not to track the interface between different phases. This reduces computational effort even further.

Due to above reasons, Mixture approach and original solver ‘driftFluxFoam’ are employed in present work as the basis to simulate nanofluid flow and heat transfer performance combining with nanoparticles sedimentation. Building on ‘driftFluxFoam’, a new OpenFOAM solver ‘nanofluidMixtureFoam’ will be developed particularly for nanofluid CFD simulations in the following section.

5.2.4 Solver ‘driftFluxFoam’ and test

In OpenFOAM, solver ‘driftFluxFoam’ is designed for two incompressible fluids using Mixture approach with drift-flux approximation for relative motion between phases. It is a newly released solver since OpenFOAM version 2.3.0. The basic idea of ‘driftFluxFoam’ is to consider two-phase flow as a mixture, the properties of which can be obtained by using appropriate models such as (4.38)~(4.41). In ‘driftFluxFoam’, the governing equations were introduced in Sections 5.2.1~5.2.3, and the main process can be given in a simplified flow chart shown in Tab. 5.1.

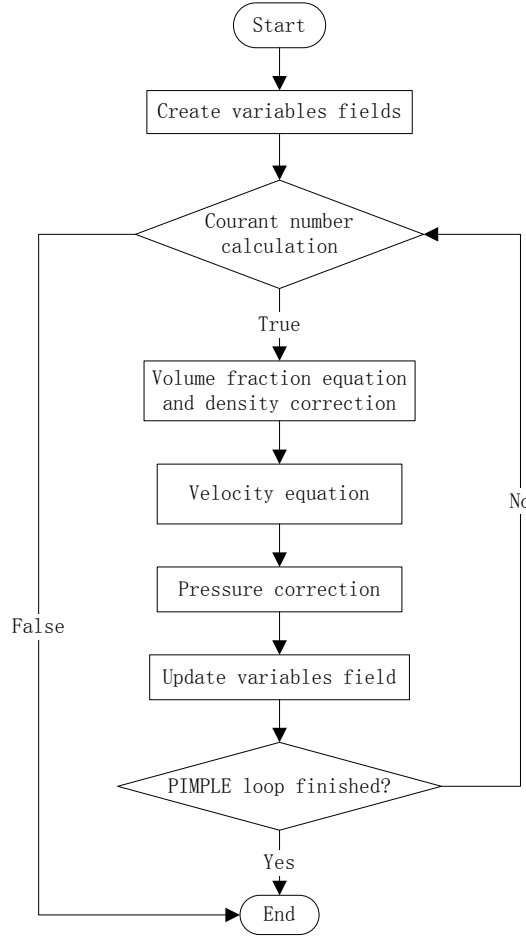


Fig. 5.1 The flow chart of solver 'driftFluxFoam'

A nanofluid case in two-dimensional square cavity ($L = 0.04m$) is used to observe the performance of original OpenFOAM solver 'driftFluxFoam' for sedimentation. The numerical results are compared to a sedimentation observation of 2.5wt% ($\phi = 0.64\%$) Al_2O_3 /water nanofluid reported by (Wen et al., 2009). To reduce computational resource requirement of this case, a two-dimensional cavity enclosure filled with 0.64% Al_2O_3 /water nanofluid is used instead of the three-dimensional vessel in the experimental observation. However, the numerical simulation is still considered to be practical because the wall impacts on any vertical sections are exactly same to the suspension in the vertically-standing cylindrical vessel.

To enhance simulation accuracy for near wall regions, non-uniform strategy is used to refine the mesh near walls as shown in Fig. 5.2. This is achieved by 'Bump' function in Gmsh and grid ratio is set as $\delta = 0.06$. Mesh strategies with cell amounts 30×30 , 40×40 , 50×50 and 60×60 are generated for grid independence check. Applying the same sedimentation setups as in the official OpenFOAM tutorial case 'dahl', it can be found that mesh strategies 50×50 and 60×60 give

nearly identical results in terms of sedimentation layer height (Fig. 5.3), indicating that mesh strategy 50×50 is fine enough to obtain trustable results in this sedimentation case.

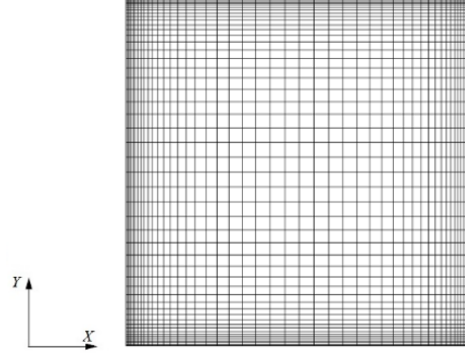


Fig. 5.2 Mesh strategy of the two-dimensional square enclosure

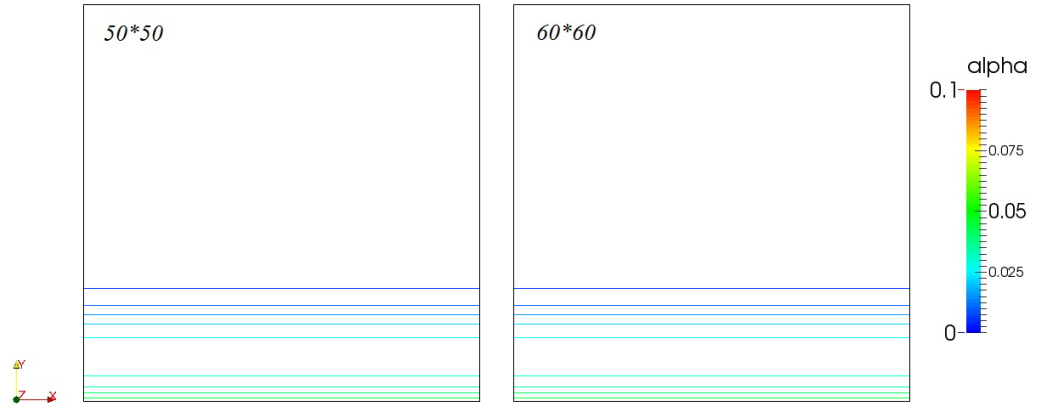


Fig. 5.3 The sedimentation layer heights at $t = 7 \text{ hours}$ given by strategies 50×50 and 60×60 in grid independence check

Furthermore, after many numerical tests and with the help of enGauge (Mitchell, 2002), sedimentation parameters $U_0 = -5 \times 10^{-6} \text{ m/s}$ and $A = 80$ are found to be appropriate for (5.15) in present work. When the simulation is carried out for seven hours until the stable sedimentation can be observed (Fig. 5.4), it can be measured and confirmed that original OpenFOAM solver ‘driftFluxFoam’ is functional to simulate nanoparticles sedimentation behaviour (Ghadimi et al., 2011). It also can be inferred that, if combined with some other functions such as heat transfer simulation and non-uniform nanofluid properties estimation, a developed version of ‘driftFluxFoam’ could be appropriate for nanofluid flow and heat transfer simulations combining with nanoparticles sedimentation.

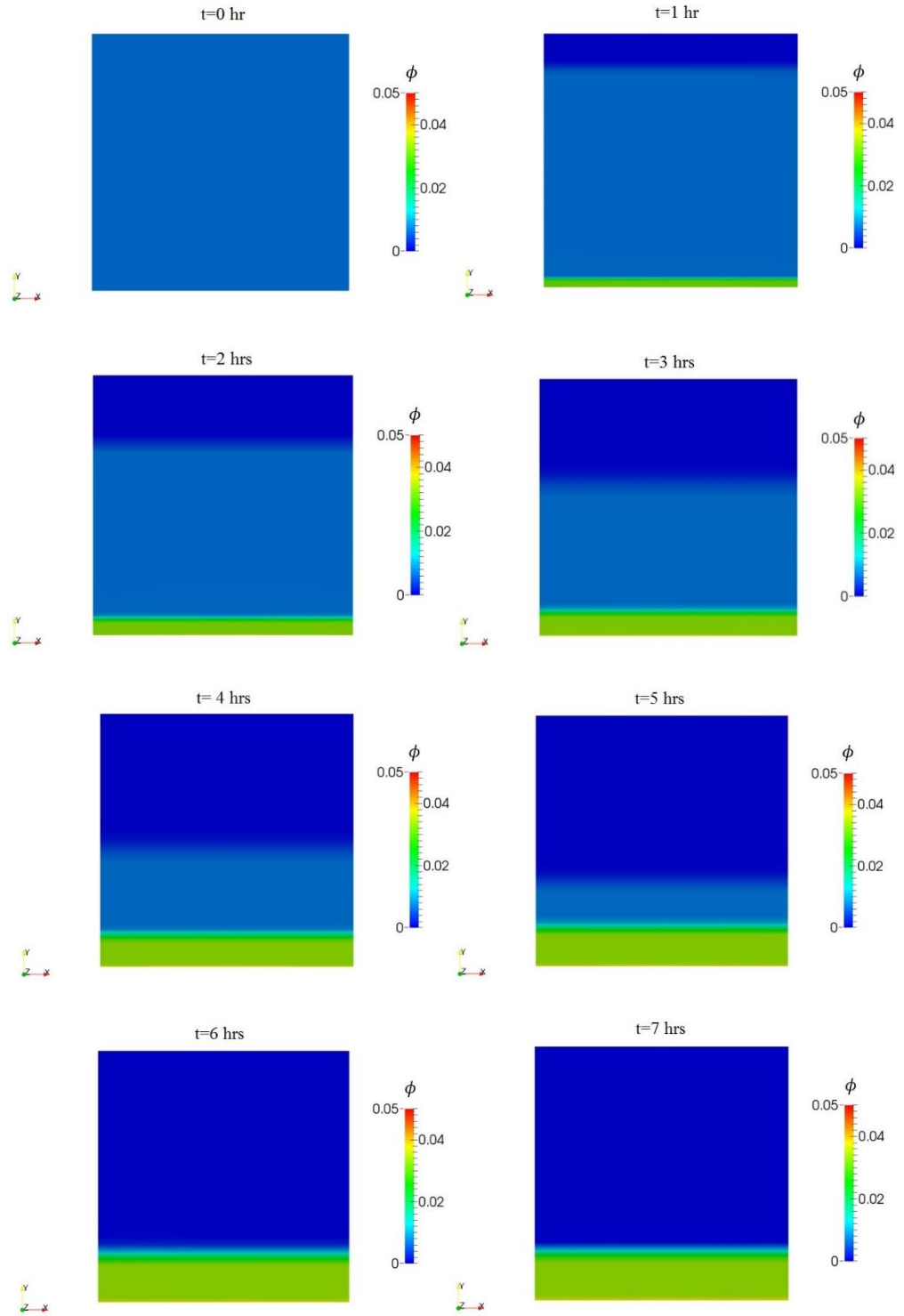


Fig. 5.4 The sedimentation simulation of 0.64% Al_2O_3 /water nanofluid in a two-dimensional cavity by solver 'driftFluxFoam'

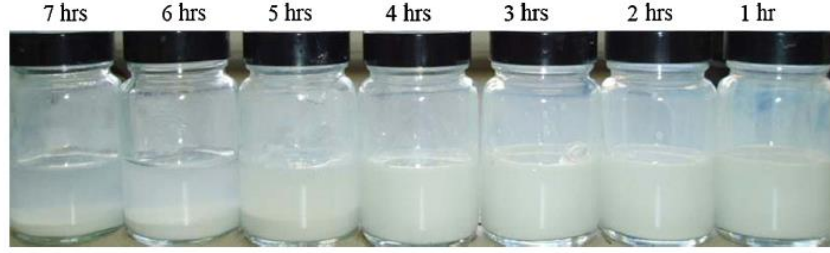


Fig. 5.5 Stability of 0.64% Al_2O_3 /water nanofluids (without any stabilizer) changing with time (Wen et al., 2009)

5.3 Development of ‘nanofluidMixtureFoam’

5.3.1 Basic idea

Compared to original solver ‘driftFluxFoam’, two mainly developments will be made for the new solver ‘nanofluidMixtureFoam’ as follows:

1. Energy (temperature) equation will be added to ensure the solver is able to carry out heat transfer simulation.
2. Nanofluid properties (e.g. density, viscosity and heat capacity etc) will be estimated at each time step according to non-uniform nanoparticles concentration (caused by nanoparticles sedimentation).

Therefore, for solver ‘nanofluidMixtureFoam’, nanofluid properties such as thermal conductivity, density and viscosity are initialised separately at the very beginning of simulation. This mechanism will displace the original one by which fluid properties are defined simply by the values in file ‘*transportProperties*’ of the case in consideration. By the new mechanism in ‘nanofluidMixtureFoam’, the values of density ρ_{nf} , dynamic viscosity μ_{nf} , thermal conductivity k_{nf} and heat capacity $c_{p_{nf}}$ for each numerical cell can be updated separately according to the information such as nanoparticles volume fraction and environmental temperature.

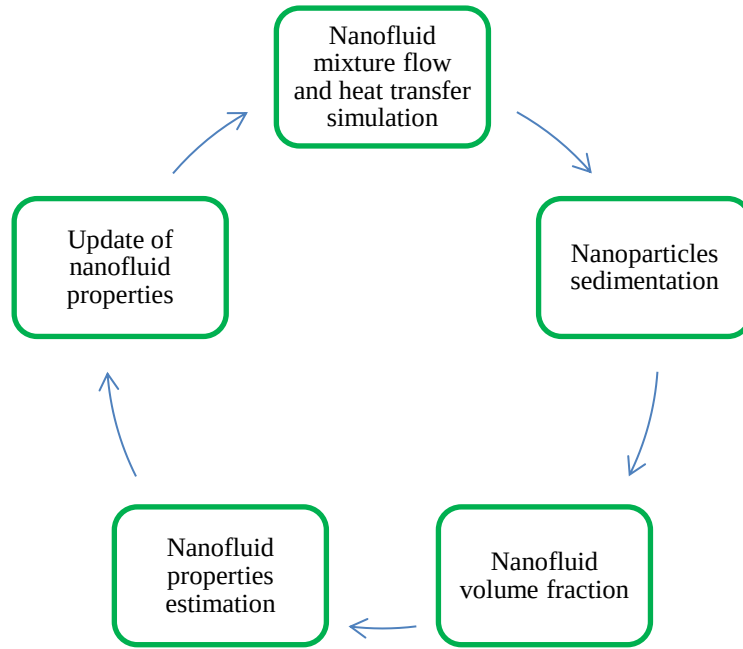


Fig. 5.6 Basic idea of solver ‘nanofluidMixtureFoam’

In a single simulation cycle of solver ‘nanofluidMixtureFoam’ (Fig. 5.6), nanofluid is considered as single-phase first, the flow and heat transfer performances are simulated in a similar way as solver ‘buoyantBoussinesqPimpleFoam’. However, when nanoparticles sedimentation features are predicted, nanofluid is considered as two-phase and the volume fraction ϕ of the dispersed phase will be solved for each numerical cell. Based on the new information such as nanoparticles volume fraction and nanofluid temperature, nanofluid properties are to estimated and updated throughout the whole computational domain according to pre-defined models such as (4.38)~(4.41) (see in Section 5.3.2). The newly solved fields will be ready for the simulation in next time step.

5.3.2 Solver development in OpenFOAM

To develop the new solver ‘nanofluidMixtureFoam’, original solver ‘driftFluxFoam’ is used as the basis. Energy equation (3.21) will be added to solve energy transfer between numerical cells due to flow and temperature difference. According to the working procedure of ‘buoyantBoussinesqPimpleFoam’, this function will be implemented after momentum equation is solved for each time step but before the velocity-pressure corrections for ‘PIMPLE’ loop. The program section can be given as follows:

```

while (pimple.loop())
{
    #include "alphaControls.H"
    UdmModel.correct();
    #include "alphaEqnSubCycle.H"
    mixture.correct();
    #include "UEqn.H"
    #include "TEqn.H" //Newly added energy equation
    // --- Pressure corrector loop
    while (pimple.correct())
    {
        #include "pEqn.H"
    }
}

```

At each time step, the velocity of nanoparticles sedimentation is estimated by function ‘UdmModel.correct()’. By continuity equation of the dispersed phase (5.18), each cell in computational region will have a volume fraction value ϕ_s . Together with the field of current temperature T_m , nanofluid properties can be updated according to (4.38)~(4.41) as given in Section 4.4.

To illustrate the main structure of ‘nanofluidMixtureFoam’, a simplified flow chart can be given as in Fig. 5.7. By such a simulation strategy, the newly developed solver ‘nanofluidMixtureFoam’ will be suitable for nanofluid CFD simulation due to its ability of solving nanofluid flow, heat transfer and nanoparticles sedimentation phenomena together. However, before applying this solver to typical nanofluid simulations, it will be tested first in the following section.

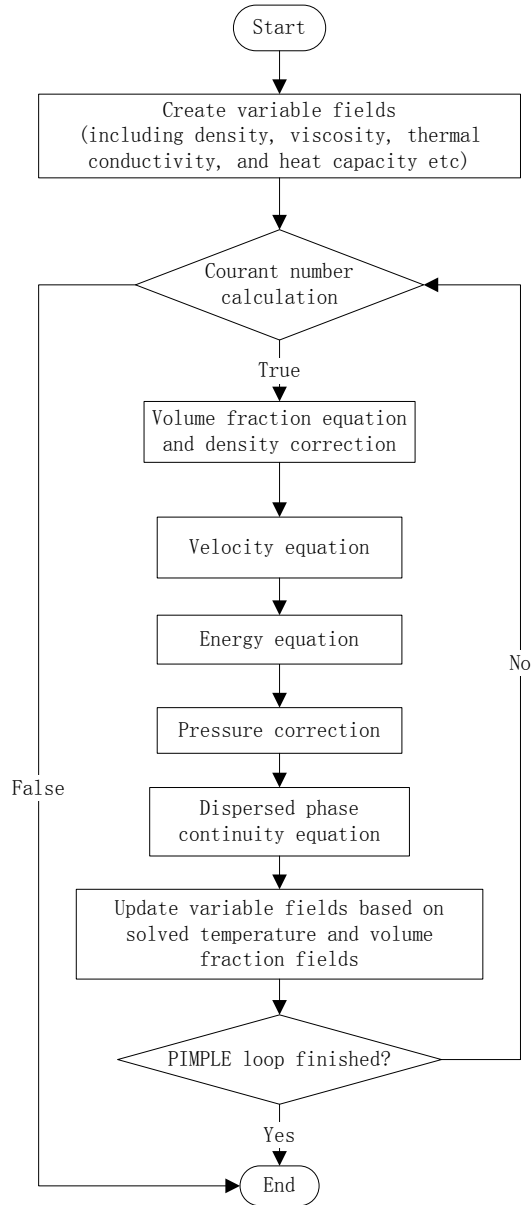


Fig. 5.7 The flow chart of solver 'nanofluidMixtureFoam'

5.4 Solver test

To test the newly developed solver 'nanofluidMixtureFoam', a case of laminar air natural convection in a two-dimensional square cavity is performed. The reason to use such a case for testing is the newly developed solver was developed from original OpenFOAM solver 'driftFluxFoam', it should, apparently maintain the functions for particles sedimentation (which has been presented in Section 5.2.4). However, the newly added energy equation should be tested to confirm that 'nanofluidMixtureFoam' is also available for heat transfer. The simulation results will be compared to previous numerical study reported by (Barakos et al., 1994) and the results

predicted by solver ‘buoyantBoussinesqPimpleFoam’. This is to ensure the new solver ‘nanofluidMixtureFoam’ is functional for natural convection simulation.

In this test, the square cavity size is $L = 0.0168m$ and filled with air (Fig. 5.8a). The top and bottom walls are insulated, while the temperatures at left and right walls are controlled as $T_H = 303K$ and $T_C = 283K$ respectively for a given Rayleigh number $Ra = 1 \times 10^4$. According to the grid independence check in (Barakos et al., 1994), uniform structured mesh strategy 80×80 is employed for both of the two cases (Fig. 5.8b). Key parameters for the simulation can be found in Tab. 5.2. Dimensionless velocity u^* , v^* and temperature T^* are employed and compared to the numerical results predicted by original OpenFOAM solver ‘buoyantBoussinesqPimpleFoam’ and the numerical simulations reported by (Barakos et al., 1994). The definitions of them can be found in (5.19)~(5.21).

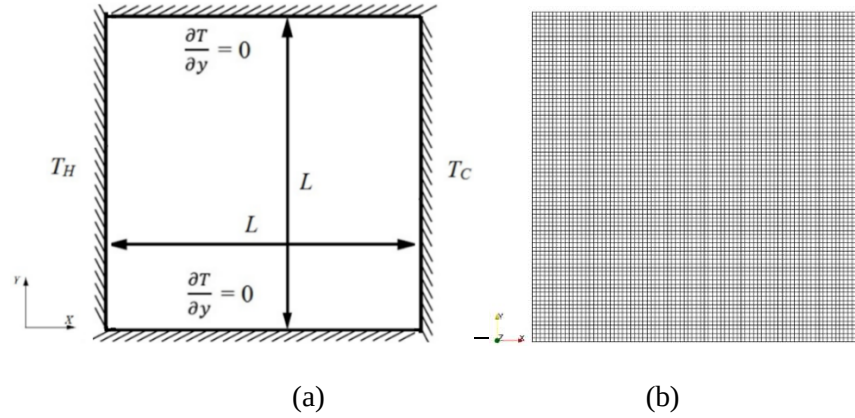


Fig. 5.8 Schematic features (a) and mesh strategy (b) for the cavity case

Tab. 5.2 Key parameters for solver ‘nanofluidMixtureFoam’ validation

$\rho_{air} (kg/m^3)$	$c_p (J/kgK)$	$k (W/mK)$	$\beta (1/K)$	$\nu (m^2/s)$
1.205	1006	0.0256	0.00343	1.511×10^{-5}
$g (m/s^2)$	Pr	$L (m)$	$T_H (K)$	$T_C (K)$
9.81	0.715	0.0168&0.0448	303	283

$$u^* = u / \sqrt{g\beta(T_H - T_C)H} \quad (5.19)$$

$$v^* = v / \sqrt{g\beta(T_H - T_C)H} \quad (5.20)$$

$$T^* = (T - T_C) / (T_H - T_C) \quad (5.21)$$

By the velocity and temperature features presented in Fig. 5.9, it can be observed that the newly developed solver ‘nanofluidMixtureFoam’ worked well for this natural convection case.

From the comparisons of u^* , v^* and T^* in Fig. 5.10~Fig. 5.12, it can be found that the newly developed solver ‘nanofluidMixtureFoam’ obtained the identical results as those predicted by original solver ‘buoyantBoussinesqPimpleFoam’ and the numerical simulations reported by (Barakos et al., 1994). It is indicating that the energy equation has been added successfully and fully functional for heat transfer simulation.

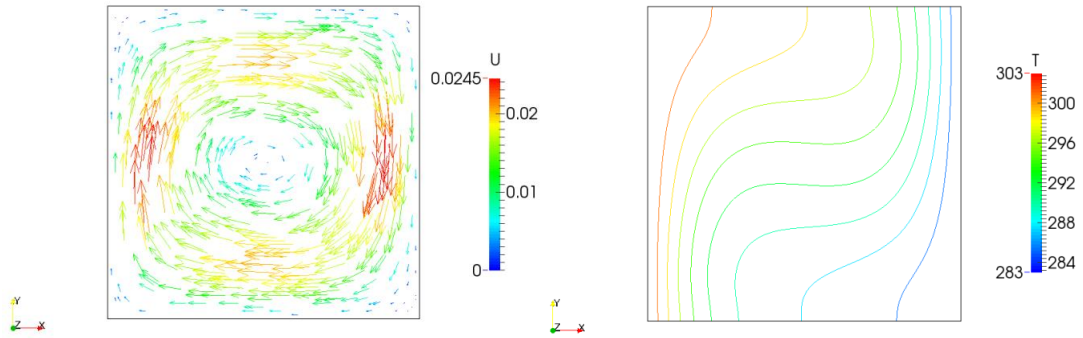


Fig. 5.9 Velocity and temperature features in the first test ($Ra = 1 \times 10^4$)

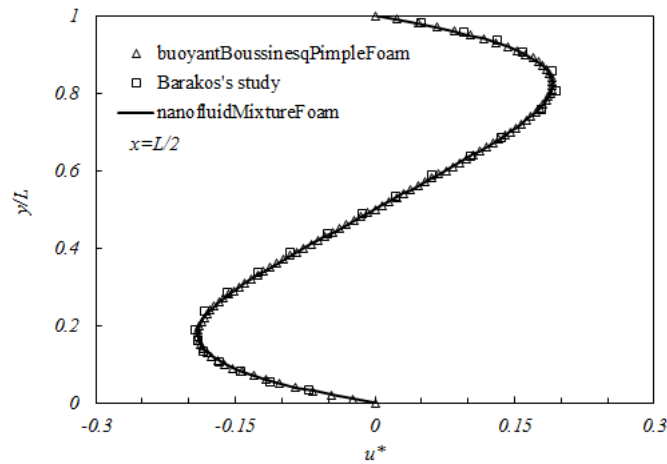


Fig. 5.10 Dimensionless velocity component u^* along $x = L/2$ in the first test

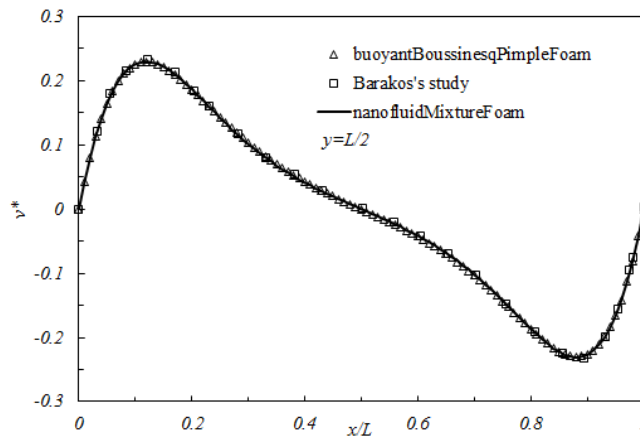


Fig. 5.11 Dimensionless velocity component v^* along $y = L/2$ in the first test

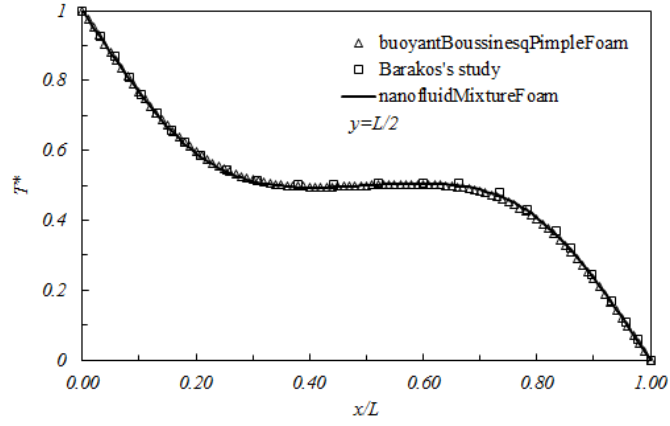


Fig. 5.12 Dimensionless T^* along $y = L/2$ in the first test

5.5 Conclusions

In this chapter, Eulerian-Mixture approach for nanofluid simulation is discussed. Thanks to the features of multi-phase consideration and high computational efficiency, Eulerian-Mixture approach is believed to be the best multi-phase option for nanofluid simulation so far. With this consideration, a new solver ‘nanofluidMixtureFoam’ is developed based on two original OpenFOAM solvers ‘buoyantBoussinesqPimpleFoam’ and ‘driftFluxFoam’. According to the two solver tests in this chapter, it can be concluded that the new solver ‘nanofluidMixtureFoam’ is fully functional for the following investigations: it is able to combine the simulations of nanofluid flow, heat transfer and nanoparticles sedimentation together.

6. APPLICATIONS OF ‘nanofluidMixtureFoam’ TO THREE TYPICAL CASES

6.1 Introduction

In this chapter, three typical geometries are chosen to apply both original solver ‘buoyantBoussinesqPimpleFoam’ (based on conventional single approach) and the newly developed solver ‘nanofluidMixtureFoam’ (which couples nanofluid flow, heat transfer and nanoparticles sedimentation together). The three geometries are two-dimensional square cavity enclosure, three-dimensional horizontal cylinder and two-dimensional channel with an open cavity. More specifically, in the two-dimensional square cavity cases, nanofluid natural convection and nanoparticles sedimentation are combined together to investigate their impacts to each other. Furthermore, the impacts of temperature-dependant variation of nanofluid properties on final numerical results are also discussed. By the three-dimensional horizontal cylinder cases, the difference between two-dimensional and three-dimensional simulations are discussed. In those cases of two-dimensional channel with an open cavity, the relationships between geometrical features, nanofluid heat transfer and nanoparticles sedimentation are investigated.

Due to the difficulty to collect convincing reports related to nanofluid stability, only 0.64% Al_2O_3 /water nanofluid is used in this chapter. However, it does not mean that the newly developed solver ‘nanofluidMixtureFoam’ is valid only for 0.64% Al_2O_3 /water nanofluid. Actually, this solver can be applied to other nanofluids for sure once some more convincing stability information of other nanofluids can be confirmed.

6.2 Nanofluid natural convection combining nanoparticles sedimentation in a two-dimensional square cavity

6.2.1 Introduction

In this section, natural convection of 0.64% Al_2O_3 /water nanofluid in a two-dimensional square cavity is numerically investigated with both conventional single-phase approach and multi-phase Mixture approach. Multi-phase simulation is based on an assumption that nanoparticles sedimentation occurs as soon as the simulation starts. Under this extreme

circumstance, the relationships between nanofluid natural convection and nanoparticles sedimentation are investigated. Furthermore, for each of the two approaches, an extra solver is developed to investigate the impact of nanofluid temperature-dependent properties variations on CFD simulation. Therefore, four numerical approaches in total are applied in this section. The details of the four approaches are given in Tab. 6.1.

Tab. 6.1 Summary of the four approaches in this section

Approach N.O.	Approach name	Description
Approach1	Single-phase approach	Nanofluid is considered as stable and has homogenous properties throughout the simulation.
Approach2	Temperature-dependent single-phase approach	Same as Approach1, but nanofluid properties are updated at each time step according to newly solved nanofluid temperature field.
Approach3	Multi-phase approach	Nanofluid properties are updated at each time step according to newly solved volume fraction field. Nanofluid natural convection and nanoparticles sedimentation are coupled.
Approach4	Temperature-dependent multi-phase approach	Same as Approach3, but nanofluid properties are updated at each time step according to both newly solved volume fraction and temperature fields.

6.2.2 Geometrical configuration and boundary conditions

The geometrical square cavity model in this case is similar as the one Fig. 5.8a. The cavity size is $L = 0.04m$. Top and bottom walls are insulate, i.e. $\frac{\partial T}{\partial y} = 0$. Temperature at left wall (T_H) is higher than that at the right wall (T_C). The temperature difference between heating and cooling walls $\Delta T = T_H - T_C$ is used to adjust Rayleigh number Ra (Ho et al., 2008):

$$Ra = \frac{g\beta\Delta TL^3}{\nu\alpha} \quad (6.1)$$

where α is thermal diffusivity.

The numerical spatial schemes for gradient, Laplacian and divergence are set as ‘Gauss linear’, ‘Gauss linear corrected’ and ‘Gauss linear’ respectively in all the following cases. The correlative approaches were presented detailedly in (Open, 2011). Furthermore in this work, fluid natural convection performance is expressed in terms of average Nusselt number as given in (6.2) (Ternik and Rudolf, 2012), for which temperature gradient $\frac{\partial T}{\partial x}$ at wall can be obtained by collecting the

data from the first grid layer neighbouring to wall.

$$\overline{Nu} = \int_0^L Nu(y) dy / L \quad (6.2)$$

$$Nu(y) = \frac{k_{nf}}{k_f} \frac{\partial T}{\partial x} \bigg|_{x=0} \frac{L}{T_H - T_C} \quad (6.3)$$

As mentioned before, for this geometry, the simulations are carried out in both single- and multi-phase ways. For single-phase simulation, nanofluid is assumed to be stable and no nanoparticles sedimentation exists. For multi-phase simulation, it is assumed that neither stabilizer nor dispersing treatment are applied and nanoparticles sedimentation will occur as soon as the simulation starts. The numerical simulations in this work will be stopped at $t = 7$ hours as in Section 5.2.4.

The geometrical model and mesh are created by Gmsh 2.9.2 (Geuzaine and Remacle, 2009). To enhance simulation accuracy for near wall regions, non-uniform strategy is used to refine the mesh near walls. This is achieved by ‘Bump’ function in Gmsh and grid ratio is set as $\delta = 0.06$. A water natural convection case ($Ra = 10^7$) is employed in grid independence check among four mesh strategies, i.e. 30×30 , 40×40 , 50×50 and 60×60 . Dimensionless temperature T^* and vertical flow velocity U_y^* at different location X^* of the enclosure central line ($Y = 0.5L$) are compared. Dimensionless parameters X^* and U_y^* are defined in (6.4) and (6.5) respectively. Dimensionless temperature T^* was defined in (5.21). By the comparisons in Fig. 6.1, the maximum result differences predicted by a grid strategy and its finer strategy are 1.3%, 0.3% and 0.06% for 30×30 , 40×40 and 50×50 , respectively, indicating that mesh strategy 50×50 is fine enough to obtain trustable results (Bathe et al., 2001). Therefore, mesh strategy 50×50 is eventually selected for this case.

$$X^* = \frac{x}{L} \quad (6.4)$$

$$U_y^* = \frac{u_y}{u_{max}} \quad (6.5)$$

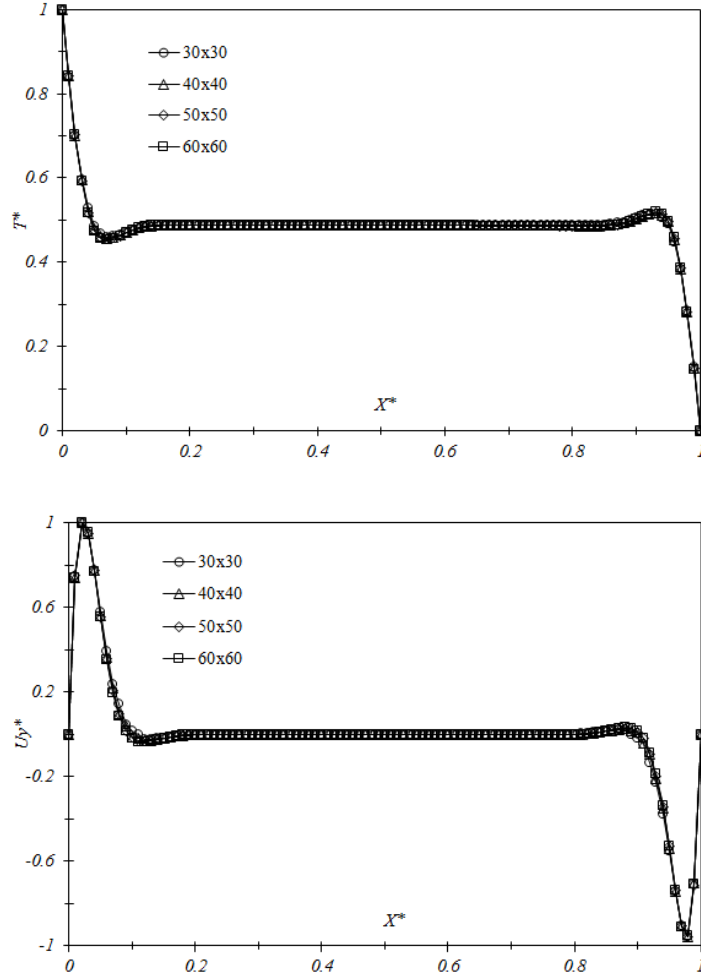


Fig. 6.1 Grid independence check ($Y = 0.5L$, $Ra = 10^7$)

6.2.3 Discussions and conclusions

By the four approaches mentioned in Tab. 6.1, numerical simulations are carried out for Rayleigh number range of $Ra = 10^6 \sim 10^7$. By applying Approach3, the interactions between nanofluid temperature driven flow and nanoparticles sedimentation when $Ra = 1.0 \times 10^6$, $Ra = 5.0 \times 10^6$ and $Ra = 1.0 \times 10^7$ at $t = 7 \text{ hours}$ can be observed in Fig. 6.2~Fig. 6.4. It can be clearly found that nanoparticles sedimentation has considerable impacts to nanofluid natural convection in all the three cases. Owing to the sedimentation layer at cavity bottom, nanofluid temperature driven flow tends to happen in upper region where nanofluid volume fraction is comparatively lower. Regarding the impact of nanofluid natural convection on nanoparticles sedimentation, nanoparticles sedimentation layer tends to be horizontally uniform at cavity bottom when Rayleigh number is small as $Ra = 1.0 \times 10^6$ (this is recognised by those horizontal volume fraction contours between 0.41% and 3.27% in the case $Ra = 1.0 \times 10^6$

presented by Fig. 6.2). However, when Rayleigh number is increased to $Ra = 5.0 \times 10^6$ and $Ra = 1.0 \times 10^7$, nanofluid natural convection is found to have more considerable influence to nanoparticles sedimentation. In the two cases, nanoparticles sedimentation layer can be observed slightly thicker at left bottom due to the circular temperature driven flow (see the volume fraction contour 3.36% in case $Ra = 5.0 \times 10^6$ presented by Fig. 6.3 and volume fraction contour 3.5% in case $Ra = 1.0 \times 10^7$ presented by Fig. 6.4 are higher at left).

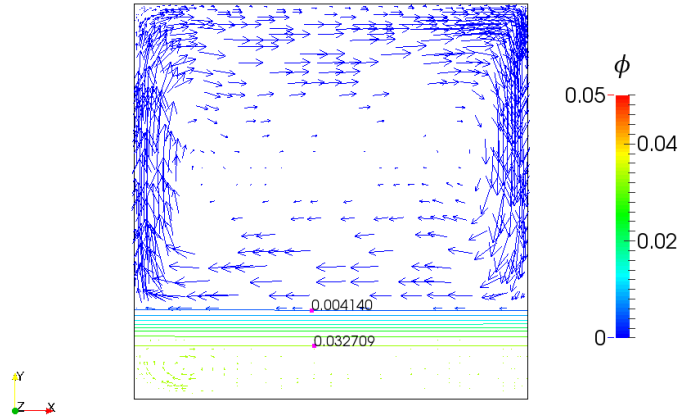


Fig. 6.2 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

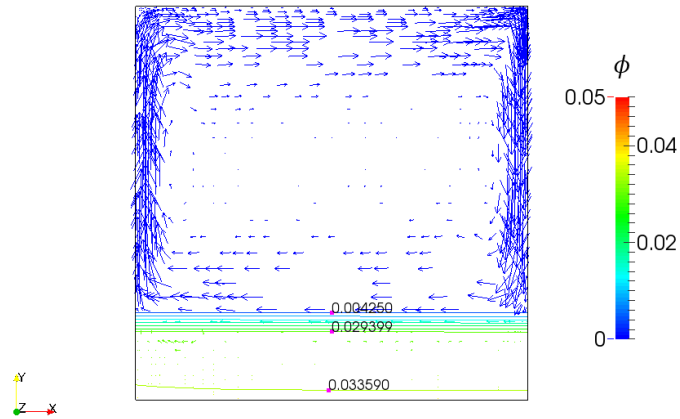


Fig. 6.3 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 5.0 \times 10^6$ ($t = 7$ hours)

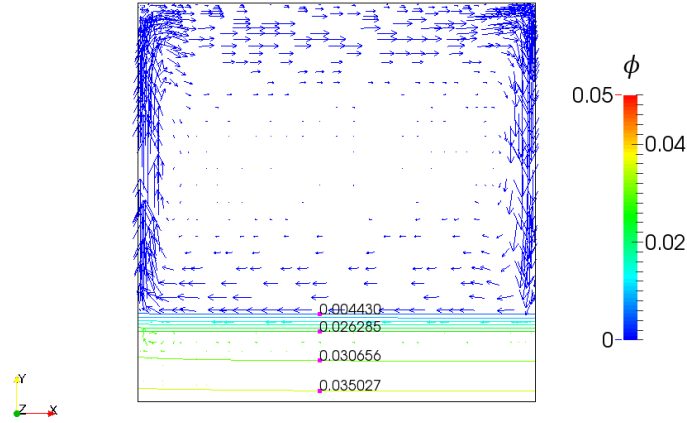


Fig. 6.4 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

Following above analysis, Fig. 6.5 presents the average Nusselt number of 0.64% Al_2O_3 /water nanofluid along the heating wall. It is found that average Nusselt number increases with Rayleigh number, no matter which CFD approach is applied. However, when nanoparticles sedimentation is considered and coupled to nanofluid natural convection in Approach3 and Approach4, average Nusselt number is considerably smaller than those scenarios in which nanofluid is assumed to be stable. The reason is supposed to be the existence of nanoparticles sedimentation layer at cavity bottom. Due to this layer, thermal energy from heating wall tends to transfer through conduction mechanism rather than convection mechanism. According to the definition of Nusselt number (i.e. ratio of convective to conductive heat transfer across the boundary (Tetsu and Motoo, 1976)), it is not a surprise that the average Nusselt number along heating wall is deteriorated.

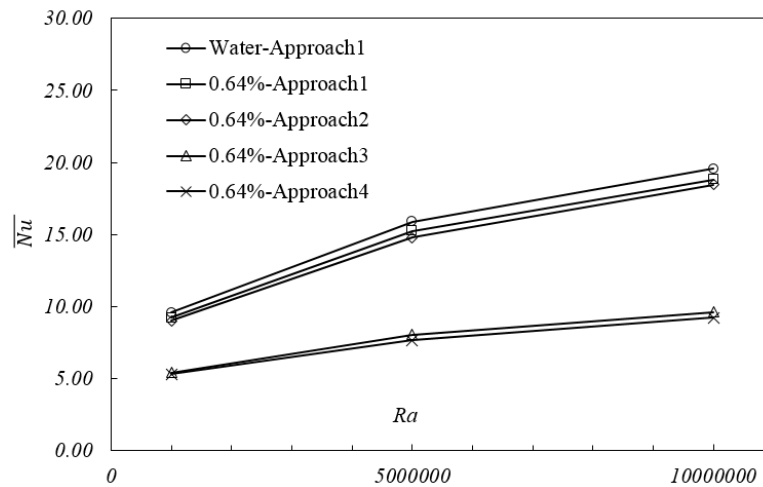


Fig. 6.5 Average Nusselt number against Rayleigh number of 0.64% Al_2O_3 /water nanofluid

Furthermore in Fig. 6.5, average Nusselt number of pure water is found slightly greater than that of 0.64% nanofluid in all the cases of $Ra = 1.0 \times 10^6$, 5.0×10^6 and 1.0×10^7 . This observation actually conforms to most experimental investigations (Putra et al., 2003, Wen and Ding, 2005b, Li and Peterson, 2009). For such observation in above cases, the reason is supposed to be the increased nanofluid viscosity is playing a dominant role to deteriorate heat convection by impeding the nanofluid flow (Li and Peterson, 2009).

Also, in Fig. 6.5, it can be found that the consideration of temperature-dependent properties variations lead to slightly lower average Nusselt number prediction. However, the maximum difference between Approach1 and Approach2 is only 2.1% (1.7% between Approach3 and Approach4). This indicates that such consideration is not necessary in CFD simulation in terms of raising computational efficiency. Therefore, the simulations in following sections will be carried out by Approach1 and Approach3 only to reduce computational effort.

Fig. 6.6~Fig. 6.8 present the temperature contours of cases $Ra = 1.0 \times 10^6$, 5.0×10^6 and 1.0×10^7 respectively at $t = 7 \text{ hours}$. By comparisons, the temperature decreasing features near heating wall predicted by single-phase approaches (Approach1 and Approach2) are found to have same tendencies as the temperature increasing features near cooling wall. However, due to nanoparticles sedimentation layer at cavity bottom, the temperature contours such as $T = 301.58K$, $T = 301.18K$, $T = 300.79K$ and $T = 300.39K$ predicted by multi-phase approaches (Approach3 and Approach4) are found to be more horizontally uniform at cavity bottom. This is a good support to the previous conclusion that thermal energy from heating wall tends to transfer through conduction rather than convection due to nanoparticles sedimentation layer at cavity bottom.

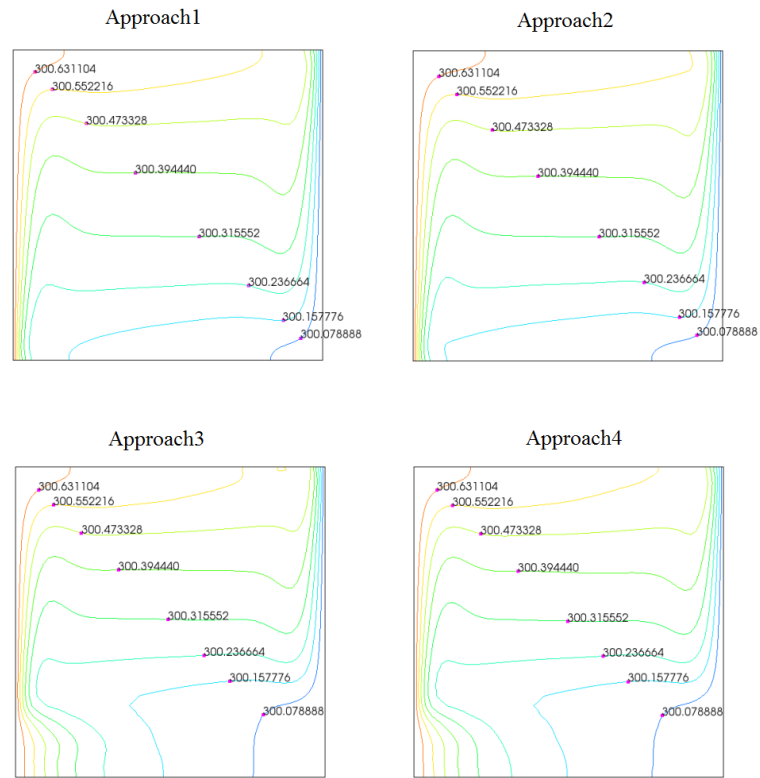


Fig. 6.6 Temperature contours comparison of 0.64% nanofluid at $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

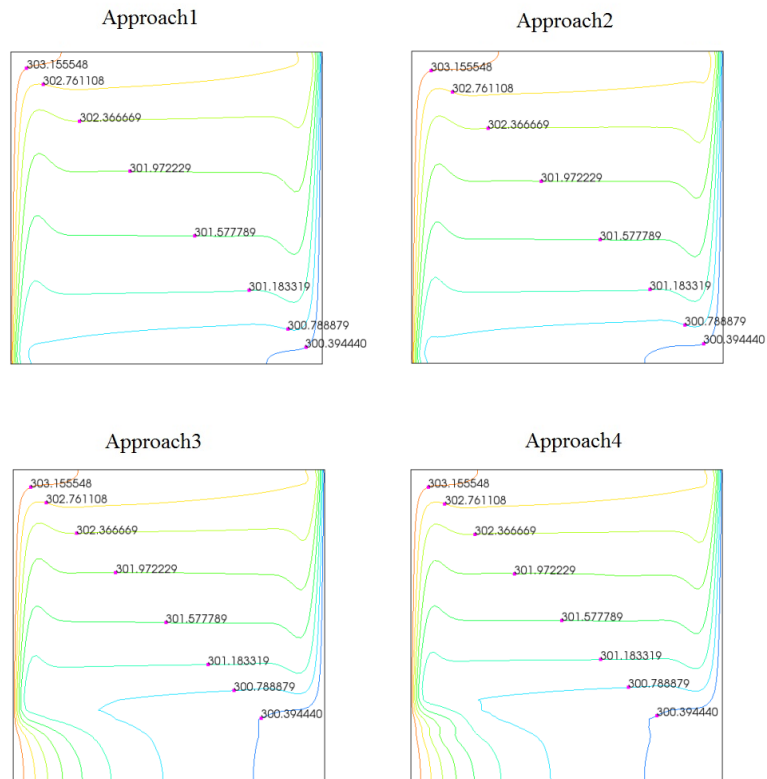


Fig. 6.7 Temperature contours comparison of 0.64% nanofluid at $Ra = 5.0 \times 10^6$ ($t = 7$ hours)

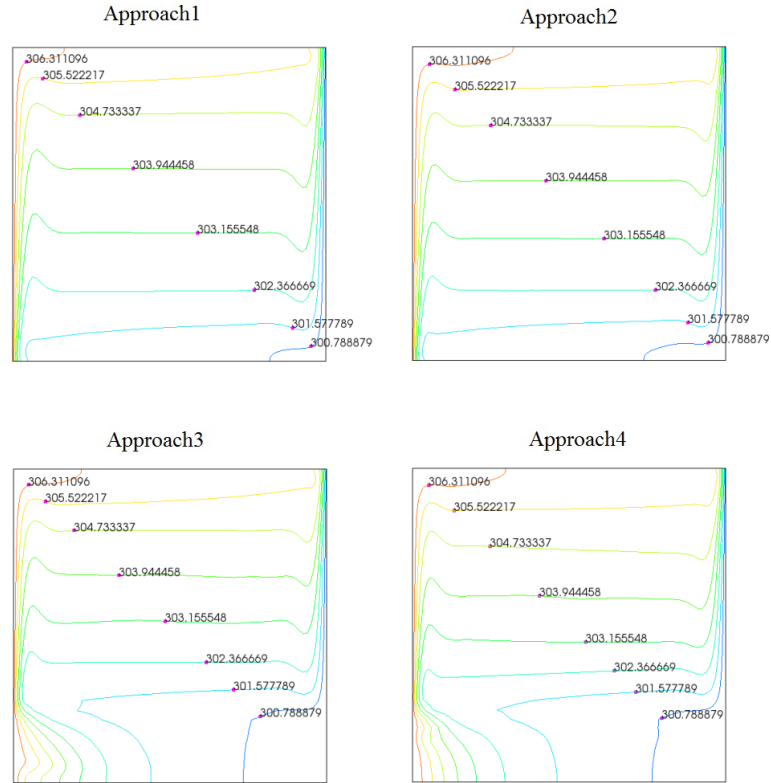


Fig. 6.8 Temperature contours comparison of 0.64% nanofluid at $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

6.3 Nanofluid natural convection combining nanoparticles sedimentation in a three-dimensional horizontal cylinder

6.3.1 Introduction

In this Section, only Approach1 and Approach3 are applied rather than all of the four approaches in Tab. 6.1. Regarding the case geometry, it is a three-dimensional cylinder which is laid down horizontally as in the experimental investigation reported by (Putra et al., 2003). This section aims to compare the difference between two-dimensional investigation (which is popular in previous numerical studies of nanofluid natural convection) and three-dimensional investigation (which is rarely to see previously).

More specifically, two questions are going to be answered in this section:

1. When nanofluid is treated as a stable single-phase fluid, what is the difference between the results predicted in the same sized two-dimensional cavity and three-dimensional cylinder?

2. When nanofluid can be treated as multi-phase mixture, what is the difference between the results predicted by multi-phase solver ‘nanofluidMixtureFoam’ and single-phase solver ‘buoyantBoussinesqPimpleFoam’?

6.3.2 Geometrical configuration and boundary conditions

The horizontal cylinder model in this case is created by Gmsh 2.9.2. The schematic model is shown in Fig. 6.9. The length and diameter of the cylinder are $L = 0.04m$ and $D = 0.04m$ respectively. Cylinder wall is considered as insulate, i.e. $\left. \frac{\partial T}{\partial r} \right|_{r=\frac{D}{2}} = 0$. $\Delta T = T_H - T_C$, the temperature difference between heating and cooling ends is used to adjust Rayleigh number to $Ra = 1.0 \times 10^6$, $Ra = 5.0 \times 10^6$ and $Ra = 1.0 \times 10^7$ as in Section 6.2.3. For this cylinder, mesh is refined at near wall regions (Fig. 6.10) for more accurate sedimentation and natural convection simulations. After grid independence check following the same way as in Section 6.2.2, the strategy with cells amount $n = 84000$ is chosen for following simulations.

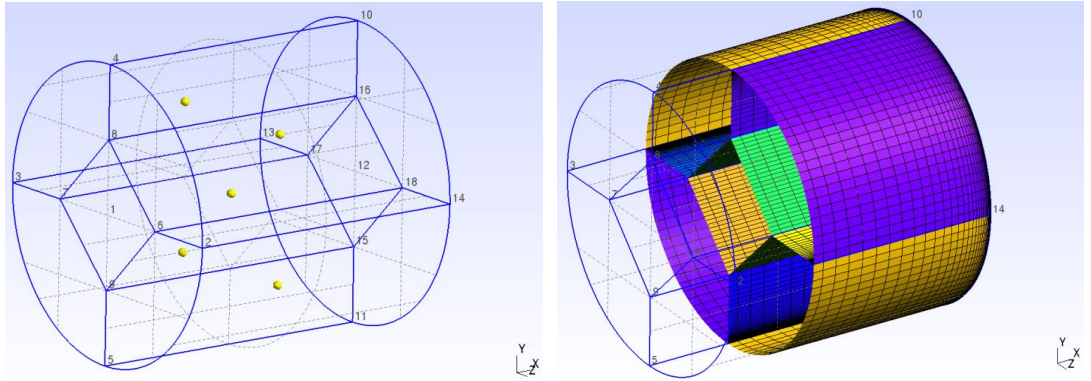


Fig. 6.9 The schematic model of the cylinder in Gmsh 2.9.2

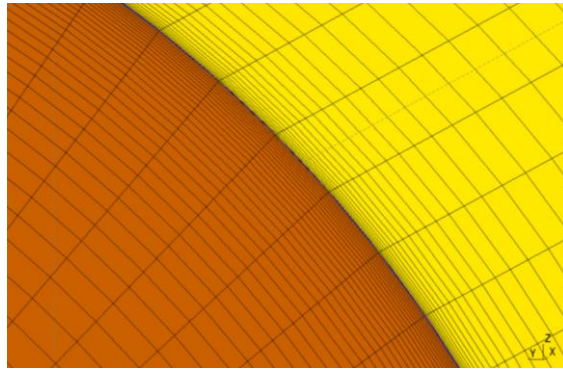


Fig. 6.10 The non-uniform mesh strategy for the horizontal cylinder

Due to the cells amount in this three-dimensional region is considerably more than that in the

corresponding two-dimensional cases, it is necessary to optimise computational resource to shorten computational time. For this purpose, parallel simulation strategy is employed in this work. In OpenFOAM, the basic idea of parallel simulation is to divide the whole computational region into different parts and allocate them to different CPUs before running the case. During the simulation, different CPUs will exchange data between different divided regions continuously. After the simulation, the whole computational region should be reconstructed again for post processing.

However, it does not mean that using more CPUs (or more sub-regions) is definitely better to shorten computational time (Culpo, 2011). The data exchange between different sub-regions also consumes resource of random-access memory (RAM), as well as computational time. For the cylinder in this work with the given mesh plan, a computer with four CPUs is used, the configuration of which is Intel (R) Core (TM) i5-4690@3.50GHz. According to the hardware configuration, the computational region domain of each case in present simulations is divided to two identical ones along x axis. Thus, by this computer, two cases can run together for the best working efficiency.

6.3.3 Discussions and conclusions

6.3.3.1 Approach1 for the three-dimensional horizontal cylinder

Firstly, original solver ‘buoyantBoussinesqPimpleFoam’ is applied to the three-dimensional cases with $Ra = 1.0 \times 10^6$, $Ra = 5.0 \times 10^6$ and $Ra = 1.0 \times 10^7$. The results are compared to those corresponding two-dimensional cases. This aims to investigate the difference between nanofluid flow and heat transfer performances in two-dimensional cavity and three-dimensional cylinder. Fig. 6.11 shows a general view of temperature distribution in case $Ra = 1.0 \times 10^6$ at $t = 7$ hours. It can be seen that, in horizontal cylinder, the temperature driven flow still circulates in the direction which is vertical to heating and cooling ends. Furthermore, the temperature driven flow has noticeable impacts to temperature field.

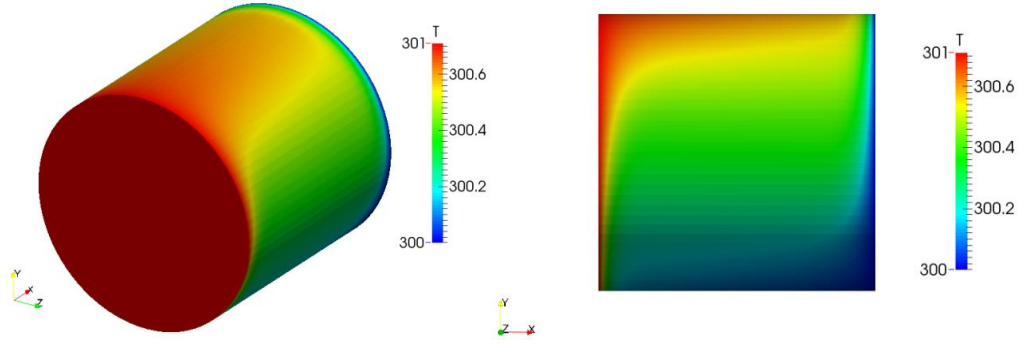


Fig. 6.11 Temperature distribution with a three-dimensional view, $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

In Fig. 6.12, the comparison of temperature distributions on central planes of two- and three-dimensional cases ($Ra = 1.0 \times 10^6$) is presented at $t = 7$ hours. It can be seen that there is no noticeable difference between the two cases. More strictly, by the comparison of temperature distribution T (on central plane) along $y = 0.02m$ in cases $Ra = 1.0 \times 10^6$, $Ra = 5.0 \times 10^6$ and $Ra = 1.0 \times 10^7$ (Fig. 6.13, Fig. 6.14 and Fig. 6.15), there is nearly no difference can be found between the temperature measurements from two- and three-dimensional cases.

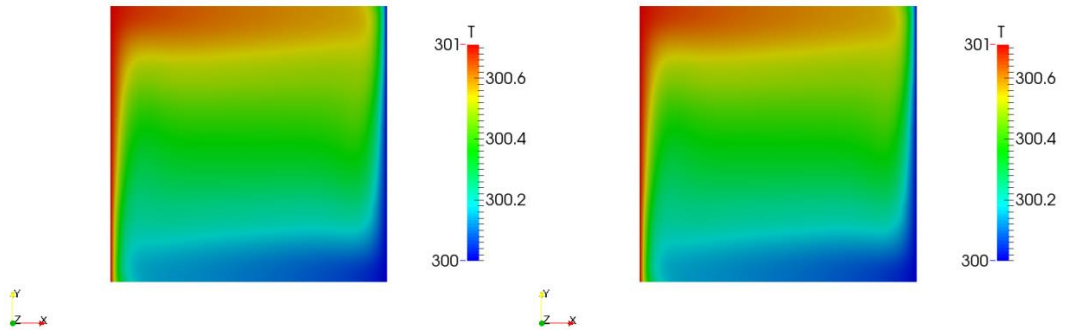


Fig. 6.12 Temperature distributions on central planes of two- and three-dimensional cases, $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

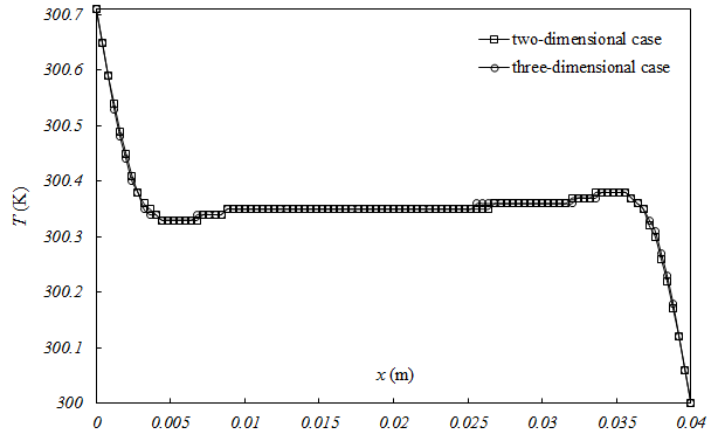


Fig. 6.13 Temperature T along $y = 0.02m$, $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

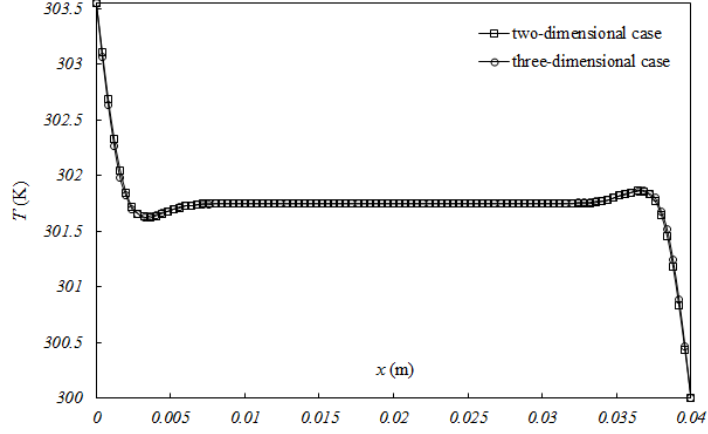


Fig. 6.14 Temperature T along $y = 0.02m$, $Ra = 5.0 \times 10^6$ ($t = 7$ hours)

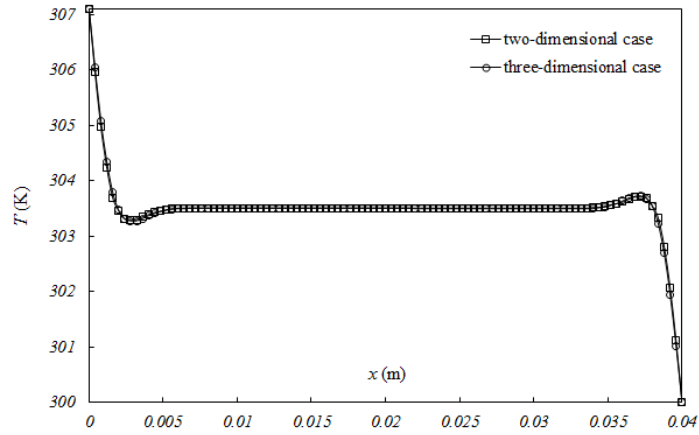


Fig. 6.15 Temperature T along $y = 0.02m$, $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

Regarding the flow conditions of the three cases, they are all laminar flow with Reynolds number (calculated by maximum velocity) $Re = 35$, $Re = 85$ and $Re = 120$, respectively. However, by the comparisons of vertical velocity u_y (on central plane) along $y = 0.02m$ as shown in Fig. 6.16, Fig. 6.17 and Fig. 6.18, it can be found that differences actually exist between two- and three-dimensional cases. More specifically, taking Fig. 6.16 as example, U_y along $y = 0.02m$ can be observed slightly smaller in three-dimensional case (due to the friction impacts from cylinder wall) than that in the corresponding two-dimensional case. This phenomenon indicates that the investigations of simplified two-dimensional cases should be a bit different from those experimental studies which are carried out in horizontal cylinders.

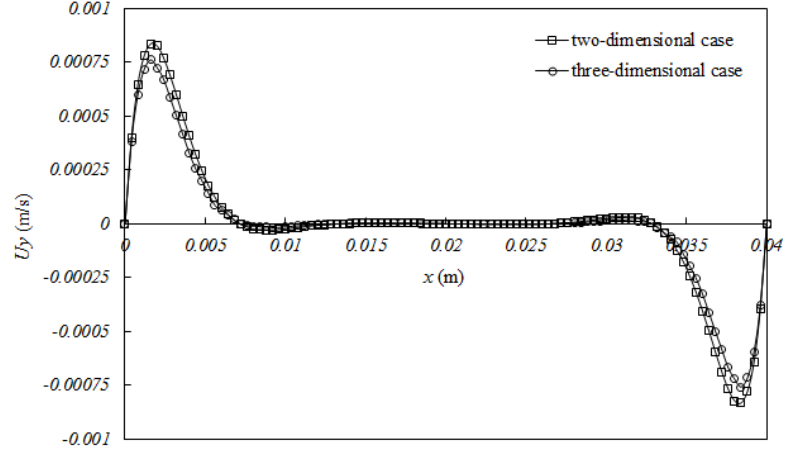


Fig. 6.16 Velocity U_y along $y = 0.02m$, $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

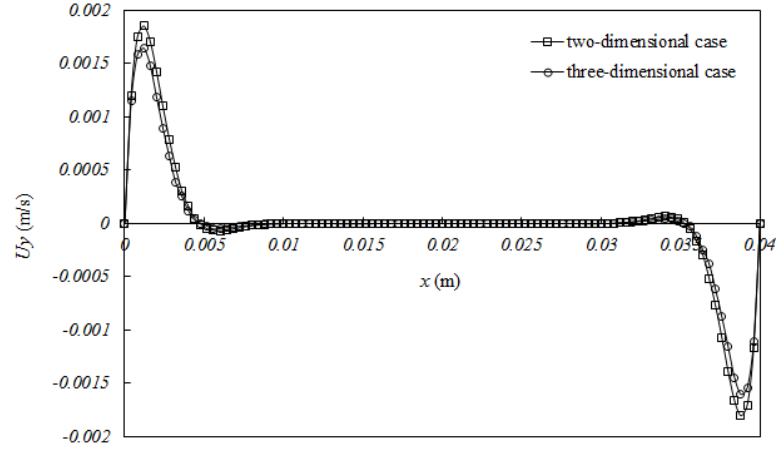


Fig. 6.17 Velocity U_y along $y = 0.02m$, $Ra = 5.0 \times 10^6$ ($t = 7$ hours)

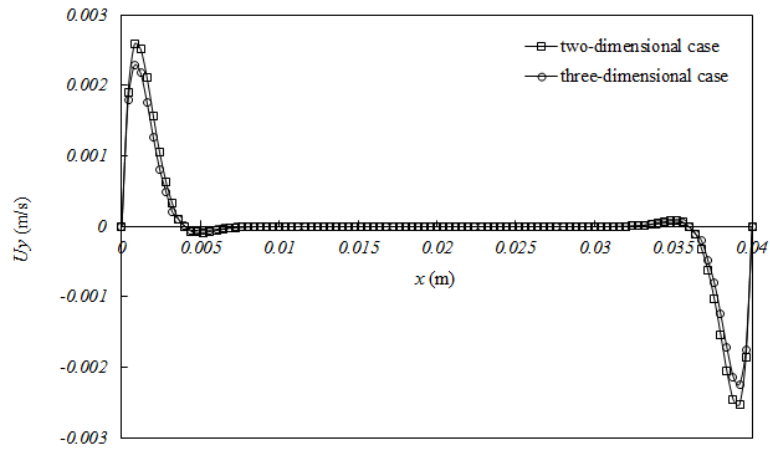


Fig. 6.18 Velocity U_y along $y = 0.02m$, $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

6.3.3.2 Coupling of nanofluid natural convection and nanoparticles sedimentation

By applying Approach3, the interactions between nanofluid natural convection flow and

nanoparticles sedimentation when $Ra = 1.0 \times 10^6$, $Ra = 5 \times 10^6$ and $Ra = 1.0 \times 10^7$ at $t = 7 \text{ hours}$ can be observed in Fig. 6.19~Fig. 6.21. In this horizontal cylinder, it also can be found that nanoparticles sedimentation has considerable impacts to nanofluid natural convection in all the three cases. Owing to the sedimentation layer at cavity bottom, nanofluid temperature driven flow tends to happen in upper region where nanofluid volume fraction is comparatively lower. This conclusion is same to that in Section 6.2.3.

However, if compared Fig. 6.19~Fig. 6.21 to those corresponding two-dimensional cases presented in Fig. 6.2~Fig. 6.4, it can be found that the nanoparticles sedimentation layers in three-dimensional cases are notably thicker. Due to this observation, it can be confirmed that the (three-dimensional) geometrical features have considerable impacts on nanoparticles sedimentation, indicating that using two-dimensional cavity geometry instead of horizontal cylinder may be not a good idea, although this simplification saves computational resource considerably.

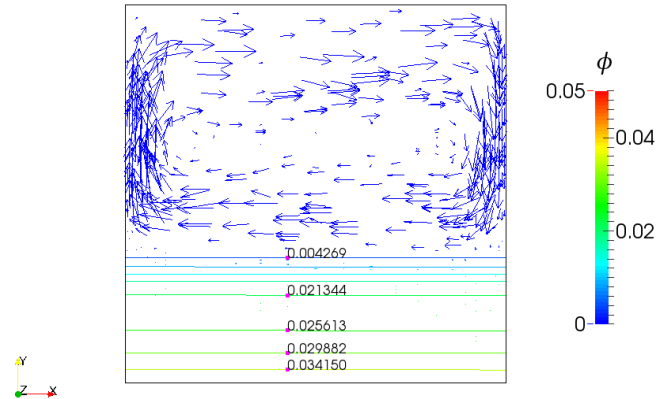


Fig. 6.19 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^6$ ($t = 7 \text{ hours}$)

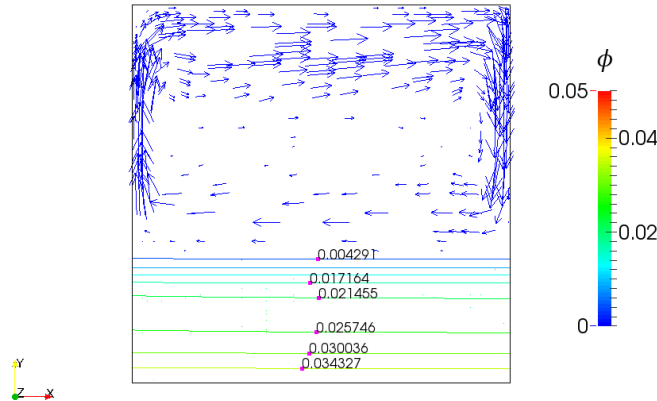


Fig. 6.20 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 5.0 \times 10^6$ ($t = 7 \text{ hours}$)

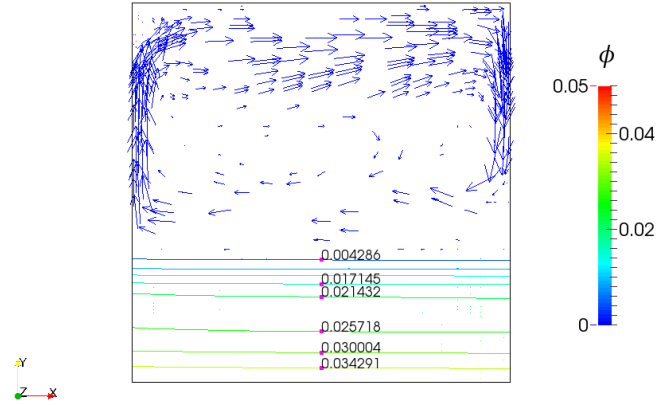


Fig. 6.21 0.64% Al_2O_3 /water nanofluid temperature driven flow and nanoparticles sedimentation interactions in case $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

Following above analysis, Fig. 6.22 presents the average Nusselt number of 0.64% Al_2O_3 /water nanofluid at the heating end of cylinder. It is found that average Nusselt number increases with Rayleigh number (no matter which CFD approach is applied). However, when nanoparticles sedimentation is considered and coupled to nanofluid natural convection in Approach3, average Nusselt number is considerably smaller than those scenarios in which nanofluid is assumed to be stable without sedimentation considerations. As analysed in Section 6.2.3, the possible reason is supposed to be the existence of nanoparticles sedimentation layer at cavity bottom (Fig. 6.19~Fig. 6.21). Due to this layer, thermal energy from the heating wall tends to transfer through conductivity mechanism rather than convection mechanism.

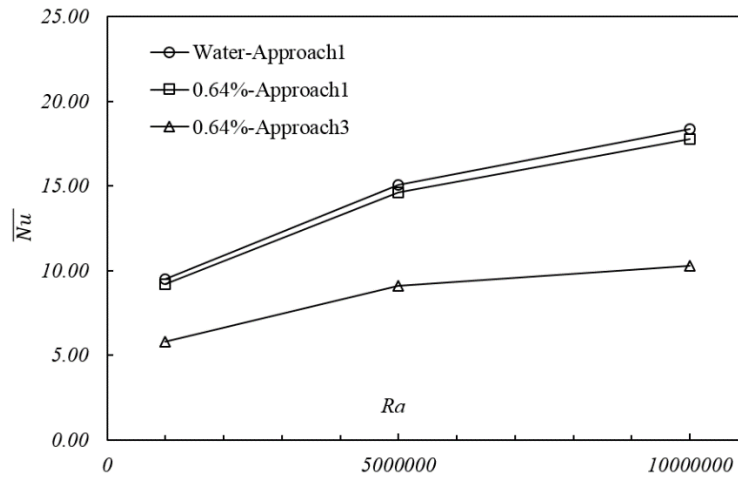


Fig. 6.22 Average Nusselt number against Rayleigh number predicted by different approaches ($t = 7$ hours)

If compared Nusselt number of 0.64% Al_2O_3 /water nanofluid along the heating wall between two- and three-dimensional cases (Fig. 6.23), some interesting things also can be observed. When

nanofluid is assumed to be stable with homogenous properties, predictions of Nusselt number are slightly lower in three-dimensional cases. The reason is still supposed to be the impacts from three-dimensional cylinder wall, which is not able to be considered in two-dimensional cases. Again, this observation indicates that two-dimensional cavity cases cannot be used instead of three-dimensional cylinder cases when high simulation accuracy is required.

However in present simulations, when nanofluid is considered as multi-phase mixture and nanoparticles sedimentation is combined to nanofluid natural convection, predictions of Nusselt number (presented by the solid line at bottom of Fig. 6.23) are slightly higher in three-dimensional cases. Unfortunately, the reason for this is still unclear yet and further experimental investigations are necessary to verify this numerical phenomenon in present work.

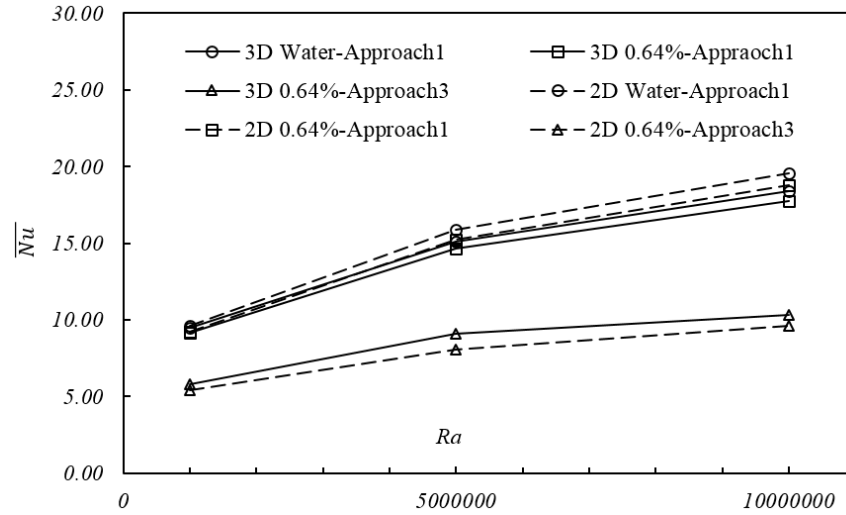


Fig. 6.23 Average Nusselt number comparison between two- and three-dimensional cases ($t = 7$ hours)

Fig. 6.24~Fig. 6.26 present the temperature contours of three-dimensional cases $Ra = 1.0 \times 10^6$, 5.0×10^6 and 1.0×10^7 respectively at $t = 7$ hours. By the comparisons between Approach1 and Approach3, it can be clearly found that temperature variations are more considerable between heating and cooling walls at cavity bottom region due to nanoparticles sedimentation layer. This observation is similar to that in Section 6.2.3. Again, this is a good support to the conclusion that thermal energy from heating wall tends to transfer through conduction rather than convection due to the nanoparticles sedimentation layer at cavity bottom.

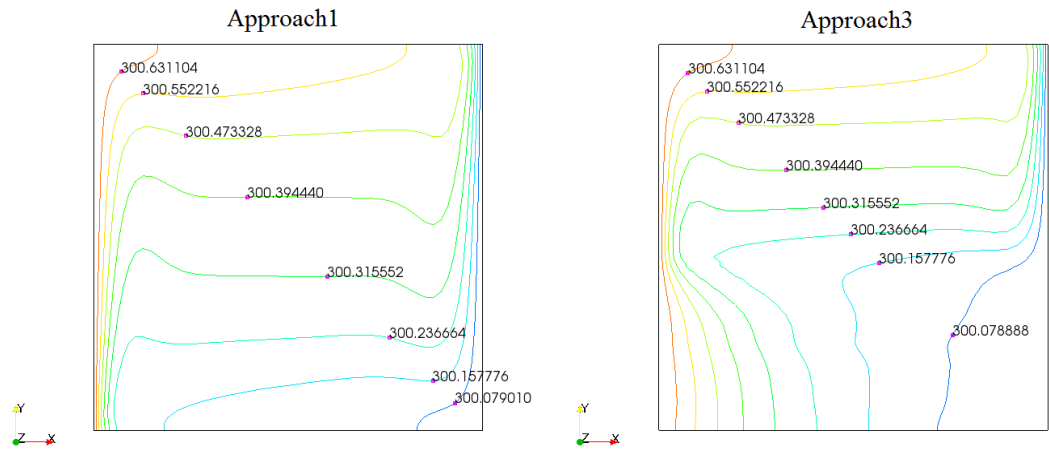


Fig. 6.24 Temperature contours comparison at $Ra = 1.0 \times 10^6$ ($t = 7$ hours)

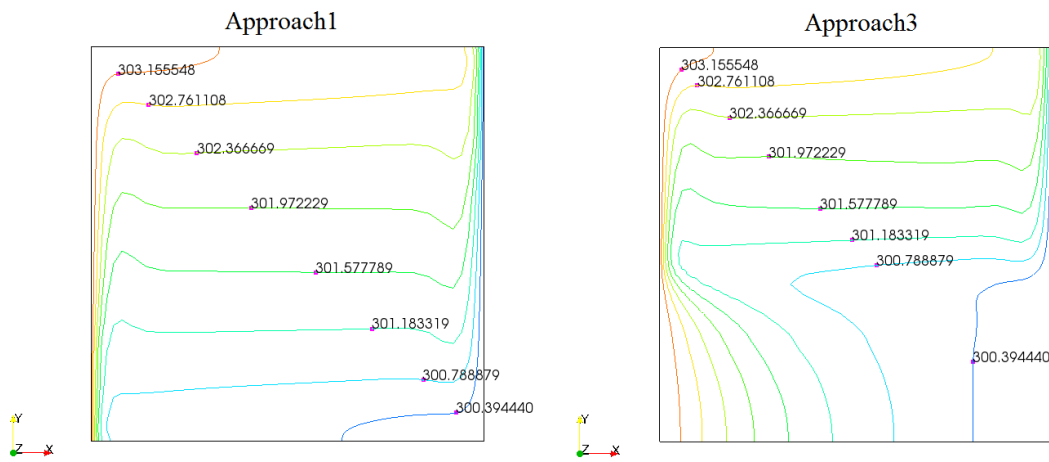


Fig. 6.25 Temperature contours comparison at $Ra = 5.0 \times 10^6$ ($t = 7$ hours)

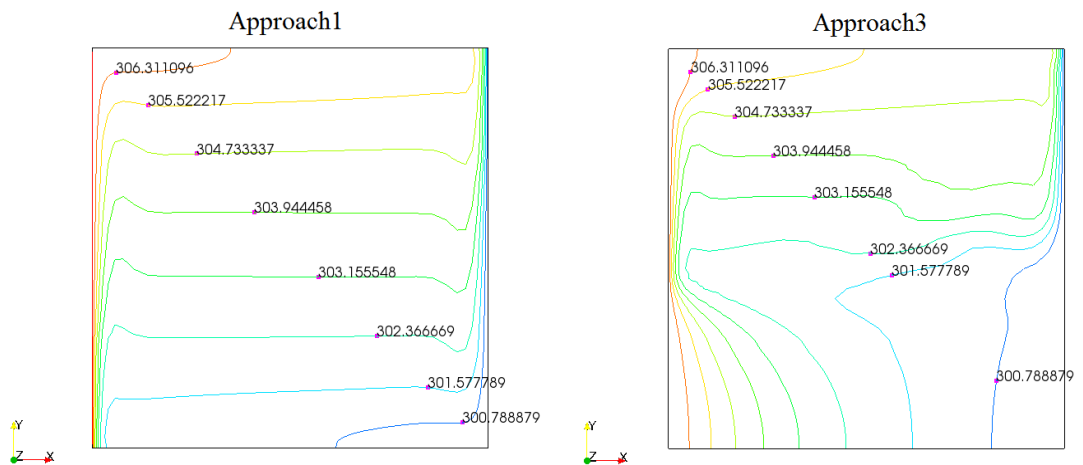


Fig. 6.26 Temperature contours comparison at $Ra = 1.0 \times 10^7$ ($t = 7$ hours)

6.4 Forced convection in a two-dimensional channel with open cavity

6.4.1 Introduction

In previous CFD simulations of forced nanofluid convection, geometries such as horizontal

channel (Lee and Mudawar, 2007), circular tube (Bianco et al., 2009), channel with a back-facing step (Mohammed et al., 2011) and channel with an open cavity (Mehrez et al., 2013) are investigated quite often. In which, horizontal channel with an open cavity heated at bottom is the best one to present the interactions between nanoparticles sedimentation and nanofluid heat transfer performance.

In this section, for a two-dimensional channel with different sized open cavities heating at bottom, a numerical study is carried out to investigate the relationships between nanofluid flow, heat transfer and nanoparticles sedimentation. More specifically, Approach3 in Tab. 6.1 is used for 0.64% Al_2O_3 /water nanofluid in multi-phase way. Furthermore, to obtain the references for later comparisons, Approach1 in Tab. 6.1 is also employed correspondingly in single-phase way. The simulations are going to be carried out under two different laminar flow conditions (i.e. $Re = 500$ and 1000) and three questions as follows will be answered in this section:

1. Without nanoparticles sedimentation consideration, does 0.64% Al_2O_3 /water nanofluid has better heat transfer performance than water in the horizontal channel with different sized open cavities heated at bottom?
2. With nanoparticles sedimentation consideration, how does nanoparticles sedimentation influence nanofluid heat transfer performance in the horizontal channel with different sized open cavities heated at bottom?
3. With different open cavity geometrical features, how the heat transfer performance of the channel heated at cavity bottom can be influenced?

6.4.2 Geometrical configuration and boundary conditions

The geometrical model in consideration is a horizontal channel (total length is L and inlet height is H) with inlet at upstream and outlet at downstream. In middle section, there is an open cavity with a heating bottom (Fig. 6.27). With such geometrical features, the cases with different cavity heights h and lengths L_H are considered (Fig. 6.28). More specifically, the ratio of h/H is set to $h/H = 0.5, 1$ and 1.5 , respectively, while the ratio L_H/H is set to $L_H/H = 0.5, 1$ and 1.5 , respectively (Tab. 6.2). In present simulations, the inlet height H is $H = 1\text{cm}$, while the total channel length is $L = 7\text{cm}$.

Tab. 6.2 Codes of the nine geometrical models

	$h = 0.5H$	$h = H$	$h = 1.5H$
$L_H = 0.5H$	Geometry1	Geometry2	Geometry3
$L_H = H$	Geometry4	Geometry5	Geometry6
$L_H = 1.5H$	Geometry7	Geometry8	Geometry9

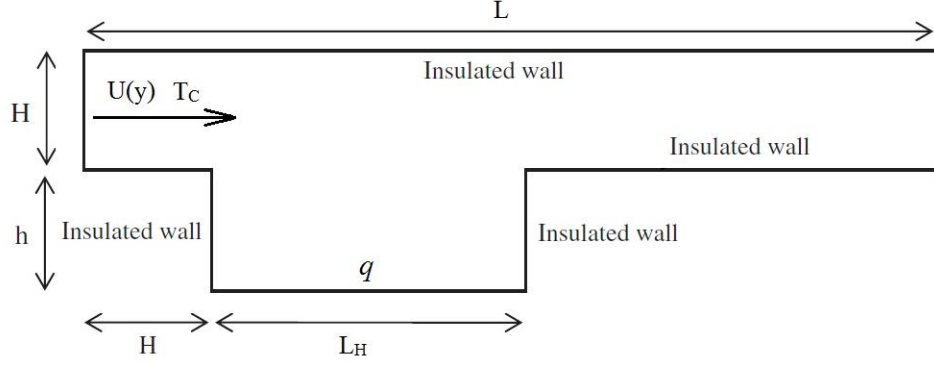


Fig. 6.27 Schematic of the channel with an open cavity heated at bottom

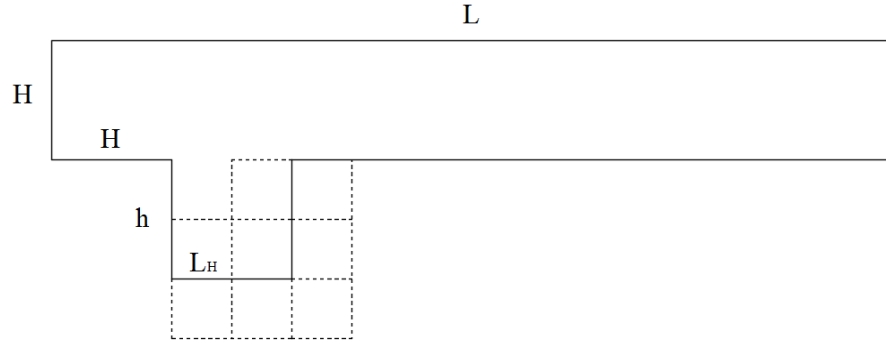


Fig. 6.28 Geometrical variations of the cavity with heating bottom

For inlet boundary, fluid temperature is set as uniform and stable with $T_c = 300K$. Regarding the average inlet velocity, it is set as $U_{ave} = 0.0425m/s$ and $0.085m/s$, respectively. Thus the Reynolds number are $Re = 500$ and 1000 , respectively. To create a well-developed flow regime at the inlet, inlet velocity $U(y)$ is considered as parabolic. All walls are insulated in terms of temperature gradient, i.e. $\frac{\partial T}{\partial x} = 0$ for vertical walls and $\frac{\partial T}{\partial y} = 0$ for horizontal walls. The cavity bottom is heated with a uniform heat flux q . In OpenFOAM, the heating boundary is set as a uniform fixed temperature gradient $\frac{\partial T}{\partial y} = 2000 K/m$.

Gmsh 2.9.2 is still employed to create the geometrical models in Tab. 6.2 (as shown in Fig. 6.29) and mesh. To obtain more reliable simulations of flow and heat transfer performance as well as nanoparticles sedimentation features, the mesh in near wall regions are refined by functions of

‘Bump’ and ‘Progression’ in Gmsh (Fig. 6.30). The bumping and progression constants are used as 0.1 and 0.95, respectively. Following the grid independence check method applied by (Mehrez et al., 2013), a non-uniform mesh strategy with cell number $n = 7200$ is selected for the most typical geometry, i.e. Geometry5 in Fig. 6.29 ($h = L_H = H$). This strategy is used as the meshing reference for other geometrical models.

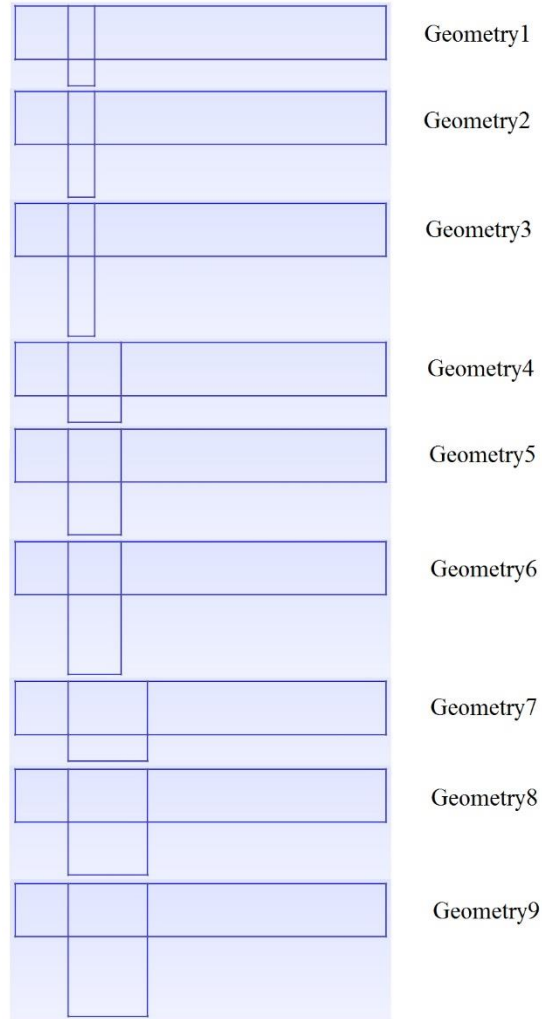


Fig. 6.29 Different geometry models created by Gmsh

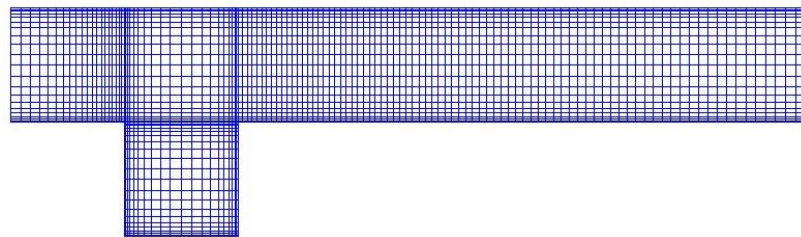


Fig. 6.30 Non-uniform scheme for mesh in following cases

6.4.3 Discussions and conclusions

In practical applications, the temperature along heating surface is the most straightforward parameter to measure the heat transfer performance of coolant. In this section, the temperature at open cavity bottom (heating surface) is used to present heat transfer performance of water and 0.64% Al_2O_3 /water nanofluid in different designs. With this consideration, Fig. 6.31 and Fig. 6.32 are presented to compare heating bottom temperature features of the nine geometries with $Re = 500$ and 1000, respectively. The positions of the sub-figures in Fig. 6.31 and Fig. 6.32 are corresponding to those geometrical designs which were listed in Tab. 6.2.

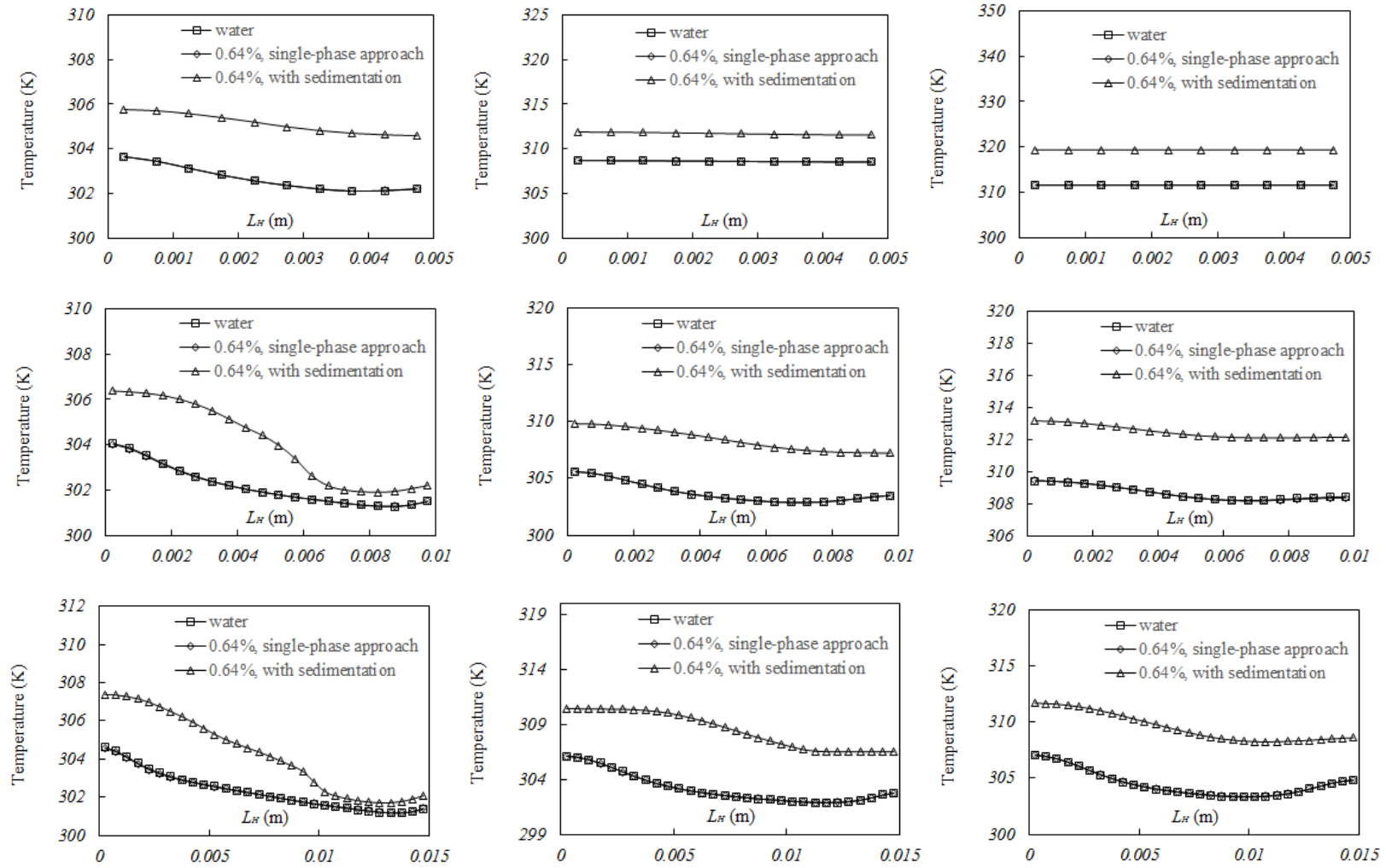


Fig. 6.31 Heating bottom temperatures of the nine geometries ($Re = 500$)

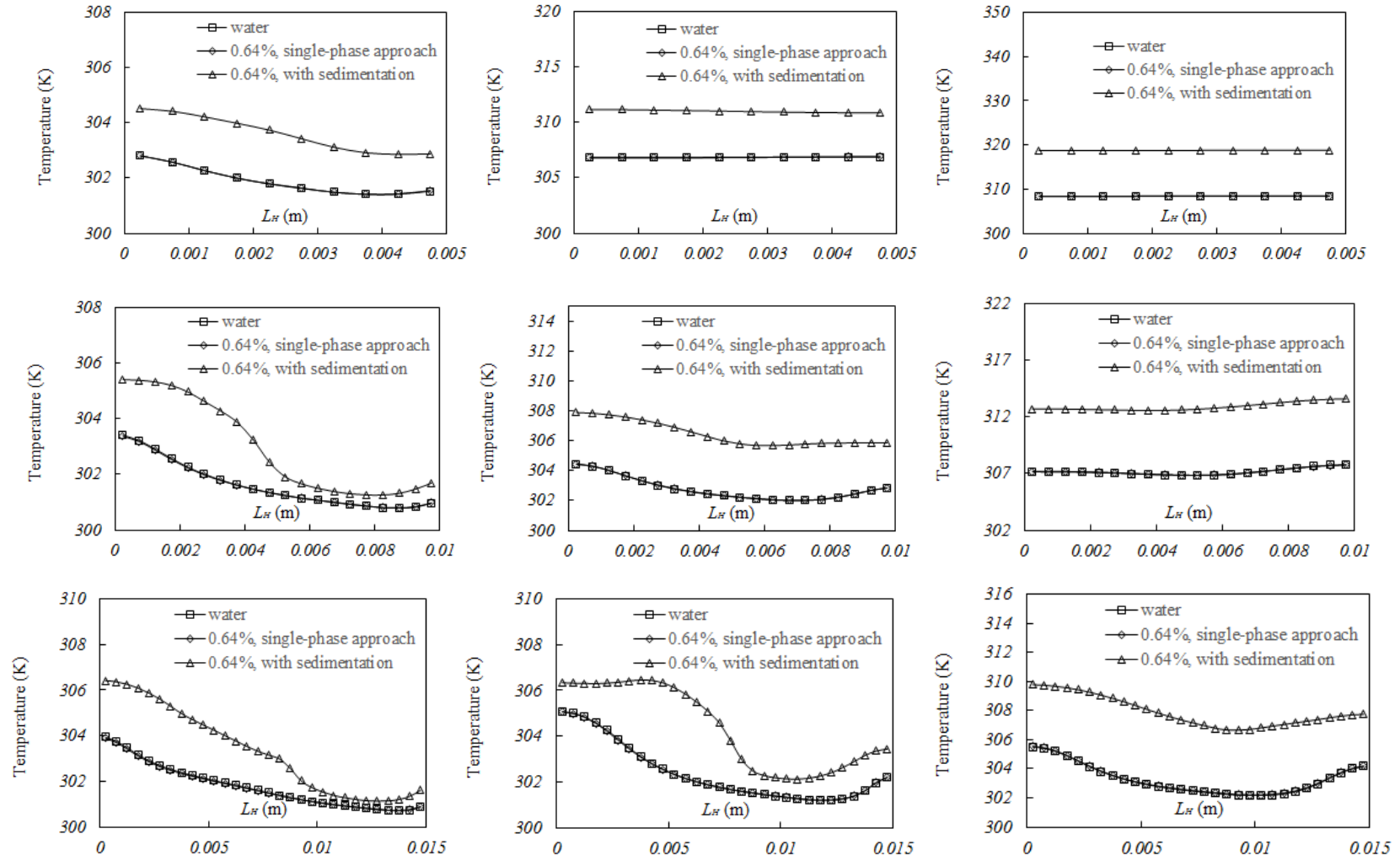


Fig. 6.32 Heating bottom temperatures of the nine geometries ($Re = 1000$)

By the comparisons in Fig. 6.31 and Fig. 6.32, it can be observed that the difference of heat transfer performance between water and 0.64% Al_2O_3 /water nanofluid is not noticeable without nanoparticles sedimentation consideration. The temperatures at cavity bottom in those water cases and 0.64% Al_2O_3 /water nanofluid cases predicted by single-phase approach are nearly exactly same (the maximum difference in all the comparisons is 0.07% only). It is indicating that if without nanoparticles sedimentation problems, pure water should be the first choice rather than Al_2O_3 /water nanofluid with a low volume fraction such as $\phi = 0.64\%$. This is because pure fluid coolant is usually much easier and cheaper to obtain than nanofluid.

Furthermore, when nanoparticles sedimentation is considered in the simulations with multi-phase approach, nanoparticles sedimentation is clearly found to deteriorate heat transfer performance of 0.64% Al_2O_3 /water nanofluid in all designs. This conclusion is made based on the phenomenon that the heating surface temperature is noticeably higher in each geometry if nanoparticles sedimentation problem is considered in simulations (Fig. 6.31 and Fig. 6.32). The reason is supposed to be nanoparticles sedimentation layer at cavity bottom increases the mixture viscosity (which could impede flow). Consequently, the deteriorated flow will lead to bad convective heat transfer performance.

Regarding geometrical designs with different ratios of h/H and L_H/H , it can be found that greater cavity height h induces higher heating surface temperature. Taking the case $L_H/H = 0.5$ with $Re=500$ (the top three sub-figures of Fig. 6.31) as example (water or 0.64% Al_2O_3 /water nanofluid without nanoparticles sedimentation consideration), the maximum temperature at heating surface could be $T = 311.5K$ in design $h/H=1.5$ (the top right sub-figure of Fig. 6.31) while $T = 303.6K$ in design $h/H=0.5$ (the top left sub-figure of Fig. 6.31). If nanoparticles sedimentation is considered, the temperature difference could be $\Delta T = 13.6K$ between the designs of $h/H=1.5$ and $h/H=0.5$. To a cooling system with similar open cavity design as in Fig. 6.27, this indicates that cavity steps (cooling fins) are better to be designed with lower heights.

6.5 Conclusions

In this chapter, the newly developed solver ‘nanofluidMixtureFoam’ is applied to the three most popular geometries in nanofluid heat transfer research: two-dimensional square cavity,

three-dimensional horizontal cylinder and two-dimensional channel with an open cavity. For all the geometrical models, the relationships among nanofluid flow, heat transfer and nanoparticles sedimentation are investigated. Furthermore in those cases of two-dimensional channel with an open cavity, the impacts of geometrical design on nanofluid heat transfer and nanoparticles sedimentation features are also discussed. Based on the investigations performed by solver ‘nanofluidMixtureFoam’ in this chapter, it can be concluded that nanoparticles sedimentation deteriorates nanofluid heat transfer performance considerably in both natural and forced convection cases.

7. CONCLUSIONS AND RECOMMENDATIONS

7.1 Summary and conclusions

Nanofluid is increasingly popular to be applied as coolant in practical engineering. To gain a better understanding of nanofluid heat transfer performance, many numerical studies have been carried out recently, in which CFD simulation is considered as one of the most prevail methods at present. However in previous CFD investigations, nanofluids were considered as stable with uniform properties very often. Such assumption may lead to unpractical numerical results for some cases in which nanoparticles sedimentation can be observed in a short period after nanofluid preparation.

To address the above issue, nanoparticles sedimentation is coupled to CFD simulation of nanofluid flow and heat transfer for the first time in this project. For such a target, both Eulerian-Lagrangian and Eulerian-Mixture approaches are discussed in this thesis. Based on the two different ideas, two new numerical solvers ‘nanofluidELFoam’ and ‘nanofluidMixtureFoam’ under OpenFOAM frame are developed. However, because computational resource is usually limited in simulations, comparatively, the Eulerian-Mixture approach is considered as a better option to achieve the balance between computational effort and simulation accuracy in this project. The newly developed solver ‘nanofluidMixtureFoam’ is applied to three typical geometries in nanofluid heat transfer research to investigate the impacts among nanofluid flow, heat transfer and nanoparticles sedimentation.

Based on the works finished in this thesis, some conclusions can be summarised as follows:

1. To address the problem of nanoparticles sedimentation, multi-phase CFD approaches can be carried out under both Eulerian-Lagrangian and Eulerian frames. For these numerical implementations, OpenFOAM is believed to be one of the best tools to develop new solvers particularly to couple nanofluid flow, heat transfer and nanoparticles sedimentation by its ‘open’ nature.
2. Theoretically, CFD simulations in Eulerian-Lagrangian way are believed to be more practical due to its nature of treating nanofluid as a real two-phase suspension with basefluid and suspending particles. In this project, original OpenFOAM solvers

‘buoyantBoussinesqPimpleFoam’ and ‘icoUncoupledKinematicParcelFoam’ are coupled to develop a new Eulerian-Lagrangian solver ‘nanofluidELFoam’. According to the solver test in Section 4.5, it can be seen that nanofluid flow, heat transfer and nanoparticles’ motions can be coupled successfully by solver ‘nanofluidELFoam’. However, in terms of computational time, it is also found that Eulerian-Lagrangian approach is actually not ideal when too many particles need to be tracked.

3. By assuming nanoparticles as a pseudo continuous phase, nanofluid CFD simulation under Eulerian-Eulerian frame is another reasonable option. And a simplified Eulerian-Eulerian version-Mixture model can be applied for even more efficient simulations. With Eulerian-Mixture approach, only one set of governing equations need to be solved for mixture. But at each time step, a nanoparticle sedimentation velocity U_{sm} is needed to correct the whole velocity field of nanofluid mixture. Because U_{sm} can be defined by appropriate experimental investigations particularly for nanofluids, this approach is believed to be practical to address the problem of nanoparticles sedimentation during nanofluid CFD simulation.
4. In this PhD project, a new OpenFOAM solver ‘nanofluidMixtureFoam’ is developed based on Eulerian-Mixture approach. After validating the necessary functions for nanofluid flow, heat transfer and nanoparticles sedimentation respectively, ‘nanofluidMixtureFoam’ is applied to three popular typical geometries in nanofluid investigation: two-dimensional square cavity, three-dimensional horizontal cylinder and two-dimensional channel with an open cavity.
5. In the cases of two-dimensional square cavity, it can be found that nanofluid heat transfer and nanoparticles sedimentation have noticeable impacts to each other. The impacts are increasingly considerable when Rayleigh number Ra is increased.
6. In the cases of three-dimensional horizontal cylinder, although all the numerical phenomena are quite similar to those corresponding two-dimensional square cavity cases, however, some tiny difference still can be observed. More specifically, in three-dimensional horizontal cylinder cases, the maximum vertical velocity of temperature driven flow is slightly lower than that in the corresponding two-dimensional square cavity

cases due to walls friction. This indicates that it is actually not sufficiently practical to compare numerical results of two-dimensional nanofluid natural convection investigations to those experimental investigations which are performed in three-dimension.

7. In the cases of two-dimensional channel with an open cavity heated at bottom, nanoparticles sedimentation is found to deteriorate nanofluid heat transfer performance considerably. If without consideration of nanoparticles sedimentation, pure water should be preferred rather than low volume fraction nanofluid because no considerable different results are found between the cases of water and 0.64% Al_2O_3 /water nanofluid. Furthermore, geometrical design is found to have remarkable impacts on heat transfer performance of the channel. More specifically, greater geometrical ratio h/H induces higher temperature at heating surface (indicating worse heat transfer performance).

7.2 Recommendations and future development

Although the newly developed solvers ‘nanofluidELFoam’ and ‘nanofluidMixtureFoam’ have been verified working well in this project. However, during the simulations carried out by above two solvers, some problems still exist and should be addressed in future. Regarding the present issues, some recommendations can be given for future solver developments as follows:

1. For solver ‘nanofluidELFoam’, it is necessary to give a general criteria for the required amounts of numerical particles in different cases. By this criteria, a proper particle number needed by a certain case should be suggested once the mesh strategy has been confirmed. This will be a very good practice to ensure the amount of numerical particles is big enough to present real nanoparticles in computational region with sufficient confidence.
2. In solver ‘nanofluidMixtureFoam’, it is suggested to compile a new library in which some nanofluid temperature- and volume fraction-dependent properties are stored. This will introduce more experimental information and is believed to be necessary to make nanofluid CFD simulations even more practical.
3. Furthermore, it is also recommended to compile a new library to store nanoparticles sedimentation features under different conditions and can be called by solver

‘nanofluidMixtureFoam’ during simulation. In this thesis, only 0.64% Al_2O_3 /water is simulated because too few published experimental investigations particularly for nanofluid stability can be found.

4. In most practical applications, coolant does not flow over heat source directly for safety reasons. Instead, a certain cooling system is usually attached to the surface of heat source. Thus for engineering application, a solver aiming to simulate heat transfer between heat source, solid cooling system and nanofluid coolant would be required.

REFERENCE

- Abu-Nada, E. 2008. Application of nanofluids for heat transfer enhancement of separated flows encountered in a backward facing step. *International Journal of Heat and Fluid Flow*, 29, 242-249.
- Akbari, M., Galanis, N. & Behzadmehr, A. 2011. Comparative analysis of single and two-phase models for CFD studies of nanofluid heat transfer. *International Journal of Thermal Sciences*, 50, 1343-1354.
- Akbari, M., Galanis, N. & Behzadmehr, A. 2012. Comparative assessment of single and two-phase models for numerical studies of nanofluid turbulent forced convection. *International Journal of Heat and Fluid Flow*, 37, 136-146.
- Akoh, H., Tsukasaki, Y., Yatsuya, S. & Tasaki, A. 1978. Magnetic properties of ferromagnetic ultrafine particles prepared by vacuum evaporation on running oil substrate. *Journal of Crystal Growth*, 45, 495-500.
- Aminossadati, S. M. & Ghasemi, B. 2009. Natural convection cooling of a localised heat source at the bottom of a nanofluid-filled enclosure. *European Journal of Mechanics-B/Fluids*, 28, 630-640.
- Anderson, J. D. & Wendt, J. F. 1995. *Computational fluid dynamics*, Springer.
- Anderson Jr, J. D. 2009. Governing equations of fluid dynamics. *Computational fluid dynamics*. Springer.
- Andersson, J. 2011. Simulation of Wave Induced Forces on Semi Submerged Horizontal Cylinders Using OpenFOAM®.
- Anoop, K. B., Sundararajan, T. & Das, S. K. 2009. Effect of particle size on the convective heat transfer in nanofluid in the developing region. *International Journal of Heat and Mass Transfer*, 52, 2189-2195.
- Antypov, D. & Elliott, J. A. 2011. On an analytical solution for the damped Hertzian spring. *EPL (Europhysics Letters)*, 94, 50004.
- Aris, R. 2012. *Vectors, tensors and the basic equations of fluid mechanics*, Courier Corporation.
- Ashorynejad, H. R., Mohamad, A. A. & Sheikholeslami, M. 2013. Magnetic field effects on natural convection flow of a nanofluid in a horizontal cylindrical annulus using Lattice Boltzmann method. *International Journal of Thermal Sciences*, 64, 240-250.
- Assael, M. J., Metaxa, I. N., Arvanitidis, J., Christofilos, D. & Lioutas, C. 2005. Thermal conductivity enhancement in aqueous suspensions of carbon multi-walled and double-walled nanotubes in the presence of two different dispersants. *International Journal of Thermophysics*, 26, 647-664.
- Auerkari, P. 1996. *Mechanical and physical properties of engineering alumina ceramics*, Technical Research Centre of Finland Finland.
- Bajestan, E. E., Niazmand, H. & Renksizbulut, M. Flow and Heat Transfer of

Nanofluids With Temperature Dependent Properties. ASME 2010 8th International Conference on Nanochannels, Microchannels, and Minichannels collocated with 3rd Joint US-European Fluids Engineering Summer Meeting, 2010. American Society of Mechanical Engineers, 733-739.

Barakos, G., Mitsoulis, E. & Assimacopoulos, D. 1994. Natural convection flow in a square cavity revisited: laminar and turbulent models with wall functions. *International Journal for Numerical Methods in Fluids*, 18, 695-719.

Bathe, K.-J., Brezzi, F. & Pironneau, O. 2001. *Computational fluid and solid mechanics*, Springer.

Behzadmehr, A., Saffar-Avval, M. & Galanis, N. 2007. Prediction of turbulent forced convection of a nanofluid in a tube with uniform heat flux using a two phase approach. *International Journal of Heat and Fluid Flow*, 28, 211-219.

Bejan, A. & Kraus, A. D. 2003. *Heat transfer handbook*, John Wiley & Sons.

Bianco, V., Chiacchio, F., Manca, O. & Nardini, S. 2009. Numerical investigation of nanofluids forced convection in circular tubes. *Applied Thermal Engineering*, 29, 3632-3642.

Boussinesq, J. 1903. *Théorie analytique de la chaleur: mise en harmonie avec la thermodynamique et avec la théorie mécanique de la lumière*, Gauthier-Villars.

Breinlinger, T., Hashibon, A. & Kraft, T. 2015. Simulation of the influence of surface tension on granule morphology during spray drying using a simple capillary force model. *Powder Technology*, 283, 1-8.

Brennan, D. 2001. *The numerical simulation of two phase flows in settling tanks*. Imperial College London (University of London).

Brown, W. K. & Wohletz, K. H. 1995. Derivation of the Weibull distribution based on physical principles and its connection to the Rosin–Rammler and lognormal distributions. *Journal of Applied Physics*, 78, 2758-2763.

Brust, M., Walker, M., Bethell, D., Schiffrin, D. J. & Whyman, R. 1994. Synthesis of thiol-derivatised gold nanoparticles in a two-phase liquid–liquid system. *J. Chem. Soc., Chem. Commun.*, 801-802.

Buongiorno, J., Venerus, D. C., Prabhat, N., McKrell, T., Townsend, J., Christianson, R., Tolmachev, Y. V., Keblinski, P., Hu, L.-w. & Alvarado, J. L. 2009. A benchmark study on the thermal conductivity of nanofluids. *Journal of Applied Physics*, 106, 094312.

Caretto, L. S., Curr, R. M. & Spalding, D. B. 1972. Two numerical methods for three-dimensional boundary layers. *Computer Methods in Applied Mechanics and Engineering*, 1, 39-57.

Cengel, Y. A., Boles, M. A. & Kanoglu, M. 1998. *Thermodynamics: an engineering approach*, McGraw-Hill New York.

ÇEr, B. B., Krane, E. & Andric, J. 2014. CFD WITH OPENSOURCE SOFTWARE.

Chang, H., Tsung, T.-T., Yang, Y. C., Chen, L. C., Lin, H.-M., Lin, C.-K. & Jwo, C.

S. 2005. Nanoparticle suspension preparation using the arc spray nanoparticle synthesis system combined with ultrasonic vibration and rotating electrode. *The international journal of advanced manufacturing technology*, 26, 552-558.

Chen, F. 2009. Coupled flow discrete element method application in granular porous media using open source codes. *Doctoral Dissertations*, 21.

Chen, G., Xiong, Q., Morris, P. J., Paterson, E. G., Sergeev, A. & Wang, Y. 2014a. OpenFOAM for computational fluid dynamics. *Notices of the AMS*, 61, 354-363.

Chen, H.-J. & Wen, D. 2011. Ultrasonic-aided fabrication of gold nanofluids. *Nanoscale research letters*, 6, 1-8.

Chen, L. F., Zang, J., Hillis, A. J., Morgan, G. C. J. & Plummer, A. R. 2014b. Numerical investigation of wave–structure interaction using OpenFOAM. *Ocean Engineering*, 88, 91-109.

Choi, S. U. S. & Eastman, J. A. 2001. Increasing conductivity of liquids with nanofluid by dispersion of particles, evaporation and adding thioglycolic acid. Google Patents.

Choi, S. U. S. 1995. Enhancing thermal conductivity of fluids with nanoparticles. *ASME-Publications-Fed*, 231, 99-106.

Chung, T. J. 2010. *Computational fluid dynamics*, Cambridge university press.

Cooper & Stevenson, P. 1951. A study of the nucleation and growth processes in the synthesis of colloidal gold. *Discussions of the Faraday Society*, 11, 55-75.

Crowe, C. T. 2005. *Multiphase flow handbook*, CRC press.

Cui, W., Shen, Z., Yang, J. & Wu, S. 2015. Effect of chaotic movements of nanoparticles for nanofluid heat transfer augmentation by molecular dynamics simulation. *Applied Thermal Engineering*, 76, 261-271.

Cundall, P. A. & Strack, O. D. L. 1979. A discrete numerical model for granular assemblies. *Geotechnique*, 29, 47-65.

Das, S. K., Choi, S. U. S. & Patel, H. E. 2006. Heat transfer in nanofluids—a review. *Heat transfer engineering*, 27, 3-19.

Das, S. K., Putra, N. & Roetzel, W. 2003a. Pool boiling characteristics of nano-fluids. *International Journal of Heat and Mass Transfer*, 46, 851-862.

Das, S. K., Putra, N., Thiesen, P. & Roetzel, W. 2003b. Temperature dependence of thermal conductivity enhancement for nanofluids. *Journal of Heat Transfer*, 125, 567-574.

Di Stefano, G. 2014. Modeling thermal energy storage systems with OpenFOAM.

Ding, Y. & Wen, D. 2005. Particle migration in a flow of nanoparticle suspensions. *Powder Technology*, 149, 84-92.

Douroumis, D., Fatouros, D. G., Bouropoulos, N., Papagelis, K. & Tasis, D. 2007. Colloidal stability of carbon nanotubes in an aqueous dispersion of phospholipid.

International journal of nanomedicine, 2, 761.

Drew, D., Cheng, L. & Lahey, R. T. 1979. The analysis of virtual mass effects in two-phase flow. *International Journal of Multiphase Flow*, 5, 233-242.

Drzazga, M., Lemanowicz, M., Dzido, G. & Gierczycki, A. 2012. Preparation of metal oxide-water nanofluids by two-step method. *Inżynieria i Aparatura Chemiczna*, 213-215.

Eastman, J. A., Phillpot, S. R., Choi, S. U. S. & Keblinski, P. 2004. Thermal transport in nanofluids 1. *Annu. Rev. Mater. Res.*, 34, 219-246.

Edward, J. T. 1970. Molecular volumes and the Stokes-Einstein equation. *Journal of Chemical Education*, 47, 261.

Elghobashi, S. 1994. On predicting particle-laden turbulent flows. *Applied Scientific Research*, 52, 309-329.

Ferziger, J. H. & Perić, M. 2002. *Computational methods for fluid dynamics*, Springer Berlin.

Fezoui, L. & Stoufflet, B. 1989. A class of implicit upwind schemes for Euler simulations with unstructured meshes. *Journal of Computational Physics*, 84, 174-206.

Garg, R., Galvin, J., Li, T. & Pannala, S. 2012. Open-source MFIX-DEM software for gas–solids flows: Part I—Verification studies. *Powder Technology*, 220, 122-137.

Garoosi, F., Garoosi, S. & Hooman, K. 2014. Numerical simulation of natural convection and mixed convection of the nanofluid in a square cavity using Buongiorno model. *Powder Technology*, 268, 279-292.

Geuzaine, C. & Remacle, J. F. 2009. Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79, 1309-1331.

Ghadimi, A., Saidur, R. & Metselaar, H. S. C. 2011. A review of nanofluid stability properties and characterization in stationary conditions. *International Journal of Heat and Mass Transfer*, 54, 4051-4068.

Glatzel, T., Litterst, C., Cupelli, C., Lindemann, T., Moosmann, C., Niekrawietz, R., Streule, W., Zengerle, R. & Koltay, P. 2008. Computational fluid dynamics (CFD) software tools for microfluidic applications—A case study. *Computers & Fluids*, 37, 218-235.

Gnielinski, V. 1976. New equations for heat and mass-transfer in turbulent pipe and channel flow. *International chemical engineering*, 16, 359-368.

Guide, O. U. 2011. Programmers Guide. *JDT Core.*, retrieved from on Apr, 27, 3.

Haddad, Z., Oztop, H. F., Abu-Nada, E. & Mataoui, A. 2012. A review on natural convective heat transfer of nanofluids. *Renewable and Sustainable Energy Reviews*, 16, 5363-5378.

Haight, F. A. & Haight, F. A. 1967. Handbook of the Poisson distribution.

He, Y., Jin, Y., Chen, H., Ding, Y., Cang, D. & Lu, H. 2007. Heat transfer and flow behaviour of aqueous suspensions of TiO_2 nanoparticles (nanofluids) flowing upward through a vertical pipe. *International Journal of Heat and Mass Transfer*, 50, 2272-2281.

He, Y., Men, Y., Zhao, Y., Lu, H. & Ding, Y. 2009. Numerical investigation into the convective heat transfer of TiO_2 nanofluids flowing through a straight tube under the laminar flow conditions. *Applied Thermal Engineering*, 29, 1965-1972.

Healy, J. J., De Groot, J. J. & Kestin, J. 1976. The theory of the transient hot-wire method for measuring thermal conductivity. *Physica B+ C*, 82, 392-408.

Henderson, A., Ahrens, J. & Law, C. 2004. *The ParaView Guide*, Kitware Clifton Park, NY.

Hertz, H. 1882. Über die Berührung fester elastischer Körper.

Hirt, C. W. & Nichols, B. D. 1981. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of computational physics*, 39, 201-225.

Ho, C.-J., Chen, M. W. & Li, Z. W. 2008. Numerical simulation of natural convection of nanofluid in a square enclosure: effects due to uncertainties of viscosity and thermal conductivity. *International Journal of Heat and Mass Transfer*, 51, 4506-4516.

Ho, C. J., Liu, W. K., Chang, Y. S. & Lin, C. C. 2010. Natural convection heat transfer of alumina-water nanofluid in vertical square enclosures: an experimental study. *International Journal of Thermal Sciences*, 49, 1345-1353.

Hu, Y., He, Y., Wang, S., Wang, Q. & Schlager, H. I. 2014. Experimental and Numerical Investigation on Natural Convection Heat Transfer of TiO_2 -Water Nanofluids in a Square Enclosure. *Journal of Heat Transfer*, 136, 022502.

Hwang, Y., Lee, J.-K., Lee, J.-K., Jeong, Y.-M., Cheong, S.-i., Ahn, Y.-C. & Kim, S. H. 2008. Production and dispersion stability of nanoparticles in nanofluids. *Powder Technology*, 186, 145-153.

Hwang, Y., Lee, J. K., Lee, C. H., Jung, Y. M., Cheong, S. I., Lee, C. G., Ku, B. C. & Jang, S. P. 2007. Stability and thermal conductivity characteristics of nanofluids. *Thermochimica Acta*, 455, 70-74.

Hwang, Y. J., Ahn, Y. C., Shin, H. S., Lee, C. G., Kim, G. T., Park, H. S. & Lee, J. K. 2006. Investigation on characteristics of thermal conductivity enhancement of nanofluids. *Current Applied Physics*, 6, 1068-1071.

Incropera, F. P. 2011. *Fundamentals of heat and mass transfer*, John Wiley & Sons.

Ishii, M. & Grolmes, M. A. 1975. Inception criteria for droplet entrainment in two-phase concurrent film flow. *AIChE Journal*, 21, 308-318.

Issa, R. I., Ahmadi-Befrui, B., Beshay, K. R. & Gosman, A. D. 1991. Solution of the implicitly discretised reacting flow equations by operator-splitting. *Journal of Computational Physics*, 93, 388-410.

Jacquotte, O.-P. & Coussement, G. 1992. Structured mesh adaption: space accuracy

and interpolation methods. *Computer Methods in Applied Mechanics and Engineering*, 101, 397-432.

Jang, S. P. & Choi, S. U. S. 2004. Role of Brownian motion in the enhanced thermal conductivity of nanofluids. *Applied physics letters*, 84, 4316-4318.

Jang, S. P. & Choi, S. U. S. 2006. Cooling performance of a microchannel heat sink with nanofluids. *Applied Thermal Engineering*, 26, 2457-2463.

Jasak, H. 1996. Error analysis and estimation for the finite volume method with applications to fluid flows.

Jasak, H. 2009. OpenFOAM: Open source CFD in research and industry. *International Journal of Naval Architecture and Ocean Engineering*, 1, 89-94.

Jasak, H., Jemcov, A. & Tukovic, Z. 2013. OpenFOAM: A C++ library for complex physics simulations.

Jiang, L., Gao, L. & Sun, J. 2003. Production of aqueous colloidal dispersions of carbon nanotubes. *Journal of Colloid and Interface Science*, 260, 89-94.

Kakac, S., Aung, W. & Viskanta, R. 1985. Natural convection: fundamentals and applications. *Washington, DC, Hemisphere Publishing Corp., 1985, 1191 p. No individual items are abstracted in this volume.*, 1.

Kalteh, M., Abbassi, A., Saffar-Avval, M., Frijns, A., Darhuber, A. & Harting, J. 2012. Experimental and numerical investigation of nanofluid forced convection inside a wide microchannel heat sink. *Applied Thermal Engineering*, 36, 260-268.

Kamyar, A., Saidur, R. & Hasanuzzaman, M. 2012. Application of computational fluid dynamics (CFD) for nanofluids. *International Journal of Heat and Mass Transfer*, 55, 4104-4115.

Kang, S.-W., Wei, W.-C., Tsai, S.-H. & Yang, S.-Y. 2006. Experimental investigation of silver nano-fluid on heat pipe thermal performance. *Applied Thermal Engineering*, 26, 2377-2382.

Kebllinski, P., Phillpot, S. R., Choi, S. U. S. & Eastman, J. A. 2002. Mechanisms of heat flow in suspensions of nano-sized particles (nanofluids). *International journal of heat and mass transfer*, 45, 855-863.

Kempe, T. & Fröhlich, J. 2012. Collision modelling for the interface-resolved simulation of spherical particles in viscous fluids. *Journal of Fluid Mechanics*, 709, 445-489.

Khanafer, K., Vafai, K. & Lightstone, M. 2003. Buoyancy-driven heat transfer enhancement in a two-dimensional enclosure utilizing nanofluids. *International Journal of Heat and Mass Transfer*, 46, 3639-3653.

Kim, H. D., Kim, J. & Kim, M. H. 2007a. Experimental studies on CHF characteristics of nano-fluids at pool boiling. *International journal of multiphase flow*, 33, 691-706.

Kim, S. H., Choi, S. R. & Kim, D. 2007b. Thermal conductivity of metal-oxide

nanofluids: particle size dependence and effect of laser irradiation. *Journal of Heat Transfer*, 129, 298-307.

Kim, S. J., Bang, I. C., Buongiorno, J. & Hu, L. W. 2007c. Study of pool boiling and critical heat flux enhancement in nanofluids. *TECHNICAL SCIENCES*, 55.

Kim, S. J., Bang, I. C., Buongiorno, J. & Hu, L. W. 2007d. Surface wettability change during pool boiling of nanofluids and its effect on critical heat flux. *International Journal of Heat and Mass Transfer*, 50, 4105-4116.

Kole, M. & Dey, T. K. 2010. Thermal conductivity and viscosity of Al₂O₃ nanofluid based on car engine coolant. *Journal of Physics D: Applied Physics*, 43, 315501.

Komarneni, S., Parker, J. C. & Wollenburger, H. J. Nanophase and Nanocomposite Materials II. 1997. MRS.

Kouloulas, K., Sergis, A. & Hardalupas, Y. 2016. Sedimentation in nanofluids during a natural convection experiment. *International Journal of Heat and Mass Transfer*, 101, 1193-1203.

Kussin, J. & Sommerfeld, M. 2002. Experimental studies on particle behaviour and turbulence modification in horizontal channel flow with different wall roughness. *Experiments in Fluids*, 33, 143-159.

Kwak, K. & Kim, C. 2005. Viscosity and thermal conductivity of copper oxide nanofluid dispersed in ethylene glycol. *Korea-Australia Rheology Journal*, 17, 35-40.

Lai, F.-H. & Yang, Y.-T. 2011. Lattice Boltzmann simulation of natural convection heat transfer of Al₂O₃/water nanofluids in a square enclosure. *International Journal of Thermal Sciences*, 50, 1930-1941.

Lee, J. & Mudawar, I. 2007. Assessment of the effectiveness of nanofluids for single-phase and two-phase heat transfer in micro-channels. *International Journal of Heat and Mass Transfer*, 50, 452-463.

Lee, S., Choi, S.-S., Li, S. a. & Eastman, J. A. 1999. Measuring thermal conductivity of fluids containing oxide nanoparticles. *Journal of Heat Transfer*, 121, 280-289.

Li, A. & Ahmadi, G. 1992. Dispersion and deposition of spherical particles from point sources in a turbulent channel flow. *Aerosol science and technology*, 16, 209-226.

Li, C. H. & Peterson, G. P. 2009. Experimental studies of natural convection heat transfer of Al₂O₃/DI water nanoparticle suspensions (nanofluids). *Advances in Mechanical engineering*, 2010.

Li, X. F., Zhu, D. S., Wang, X. J., Wang, N., Gao, J. W. & Li, H. 2008. Thermal conductivity enhancement dependent pH and chemical surfactant for Cu-H₂O nanofluids. *Thermochimica Acta*, 469, 98-103.

Liu, M.-S., Lin, M. C.-C., Huang, I. T. & Wang, C.-C. 2005. Enhancement of thermal conductivity with carbon nanotube for nanofluids. *International Communications in Heat and Mass Transfer*, 32, 1202-1210.

Liu, M.-S., Lin, M. C.-C., Tsai, C. Y. & Wang, C.-C. 2006. Enhancement of thermal

conductivity with Cu for nanofluids using chemical reduction method. *International Journal of Heat and Mass Transfer*, 49, 3028-3033.

Lo, C.-H., Tsung, T.-T. & Chen, L.-C. 2005. Shape-controlled synthesis of Cu-based nanofluid using submerged arc nanoparticle synthesis system (SANSS). *Journal of Crystal Growth*, 277, 636-642.

Lotfi, R., Saboohi, Y. & Rashidi, A. M. 2010. Numerical study of forced convective heat transfer of nanofluids: comparison of different approaches. *International Communications in Heat and Mass Transfer*, 37, 74-78.

Lu, W.-Q. & Fan, Q.-M. 2008. Study for the particle's scale effect on some thermophysical properties of nanofluids by a simplified molecular dynamics method. *Engineering analysis with boundary elements*, 32, 282-289.

Machrafi, H. & Lebon, G. 2016. The role of several heat transfer mechanisms on the enhancement of thermal conductivity in nanofluids. *Continuum Mechanics and Thermodynamics*, 1-15.

Mahdavianesh, M., Noghrehabadi, A. R., Behbahaninejad, M., Ahmadi, G. & Dehghanian, M. 2013. Lagrangian Particle Tracking: Model Development. *Life Science Journal*, 10.

Mahian, O., Kianifar, A., Kalogirou, S. A., Pop, I. & Wongwises, S. 2013. A review of the applications of nanofluids in solar energy. *International Journal of Heat and Mass Transfer*, 57, 582-594.

McNab, G. S. & Meisen, A. 1973. Thermophoresis in liquids. *Journal of Colloid and Interface Science*, 44, 339-346.

Mehrez, Z., Bouterra, M., El Cafsi, A. & Belghith, A. 2013. Heat transfer and entropy generation analysis of nanofluids flow in an open cavity. *Computers & Fluids*, 88, 363-373.

Mintsa, H. A., Roy, G., Nguyen, C. T. & Doucet, D. 2009. New temperature dependent thermal conductivity data for water-based nanofluids. *International Journal of Thermal Sciences*, 48, 363-371.

Mitarai, N. & Nakanishi, H. 2002. Hard Sphere Limit of Soft Sphere Model for Granular Materials. *Phys. Rev. E*, 67, 021301.

Mitchell, M. 2002. Engauge Digitizer. A free open-source software to extract data points from a graph image. Hosted on SourceForge at: <http://digitizer.sourceforge.net>.

Mohammed, H. A., Al-Aswadi, A. A., Shuaib, N. H. & Saidur, R. 2011. Convective heat transfer and fluid flow study over a step using nanofluids: a review. *Renewable and Sustainable Energy Reviews*, 15, 2921-2939.

Mohebbi, A. 2012. Prediction of specific heat and thermal conductivity of nanofluids by a combined equilibrium and non-equilibrium molecular dynamics simulation. *Journal of Molecular Liquids*, 175, 51-58.

Monaghan, J. J. 1992. Smoothed particle hydrodynamics. *Annual review of astronomy*

and astrophysics, 30, 543-574.

Moore, J., Sharma, R., Shih, R., Chase, J., Patel, C. & Ranganathan, P. Going beyond CPUs: The potential of temperature-aware solutions for the data center. Proc. 2004 First Workshop Temperature-Aware Computer Systems (TACS-1) Held in Conjunction with ISCA-31, 2004. Citeseer.

Moraveji, M. K. & Ardehali, R. M. 2013. CFD modeling (comparing single and two-phase approaches) on thermal performance of Al₂O₃/water nanofluid in mini-channel heat sink. *International Communications in Heat and Mass Transfer*, 44, 157-164.

Moraveji, M. K. & Esmaeili, E. 2012. Comparison between single-phase and two-phases CFD modeling of laminar forced convection flow of nanofluids in a circular tube under constant heat flux. *International Communications in Heat and Mass Transfer*, 39, 1297-1302.

Mukherjee, S. & Paria, S. 2013. Preparation and Stability of Nanofluids-A Review. *IOSR Journal of Mechanical and Civil Engineering*, 9, 63-69.

Murshed, S. M. S., Leong, K. C. & Yang, C. 2005. Enhanced thermal conductivity of TiO₂—water based nanofluids. *International Journal of Thermal Sciences*, 44, 367-373.

Murshed, S. M. S., Leong, K. C. & Yang, C. 2008a. Investigations of thermal conductivity and viscosity of nanofluids. *International Journal of Thermal Sciences*, 47, 560-568.

Murshed, S. M. S., Leong, K. C., Yang, C. & Nguyen, N.-T. 2008b. Convective heat transfer characteristics of aqueous TiO₂ nanofluid under laminar flow conditions. *International Journal of Nanoscience*, 7, 325-331.

Nagasaka, Y. & Nagashima, A. 1981. Absolute measurement of the thermal conductivity of electrically conducting liquids by the transient hot-wire method. *Journal of Physics E: Scientific Instruments*, 14, 1435.

Narayan, G. P., Anoop, K. B. & Das, S. K. 2007. Mechanism of enhancement/deterioration of boiling heat transfer using stable nanoparticle suspensions over vertical tubes. *Journal of Applied Physics*, 102, 074317.

Nemati, H., Farhadi, M., Sedighi, K., Fattahi, E. & Darzi, A. A. R. 2010. Lattice Boltzmann simulation of nanofluid in lid-driven cavity. *International Communications in Heat and Mass Transfer*, 37, 1528-1534.

Ni, R., Zhou, S.-Q. & Xia, K.-Q. 2011. An experimental investigation of turbulent thermal convection in water-based alumina nanofluid. *Physics of Fluids (1994-present)*, 23, 022005.

Nie, X. B., Chen, S. Y. & Robbins, M. O. 2004. A continuum and molecular dynamics hybrid method for micro-and nano-fluid flow. *Journal of Fluid Mechanics*, 500, 55-64.

Nnanna, A. G. 2007. Experimental model of temperature-driven nanofluid. *Journal of heat transfer*, 129, 697-704.

- Nsofor, E. C. 2008. Recent patents on nanofluids (Nanoparticles in liquids) heat transfer. *Recent Patents on Mechanical Engineering*, 1, 190-197.
- Open, C. F. D. 2011. Openfoam programmer's guide. *OpenFOAM Foundation*, 2.
- Otanicar, T. P., Phelan, P. E., Prasher, R. S., Rosengarten, G. & Taylor, R. A. 2010. Nanofluid-based direct absorption solar collector. *Journal of renewable and sustainable energy*, 2, 033102.
- Oueslati, F. S. & Bennacer, R. 2011. Heterogeneous nanofluids: natural convection heat transfer enhancement. *Nanoscale research letters*, 6, 1-11.
- Oztop, H. F. & Abu-Nada, E. 2008. Numerical study of natural convection in partially heated rectangular enclosures filled with nanofluids. *International Journal of Heat and Fluid Flow*, 29, 1326-1336.
- Paisarn, N. & Somchai, W. 2011. Experimental study of jet nanofluids impingement system for cooling computer processing unit. *Journal of Electronics Cooling and Thermal Control*, 2011.
- Pallares, J. & Grau, F. X. 2010. Numerical simulation of natural convection of a water-based nanofluid. *Progress in Computational Fluid Dynamics, an International Journal*, 10, 218-224.
- Pang, C., Lee, J. W., Hong, H. & Kang, Y. T. 2014. Heat conduction mechanism in nanofluids. *Journal of Mechanical Science and Technology*, 28, 2925-2936.
- Patankar, S. 1980. *Numerical heat transfer and fluid flow*, CRC Press.
- Peric, M. 1985. *A finite volume method for the prediction of three-dimensional fluid flow in complex ducts*. Imperial College London (University of London).
- Pradeep, T. & Ashokreddy, A. 2012. *A Textbook of Nanoscience and Nanotechnology*, Tata McGraw-Hill Education.
- Putra, N., Roetzel, W. & Das, S. K. 2003. Natural convection of nano-fluids. *Heat and Mass Transfer*, 39, 775-784.
- Raisee, M., Niroobakhsh, Z. & Jafari, A. Hydrodynamic and thermal characteristics of nanofluids in uniformly heated 2-D and axi-symmetric passages. ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, September 5-8, 2006, 2006. Delft University of Technology; European Community on Computational Methods in Applied Sciences (ECCOMAS).
- Ren, H., Chen, X. & Huang, Q. 2012. Numerical simulation of coseismic electromagnetic fields associated with seismic waves due to finite faulting in porous media. *Geophysical Journal International*, 188, 925-944.
- Rhie, C. M. & Chow, W. L. 1983. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA journal*, 21, 1525-1532.
- Roetzel, W., Prinzen, S. & Xuan, Y. 1990. Measurement of thermal diffusivity using temperature oscillations. *Thermal conductivity*, 21, 201-207.

- Saffman, P. G. T. 1965. The lift on a small sphere in a slow shear flow. *Journal of fluid mechanics*, 22, 385-400.
- Saidur, R., Leong, K. Y. & Mohammad, H. A. 2011. A review on applications and challenges of nanofluids. *Renewable and Sustainable Energy Reviews*, 15, 1646-1668.
- Sarkar, S. & Selvam, R. P. 2007. Molecular dynamics simulation of effective thermal conductivity and study of enhanced thermal transport mechanism in nanofluids. *Journal of Applied Physics*, 102, 074302.
- Simmons, G. F. 1985. Calculus with analytic geometry. *AMC*, 10, 12.
- Stokes, G. G. 1851. *On the effect of the internal friction of fluids on the motion of pendulums*, Pitt Press.
- Sudhan, E. P. J. & Meenakshi, K. S. 2011. Synthesis of silver nanofluid by a novel one pot method for heat transfer applications. *Indian Journal of Science and Technology*, 4, 417-421.
- Tahir, S. & Mital, M. 2012. Numerical investigation of laminar nanofluid developing flow and heat transfer in a circular channel. *Applied Thermal Engineering*, 39, 8-14.
- Takács, I., Patry, G. G. & Nolasco, D. 1991. A dynamic model of the clarification-thickening process. *Water research*, 25, 1263-1271.
- Talbot, L., Cheng, R. K., Schefer, R. W. & Willis, D. R. 1980. Thermophoresis of particles in a heated boundary layer. *Journal of Fluid Mechanics*, 101, 737-758.
- Ternik, P. & Rudolf, R. 2012. Heat transfer enhancement for natural convection flow of waterbased nanofluids in a square enclosure. *International Journal of Simulation Modelling*, 11, 29-39.
- Tetsu, F. & Motoo, F. 1976. The dependence of local Nusselt number on Prandtl number in the case of free convection along a vertical surface with uniform heat flux. *International Journal of Heat and Mass Transfer*, 19, 121-122.
- Torii, S. 2009. Turbulent heat transfer behavior of nanofluid in a circular tube heated under constant heat flux. *Advances in Mechanical Engineering*, 2010.
- Tsai, C. Y., Chien, H. T., Ding, P. P., Chan, B., Luh, T. Y. & Chen, P. H. 2004. Effect of structural character of gold nanoparticles in nanofluid on heat pipe thermal performance. *Materials Letters*, 58, 1461-1465.
- Tyagi, H., Phelan, P. & Prasher, R. 2009. Predicted efficiency of a low-temperature nanofluid-based direct absorption solar collector. *Journal of solar energy engineering*, 131, 041004.
- Vaish, M. 2014. Lung-Aerosol Dynamics in Human Airway Models: Validation and Application of OpenFOAM Software.
- Vanka, S. P. 1986. Block-implicit multigrid solution of Navier-Stokes equations in primitive variables. *Journal of Computational Physics*, 65, 138-158.
- Vesilind, P. A. 1968. Design of prototype thickeners from batch settling tests. *Water*

Sewage Works, 115, 302-307.

Wang, X.-j. & Zhu, D.-s. 2009. Investigation of pH and SDBS on enhancement of thermal conductivity in nanofluids. *Chemical Physics Letters*, 470, 107-111.

Wang, X.-Q. & Mujumdar, A. S. 2007. Heat transfer characteristics of nanofluids: a review. *International journal of thermal sciences*, 46, 1-19.

Wang, X.-Q. & Mujumdar, A. S. 2008a. A review on nanofluids-part I: theoretical and numerical investigations. *Brazilian Journal of Chemical Engineering*, 25, 613-630.

Wang, X.-Q. & Mujumdar, A. S. 2008b. A review on nanofluids-Part II: experiments and applications. *Brazilian Journal of Chemical Engineering*, 25, 631-648.

Wen, D., Corr, M., Hu, X. & Lin, G. 2011. Boiling heat transfer of nanofluids: the effect of heating surface modification. *International Journal of Thermal Sciences*, 50, 480-485.

Wen, D. & Ding, Y. 2004. Experimental investigation into convective heat transfer of nanofluids at the entrance region under laminar flow conditions. *International journal of heat and mass transfer*, 47, 5181-5188.

Wen, D. & Ding, Y. 2005a. Experimental investigation into the pool boiling heat transfer of aqueous based γ -alumina nanofluids. *Journal of Nanoparticle Research*, 7, 265-274.

Wen, D. & Ding, Y. 2005b. Formulation of nanofluids for natural convective heat transfer applications. *International Journal of Heat and Fluid Flow*, 26, 855-864.

Wen, D., Lin, G., Vafaei, S. & Zhang, K. 2009. Review of nanofluids for heat transfer applications. *Particuology*, 7, 141-150.

White, F. M. & Corfield, I. 2006. *Viscous fluid flow*, McGraw-Hill New York.

Witharana, S., Hodges, C., Xu, D., Lai, X. & Ding, Y. 2012. Aggregation and settling in aqueous polydisperse alumina nanoparticle suspensions. *Journal of Nanoparticle Research*, 14, 1-11.

Witharana, S., Palabiyik, I., Musina, Z. & Ding, Y. 2013. Stability of glycol nanofluids—the theory and experiment. *Powder Technology*, 239, 72-77.

Xie, H., Wang, J., Xi, T., Liu, Y., Ai, F. & Wu, Q. 2002. Thermal conductivity enhancement of suspensions containing nanosized alumina particles. *Journal of Applied Physics*, 91, 4568-4572.

Xu, B. H. & Yu, A. B. 1997. Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics. *Chemical Engineering Science*, 52, 2785-2809.

Xuan, Y. & Li, Q. 2000. Heat transfer enhancement of nanofluids. *International Journal of Heat and Fluid Flow*, 21, 58-64.

Xuan, Y. & Yao, Z. 2005. Lattice Boltzmann model for nanofluids. *Heat and mass transfer*, 41, 199-205.

Xue, Q.-Z. 2003. Model for effective thermal conductivity of nanofluids. *Physics letters A*, 307, 313-317.

Yang, Y., Zhang, Z. G., Grulke, E. A., Anderson, W. B. & Wu, G. 2005. Heat transfer properties of nanoparticle-in-fluid dispersions (nanofluids) in laminar flow. *International Journal of Heat and Mass Transfer*, 48, 1107-1116.

Yt Feng, P. X. L. P. Y. T., Professor Shunying Ji, P., Zhang, W.-J., Ruan, X.-H., He, G.-H., Ma, Y.-L. & Liu, Y.-F. 2015. Particles deposition on microfiltration permeable boundary: Modeling and simulation with two-ways coupling sphere model in a turbulent flow. *Engineering Computations*, 32, 1135-1152.

Yu, W. & Choi, S. U. S. 2003. The role of interfacial layers in the enhanced thermal conductivity of nanofluids: a renovated Maxwell model. *Journal of Nanoparticle Research*, 5, 167-171.

Yu, W. & Choi, S. U. S. 2004. The role of interfacial layers in the enhanced thermal conductivity of nanofluids: a renovated Hamilton–Crosser model. *Journal of Nanoparticle Research*, 6, 355-361.

Yu, W., France, D. M., Choi, S. U. S. & Routbort, J. L. 2007. Review and assessment of nanofluid technology for transportation and other applications. Argonne National Laboratory (ANL).

Yu, W., Xie, H., Chen, L. & Li, Y. 2010. Enhancement of thermal conductivity of kerosene-based Fe₃O₄ nanofluids prepared via phase-transfer method. *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, 355, 109-113.

Zhong, W., Xiong, Y., Yuan, Z. & Zhang, M. 2006. DEM simulation of gas–solid flow behaviors in spout-fluid bed. *Chemical Engineering Science*, 61, 1571-1584.