TOWARDS MODELING HEAT TRANSFER USING A LATTICE BOLTZMANN METHOD FOR POROUS MEDIA

By

Olimpia Banete

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Applied Science (MASc)

The School of Graduate Studies
Laurentian University
Sudbury, Ontario, Canada

THESIS DEFENCE COMMITTEE/COMITÉ DE SOUTENANCE DE THÈSE

Laurentian Université/Université Laurentienne
School of Graduate Studies/École des études supérieures

Title of Thesis Titre de la thèse	TOWARDS MODELING HEAT TRANSFER USING A LATTICE BOLTZMANN METHOD FOR POROUS MEDIA
Name of Candidate Nom du candidat	Banete, Olimpia
Degree Diplôme	Master of Applied Science

Department/Program Département/Programme	Natural Resources Engineering	Date of Defence Date de la soutenance	April 25, 2014

APPROVED/APPROUVÉ

Thesis Examiners/Examinateurs de thèse:

Dr. Dean Millar
(Supervisor/Directeur de thèse)

Dr. Lorrie Fava
(Committee member/Membre du comité)

Dr. Junfeng Zhang
(Committee member/Membre du comité)

Dr. Osman Abou-Rabia
(Internal Examiner/Examinateur interne)

Approved for the School of Graduate Studies
Approuvé pour l'École des études supérieures
Dr. David Lesbarrères
M. David Lesbarrères
Director, School of Graduate Studies
Directeur, École des études supérieures

ACCESSIBILITY CLAUSE AND PERMISSION TO USE

Abstract

I present in this thesis a fluid flow and heat transfer model for porous media using the lattice Boltzmann method (LBM). A computer simulation of this process has been developed and it is written using MATLAB software. The simulation code is based on a two dimensional model, D2Q9. Three physical experiments were designed to prove the simulation model through comparision with numerical results. In the experiments, physical properties of the air flow and the porous media were used as input for the computer model. The study results are not conclusive but show that the LBM model may become a reliable tool for the simulation of natural convection heat transfer in porous media.

Simulations leading to improved understanding of the processes of air flow and heat transfer in porous media may be important into improving the efficiency of methods of air heating or cooling by passing air through fragmented rock.

Acknowledgements

I would like to take this opportunity to thank my supervisor, Dr. Dean Millar for his continuous support throughout my Master of Applied Science program. He provided me with valuable guidance at every stage of the writing of this thesis and conducting the work. Without his constant encouragement, helpful suggestions and impressive patience, I would not have finished this thesis. I am also grateful to him for providing some financial support during the time of the thesis work, through the SUMIT 6 Research Fund, Research Project, funded by the Ontario Research Fund.

I am thankful to Dr. Junfeng Zhang for all explanations and advice concerning Lattice Boltzmann method.

Special thanks go to Adam Turcotte, who is most responsible for helping me implement the simulation model in MATLAB as well as completing the output analysis.

I would like to express my gratitude to Dr. Ramesh Subramanian for the great support during the educational journey to Laurentian University.

I thank to Dr. Zhaoli Guo (Huazhong University, China) for replying to my emails on questions related to Lattice Boltzmann method.

Finally I would like to acknowledge MIRARCO team for their kind support.

CONTENT

Table of Contents

List of Figures

List of Table

Nomenclature

c	particle speed [m/s]
c_s	speed of sound for fluid [m/s]
c_p	specific heat capacity [$kJ/kg.K$]
c_{ps}	solid specific heats at constant pressure [$kJ/kg.K$]
c_{pf}	fluid specific heats at constant pressure [$kJ/kg.K$]
$\mathbf{e}_a$	particle velocity
a	discrete velocity direction
f_a	density distribution function
f_a^{eq}	equilibrium distribution function of f_a
g_a	energy distribution function
g_a^{eq}	equilibrium distribution function of g_a
$\mathbf{F}$	body force
g	acceleration of gravity [m/s^2]
$\mathbf{u}$	fluid macroscopic velocity [m/s]
$\mathbf{u}_P$	velocity in physical units [m/s]
$\mathbf{u}_{LB}$	velocity in lattice units
$\mathbf{x}$	position

t	time [s]
k	coefficient of thermal conductivity of the fluid [W/(mKg)]
k_m	effective thermal conductivity [W/(mKg)]
k_e	stagnant thermal conductivity [W/(mKg)]
k_d	thermal conductivity due to thermal dispersion [W/(mKg)]]
K	Permeability [m^2]
p	pressure [Pa]
T	temperature [K]
T_0	reference temperature [K]
a	acceleration due to other external force fields [m/s^2]
q	volumetric fluid flow through the medium
Da	Darcy number
Nu	Nusselt number
Pe	Peclet number
Ra	Raynolds number
Re	Reynolds number
Pr	Prandtl number
M	Mach number
N	number of grid nodes

Greek symbols

ρ	density [kg/m^3]
ρ_s	solid densities [kg/m^3]
ρ_f	fluid densities [kg/m^3]
α	thermal diffusivity [m^2/s]
α_m	effective thermal diffusivity [m^2/s]
δ_t	time step
δ_x	lattice spacing
β	thermal expansion coefficient [1/K]
τ_v	relaxation time for f_a
τ_c	relaxation time for g_a
μ	dynamic viscosity [Pa $\cdot$ s) (N $\cdot$ s/m^2 or kg/(m $\cdot$ s)]
ν	kinematic viscosity [m^2/s]
ε	porosity of the porous medium
w_a	weight for the specific model
σ	ratio between the heat capacities of the solid and fluid phases
λ	expansion parameter [m]

List of Abbreviation

FVM - Finite Volume Method

LBM - Lattice Boltzmann Model

LGA - Lattice Gas Automata

BGK - Bhatnagar Gross Krook

REV - Representative Elementary Volume

LBE - Lattice Boltzmann Equation

GLBE - Generalized Lattice Boltzmann Equation

DF - Distribution Function

DDF - Double Distribution Function

EDF - Equilibrium Distribution Function

IEDDF - Internal Energy Density Distribution Function

MS - Multispeed Models

SIS - Sequential Indicator Simulation

TDF - Temperature Distribution Function

CFD - Computational Fluid Dynamics

TLBM - Thermal Lattice Boltzmann Methods

CIP - Cubic Interpolated Pseudo-particle

PDE - Partial Differential Equation

NSE - Navier-Stokes Equation

FDM - Finite Difference Method

FEM - Finite Element Method

SRT - Single Relaxation Time

SOR - Successive Over Relaxation Method

MRT - Multiple Relaxation Times

GPU - Graphics Processing Unit

FCC - Face Centered Cubic

CCP - Cubic Close Packed

HCP - Hexagonal Close Packed

1 Introduction

In the last few decades the problem of fluid flow and convection heat transfer in porous media have been studied by many researchers in different fields of science and engineering , including civil and mechanical engineering, chemical and petroleum engineering, hydrology, and thermal management of electronic cooling. The LBM is widely used in many types of application: aerodynamic optimization, computation of aeroacoustic sources, and thermal management simulations. Fluid flow and heat transfer through a porous medium is a common phenomenon in nature and is a subject of interest that has become a separate field of research. Nield and Bejan (1992) have given a detailed description of the subject. The LBM is a new relatively computational technique for simulating fluid flows and modelling complex physics in fluids. In this method the fluid is modelled by particles that move on a regular lattice. The particles propagate to their neighbouring lattice points and redistribute their momenta in the succeeding collision. Chen and Doolen (1998) have applied this method to simulation of fluid flows. In the present thesis, a simulation for velocity and thermal flow through porous media using lattice the Boltzmann method is presented.

1.1 Review of Alternatives to Lattice Boltzman Method

In any application that involves gas flow, liquid flow or heat transfer, fluid dynamics simulation can deliver engineering improvement or enhanced scientific understanding, in comparison to the time before their development. Computers are used to perform the calculations required to simulate the interaction of liquids and gases with surfaces and other defined boundary conditions. There are many software implementations, based on different numerical methods

available on the market or freely for scientific use for solving a variety of problems of engineering or scientific interest.

Computational Fluid Dynamics (CFD) is defined as the set of methodologies which use computers to solve simulation of fluid flows numerically. Simulation involves solving the general conservation laws that govern the fluids, that is the conservation of mass, the conservation of momentum and the conservation of energy in the macroscopic world (Milne-Thomson, L. M., 1973).

The finite element method is used in structural analysis of solids (Zienkiewicz, 1967; Yang, 1986; Zienkiewicz *et al*., 2005), but is also applicable to fluids (Lewis *et al*., 2004). This is a common approach used in CFD, especially for large problems and high Reynolds number turbulent flows (Coupez *et al.*, 2010).

ANSYS Fluent (ANSYS Fluent, 2011) is computational fluid dynamics software that contains physical modeling capabilities to model flow, turbulence, heat transfer, and reactions for industrial applications.

COMSOL (COMSOL, 2014) is simulation software based on the finite element method which performs computational fluid dynamics simulations including laminar flow, swirl flow, turbulent flow, non-isothermal flow, high Mach number flow, two-phase flow, and fluid-structure interaction.

FLOW-3D (FLOW-3D, 2009) software, again based on the finite element method, has been able to simulate natural and forced convection, conduction in fluids and solids and solid-solid heat transfer.

SolidWorks also provides a tool based on the finite element method able to simulate liquid and gas flow in real world conditions (Computer Aided, 2014), and to run “what if” scenarios, and to analyze the effects of fluid flow, heat transfer, and related forces on immersed or surrounding components.

Figure 1-1 presents a simulation of fluid flows in a micro-gas turbine using SolidWorks software (Cadorin *et al*., 2012).

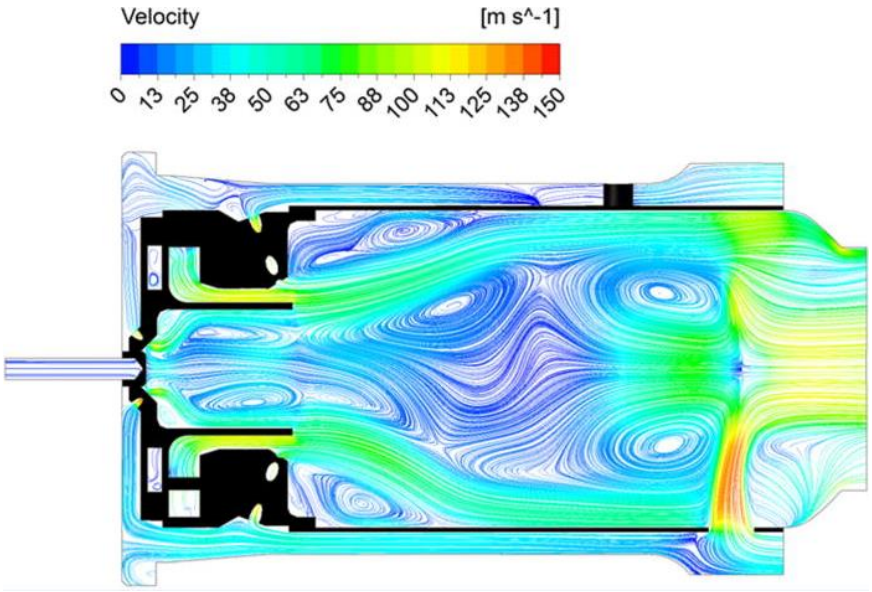


Figure 1-1 Fluid flow simulation in micro-gas turbine (Cadorin et al., 2012)

The finite difference method is a numerical method for approximating the solutions to differential equations using finite difference equations to approximate derivatives, and is simple to program. It is currently only used in few specialized codes, which handle complex geometry with high accuracy and efficiency by using embedded boundaries or overlapping grid.

All these methods can be used for modeling fluid flows and heat transfer for steady boundary conditions and no phase changes. The theoretical formulations of the different software highlighted above are typically the same, as they are all based on the same physical conservation

laws. The software products (for the ones listed are all commercially available, sophisticated tools) differ in the way the implementation discretizes the modelled domain and the mathematical numerical methods used to solve the equations. Some software will have advantages over others in solution effiencency, or in the constitutive behaviour that is accommodated in their material libraries. Some, for example, Solidworks Flow Simulation, has capability to model very large problems, solving them in reasonable time on moderately specified computers. This capability arises as a result of appreciable programming effort by the software authors.

But, given that all these valuable computational fluid dynamics tools are available, what is it about the LBM and its alternative, kinetic theory of gas formulation, that makes it specifically of interest in this thesis?

The context of this work is a concerted effort to model heat and fluid flow in fractured rock materials that are known to behave as large scale (~1km^3) seasonal heat storage systems. The use of 'standard' CFD tools by the research team has already led to significant improvements of understanding of how these systems work (Fava *et al.* 2012), but has also led to the exposure of a significant weakeness in virtually all these commercial formulations: they do not readily accommodate material phase transitions from the fluid phase to the solid phase, as would occur with the cooling of liquid water to ice. The established CFD tools can deal with the problem, but this involves interrupting the computations, remeshing the modelled domain to accommodate the new solid regions, re-initialization with the previous results and restarting the computations-until the next ice crystal is created.

The attractive aspect of the LBM is that such solidification processes can be accommodated 'on-

the–fly', that is, without stopping the computations. Although the LBM is a more recently developed approach to the simulation of fluid systems, it has been improved appreciably over the last decade such that now, large problems can be tackled with the method. It is this ease of dealing with phase transitions with the LBM which motivates its adoption as a topic of study in this thesis.

1.2 Problem Statement

To simulate the velocity and thermal flow through porous media using the lattice Boltzmann method.

1.3 Objectives

- To investigate the fluid flow and heat exchange behaviour for different porosity, temperature and velocity using lattice Boltzmann method.
- To understand how fluid simulation works and how more advanced concepts can be added to it so we can have more realistic representations of real fluids.

1.4 Thesis Overview

Chapter 1 is a brief introduction to the lattice Boltzmann method used to simulate flow and heat transfer in porous media.

Chapter 2 describes the concept of fluid and basic flow relations. Also there describes the principal dimensionless numbers and fundamental concepts of fluids used in this thesis.

Chapter 3 concerns the kinetic theory of gases, which considers the microscopic behaviour of molecules, and interactions that lead to macroscopic relationships and the bridge between microscopic and macroscopic levels of description. Also, it gives theoretical information about the LBM. The lattice Boltzmann equation (LBE) describes the evolution of the particle distribution function. The lattice Boltzmann models (LBMs) for two and three dimensions are also presented. As well, both the LBE for the velocity field and LBE for the temperature field as they are coupled for use in porous media are reviewed.

Chapter 4 provides explanations about the streaming step and the collision step. In addition, discussions of described different LBM boundary conditions, used in the current work, are reported.

Chapter 5 describes methodology and results for three physical model experiments, which were designed and executed.

Chapter 6 includes a description of the simulations of three models written in MATLAB software.

Chapter 7 presents a discussion of how the calculations are made, how the code can be used in other applications, and conclusions.

2 Properties of Fluids and Dimensionless Numbers

The purpose of this literature review is to summarise notions used in fluid dynamics for simulation of fluid flow and temperature through the porous media. The review covers the description of the concept of fluid flow and the main dimensionless numbers.

2.1 Fluid Flow and Heat Transfer in Porous Media

The problem of fluid flow and heat transfer through porous media has preoccupied physicists and engineers for last four decades. The physics of this apparently simple process is, surprisingly, poorly understood. Heat transfer in porous media is becoming of increasingly important in heat exchanger analyses and design.

A porous medium is a material containing (interconnected) voids/pores, which can contain a fluid (liquid or gas). The skeletal portion of the material is often called the "matrix" or "frame" which is usually a solid; however, in some cases structures like foams could be considered a porous medium. The solid matrix is assumed to be rigid for a matrix in which the pores are stationary and in a single-phase fluid. Frequently, both the solid matrix and the pore network are continuous, having a sponge-like structure. The porous medium has two important macroscopic parameters which are influenced by the pore structure. One of them is permeability, which refers to the measure of the ability of a material (rocks) to transmit fluids. This is largely determined by the pore geometry (how large the pores are, and how well connected they are) and is independent of the properties of the penetrating fluid in the condition that there are not meltability effect of

the material. Another important macroscopic parameter of the porous medium is the porosity which represents the amount of void space. The porosity of the porous medium is defined as:

$$\phi = \frac{void\ volume contained\ in\ porous medium\ sample}{total\ volume of\ porous medium\ sample}$$

In general the porous structure of the medium and the fluid flow are very complex and accurate analytical solutions are difficult to obtain.

Many natural materials such as rocks, soil, biological tissues, and manmade materials such as cements and ceramics can be considered as porous media. In fact, many of their important properties can only be investigated by considering of them as porous. The concept of porous media is used in many areas of applied science and engineering: filtration, mechanics (acoustics, geomechanics, soil mechanics, rock mechanics); engineering (petroleum engineering, bio-remediation, construction engineering); geosciences (hydrogeology, petroleum geology, geophysics); biology and biophysics; material science, etc.

The Figure 2-1 shows the connection of pores (Introduction to hydrology, 2010):

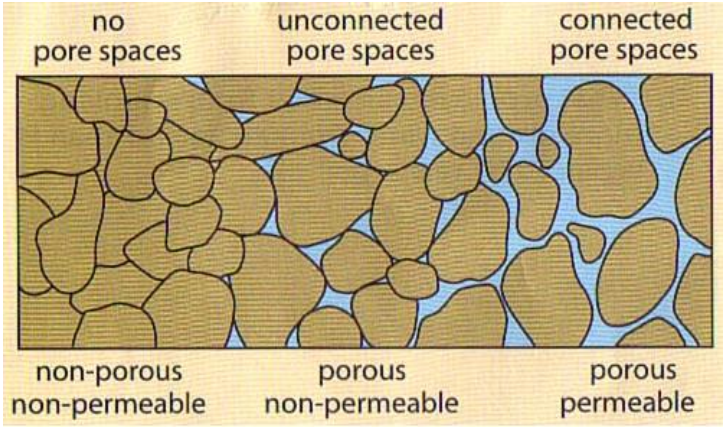


Figure 2-1 Porosity (Introduction to hydrology, 2010)

2.2 Concept of Fluid

A fluid can be defined as a substance which can deform continuously when being subjected to shear stress at any magnitude – it can flow continuously as a result of shearing action. Examples of typical fluids used in engineering applications are water, oil, and air. *Fluid dynamics* (fluid mechanics) is the study of moving (deformable) matter, and includes liquids, gases, and plasmas. In a general sense, a solid can resist shear stress by a static deformation, but a fluid cannot.

2.2.1 Fluid as continuum

In fluids, the molecules can move freely but are constrained through a traction force called *cohesion.* This force is interchangeable from one molecule to another.

Gases consist of molecules that have a very large mean distance between them, compared with their molecular size. Gases have negligible cohesive force and are free to expand until encountering confining walls. The molecules move almost freely relative to each other, and in a given control volume, the number of molecules changes continuously.

A liquid has strong cohesive forces, sufficient to hold the molecules together so that they can withstand higher comparison than gases, and which makes them suitable for fluid power applications. On the surface of the liquid, the cohesion forms a resultant force directed into the liquid region, and the combination of cohesion forces between adjacent molecules forms a tensioned membrane known as a *free surface*. Free-surface flows are dominated by gravitational effects, temperature, solute concentration, etc.

Gases cannot form a free-surface, and thus gas flows are rarely concerned with gravitational effects other than buoyancy. Gases' density has no precise meaning unless the control volume is large in comparison with the intermolecular spacing. If the volume is too large, there could be variation in the bulk density of the molecules due to other effects such as sound transmission. A reasonable value for such a volume is thus about $10^{-9}mm^3$ for all liquids and gases at atmospheric pressure (White, 1994). However, many problems have physical dimensions much larger than this size so that they display fluid properties with practically continuous spatial variation. The fluid can be called a continuum, with well-defined derivatives of the significant variables which define its dynamics.

The equilibrium of a continuum fluid is produced and maintained by collisions between molecules, over a characteristic time scale τ. In classical fluids, τ is of the order of 10^{-10} to 10^{-14} seconds (Chaikin *et al*., 1995). In the physical distance between collisions of the molecules, the mean free path λ_{mfp} is the related length scale. The equilibrium state is not homogeneous in a system with macroscopic motion and disturbances vary in time and length scales set by τ and λ_{mfp}. The macroscopic variables of the system have slow temporal variations and for these are a large amount of equilibrating collisions, where dynamical disturbances are small in space at all times.

For conventional fluid dynamics when the above conditions are satisfied, the Knudsen number Kn should be evaluated for the problem. The Knudsen number is the ratio of the molecular mean free path λ_{mfp} to a representative physical length scale λ_0 of the obstacles and flow channels:

$$Kn = \frac{\lambda_{mfp}}{\lambda_0} \tag{2.1}$$

2.2.2 Basic Flow relation

In fluid dynamics there are three levels to describe the motion of fluid: microscopic level, mesoscopic level, and macroscopic level. On each level, there are the corresponding models to represent the fluid flow. Models should satisfy the general conservation laws, i.e., conservation of mass, conservation of momentum and conservation of energy in the macroscopic world, in which the corresponding macroscopic variables (velocity, pressure and temperature) could be measured by using various kinds of sensor.

On a microscopic scale, fluid contains individual molecules and their physical property (density, velocity, etc.) is non-uniform. The phenomena studied in fluid dynamics are macroscopic and treat the fluid as continuum by seeing it at a coarse enough scale that any "small" fluid element still has very many molecules. One can then assign a local bulk flow velocity $\mathbf{u}(\mathbf{x},t)$ to the element at a point $\mathbf{x}$ in the fluid, and a locally averaged density. These locally averaged quantities then vary smoothly with $\mathbf{x}$ on the macroscopic scale of the flow.

The motion of fluid is described by the basic hydrodynamic equations, the continuity equation:

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{2.2}$$

This expresses the conservation of mass, and the momentum equation:

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p + \nabla \cdot \tau + \rho \mathbf{g} \tag{2.3}$$

which expresses the conservation of momentum (White, 1994), where:

- ρ is the fluid density
- $\mathbf{u}$ is the fluid velocity
- p is the hydrostatic pressure
- τ is the fluid stress tensor
- $\mathbf{g}$ is the acceleration due to external forces including $e.g.$ the effect of gravity on the fluid.

The equation for energy conservation can be written as (White, 1994):

$$\rho \frac{de}{dt} + p(\nabla \cdot \mathbf{u}) = \nabla \cdot (k \nabla T) + \Phi \tag{2.4}$$

where:

- Φ is a viscous dissipation function
- T is the temperature
- k is the coefficient of thermal conductivity of the fluid
- $e = e(p,T)$ is the density of thermal energy and often approximated such that $de \approx c_v dT$, where c_v is the specific heat at constant volume.

In Newtonian fluid viscous stresses are directly proportional to velocity derivatives. $\tau_{ij} = \partial_i u_j$, and for them Eq. (2.3) reduces to the Navier-Stokes equation:

$$\partial_t(\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p + \mu \nabla^2 \mathbf{u} + \rho \mathbf{g} \tag{2.5}$$

where μ is dynamic viscosity of the fluid.

For incompressible fluid fluids this equation can also be written as:

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{\mathbf{1}}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{g} \tag{2.6}$$

where $\nu = \mu / \rho$ is the kinematic viscosity of the fluid. Continuity and momentum equations are usually independent of T and they can be solved separately from the energy equation. Eq. (2.5) contains three second-order nonlinear partial differential equations for four unknown variables, i.e. pressure and three components of velocity, to be solved in space and time. The system is closed by combining it with Eq. (2.2) and using appropriate boundary conditions, such as known pressure or velocity at the inlet and outlet, or on the surfaces. At solid-fluid interfaces the called no-slip boundary conditions characteristic of viscous fluid flows applies. Thus at a solid wall the fluid velocity is zero.

For fluid-flow equations the simplest theory is frictionless flow, and Eq. (2.5) reduces to the Euler equation:

$$\mathbf{g} - \frac{1}{\rho}\nabla p = \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} \tag{2.7}$$

White (1994) used that in a gravitational field $\mathbf{g} = g\mathbf{e}_z$. Integrating Eq. (2.7) along a streamline the Bernoulli equation for incompressible steady flow results:

$$p + \frac{1}{2}\rho u^2 + \rho g z = C \tag{2.8}$$

where C is a constant for each streamline, $\mathbf{g}$ is the gravitational acceleration and z is the elevation. The Bernoulli equation is related to the steady-flow energy equation and is usually used, but has many limits. For example, on streamline no shaft work or heat transfer is accepted (White, 1994).

For stationary flow with very low inertial forces, the left-hand side of Eq. (2.5) is negligible, and it defines the Stokes equation:

$$\nabla p - \rho \mathbf{g} = \mu \nabla^2 \mathbf{u} \tag{2.9}$$

This equation is very important in theoretical and experimental work for fluid flows in porous media, where fluid velocities are low. In the flow systems defined by the Stokes equation, the pressure drop is directly proportional to the fluid velocity. If the force is reversed, the streamlines remain unchanged.

Many characteristics: permeability (μ), viscosity (ν), momentum (p), and dimensionless numbers are important for fluid dynamic investigations, such as: Reynolds number (Re), Rayleigh number (Ra), Prandtl number (Pr), Peclet number (Pe), Nusselt number (Nu), Mach number (M), etc. It can be advantageous to express the relevant quantities and equations in dimensionless form using such dimensionless numbers. For example, the dimensionless form of the Navier-Stokes equation Eq. (2.5) indicated by White (1994) is:

$$\partial_t \cdot \mathbf{u}^* + \left(\mathbf{u}^* \cdot \nabla^*\right)\mathbf{u}^* = -\nabla^* p^* + \frac{1}{\mathrm{Re}} \nabla^{*2} \mathbf{u}^* \tag{2.10}$$

The notation ($*$) is for dimensionless, $\mathbf{u}$ is the flow velocity, p is the pressure.

2.3 Permeability

2.3.1 Darcy's Law

At low Reynolds numbers (the flow is laminar), the most important relation describing fluid transport through porous media is Darcy's law (Scheidegger, 1957):

$$\mathbf{q} = -\frac{K}{\mu} \nabla p \tag{2.11}$$

where:

- $\mathbf{q}$ is the volumetric fluid flow through the medium

- K is the permeability coefficient that measures the conductivity to fluid flow of the porous media
- μ is the viscosity of the fluid
- p is the fluid pressure

Darcy's law was introduced originally as an empirical relationship based on experiments on steady flow in a vertical homogeneous sand filter (Darcy, 1856), and Eq. (2.11) has been found to work very well with a wide variety of natural porous media from sand to granite rocks (Sahimi, 1995; Darcy, 1856 and Scheidegger, 1957). Among the most important such material properties are the porosity ε and the specific surface area S_0 (i.e. the pore surface area in a unit volume of the solid material) that influences the permeability greatly.

Koponen *et al.* (1997) concluded that the pore volume of the material can include non-percolating pores and also some other non-conducting parts, which should reduce the geometrical porosity to effective porosity. Usually, the difference between these two porosities becomes more pronounced near the percolation threshold, where the medium becomes completely blocked.

The Darcy number is a dimensionless number used in the dynamics of flow through porous media and it is found from the differential form of Darcy's Law:

$$Da = \frac{K}{L^2} \tag{2.12}$$

where:

- K is the permeability of the medium [m^2]
- L is the length [m]

2.3.2 Expressions for Permeability

In theoretical and experimental work on fluid flow in porous media, attempts are usually made to find practical correlations between the permeability and some other macroscopic properties of the porous medium. The most-used expressions are the Kozeny law and Kozeny-Carman law that relate permeability to the structural characteristics of the medium and have been derived analytically for capillary tube models (Scheidegger, 1957; Bear, 1972 and Carman, 1937).

Samarasinghe *et al*. (1982) used the Kozeny law, which expresses the permeability of the medium:

$$K = \frac{\varepsilon^3}{c\tau^2 S^2} \tag{2.13}$$

where:

- ε is the porosity
- c is Kozeny constant that depends on the geometry of the porous media
- τ is the tortuosity
- S is the pore surface area per unit volume

For cylindrical capillaries $c = 2$. Sometimes the tortuosity τ is included in the Kozeny constant, and the Kozeny law becomes:

$$K = \frac{\varepsilon^3}{cS^2} \tag{2.14}$$

Using in terms of the specific surface area $S_0 = S/(1-\varepsilon)$, Eq. (2.13) can be written as the Kozeny-Carman law:

$$K = \frac{1}{cS_0^2}\frac{\varepsilon^3}{(1-\varepsilon)^2} \tag{2.15}$$

Equations (2.13) or (2.14) are the most-used expressions for the permeability of porous media. In considering flow through a porous medium, only the interconnected pores are of interest. If the pores are not connected to the main void space, they do not contribute to the flow. The term 'porosity' is then used to describe the interconnected pore space only. The 'dead-end' pores are another type of pore that contributes very little to the flow.

Fig. 2-2 shows the connected pores which give the permeability (MPG Petroleum Inc., 2003):

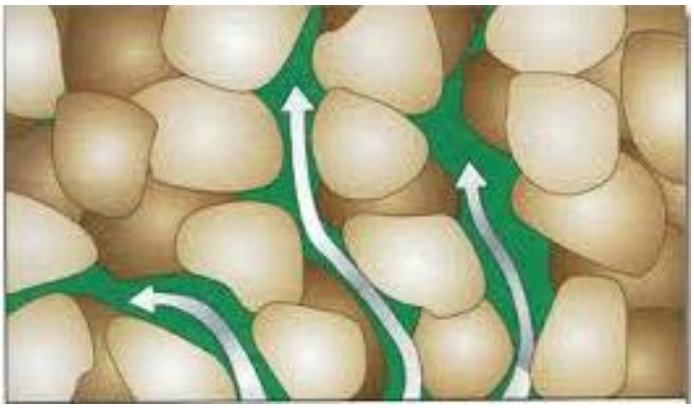


Figure 2-2 Permeability (MPG Petroleum Inc., 2003)

2.3.3 Tortuosity

Another useful characteristic of porous media is tortuosity τ, which is a measure of the complexity of the flow paths. The behaviour of the fluid that flows in a porous medium can be microscopically very complicated. Tortuosity is defined in many ways. One of them is for the case of capillary models, in Eq. (2.13), tortuosity is introduced as the relation $\tau = 1/\cos\theta$, where θ is the angle formed by the capillaries with the normal of the surface of the material. Bear (1972) and Dullien (1979) noted that the other way is in terms of the tube length L_e and the thickness of the medium L, tortuosity is $\tau = L_e / L$ or same authors prefer as $\tau = \left(L_e / L\right)^2$, or as inverses of these two definitions.

Koponen *et al.* (1996) concluded that in principal the tortuosity can be defined as the ratio of a (properly weighted) average length of microscopic flow paths to the length of the system in the direction of the macroscopic flux. A numerical estimate of tortuosity can be conveniently determined from:

$$\tau \equiv \frac{\langle |\mathbf{v}| \rangle}{\langle v_x \rangle} \tag{2.16}$$

where:

- $|\mathbf{v}|$ is the absolute value of local flow velocity
- v_x is the x component of velocity in the direction of mean flow
- $\langle\ \rangle$ is a spatial average over the pore space

The tortuosity for a random porous medium is very hard to measure experimentally. To determine the flow tortuosity the notion of diffusive tortuosity based on random-walk simulation in the pore space has been introduced. Tortuosity also can be defined just considering the shortest continuous path between any two points within the pore space. For transport phenomena it is good to use a definition like that of Eq. (2.16).

2.4 Momentum

The momentum $\mathbf{p}$ is fundamental concept in fluid mechanics. It is a product of the mass m and velocity $\mathbf{u}$ of an object.

$$\mathbf{p} = m\mathbf{u} \tag{2.17}$$

Newton's Second Law of Motion gives the force $\mathbf{F}$ as $\mathbf{F} = m\mathbf{a}$ where $\mathbf{a}$ is the acceleration. Acceleration is the time rate of change of velocity or $\mathbf{a} = \frac{d\mathbf{u}}{dt}$.

Then force can be written as:

$$\mathbf{F} = m\frac{d\mathbf{u}}{dt} = \frac{d\mathbf{p}}{dt} \tag{2.18}$$

2.5 Viscosity

Viscosity is a measure of a fluid's resistance to flow. It describes the internal friction of a moving fluid. For example: a fluid with large viscosity (*e.g.* honey) resists motion because its molecular movement is subject to high internal friction, while a fluid with low viscosity (water) flows relatively easily because its molecular movement has less friction when it is in motion. Gases usually have much lower viscosity compared to liquids.

Newton's Law of Friction presented the shear stress τ to the velocity gradient in a Newtonian fluid (Keith, 1971):

$$\tau = \mu\frac{d\mathbf{u}}{dx} \tag{2.19}$$

The proportionality factor μ in this formula is the *dynamic viscosity* of the fluid ($\mathrm{Pa}\cdot\mathrm{s}$) ($\mathrm{N}\cdot\mathrm{s/m^2}$ or $\mathrm{kg/(m}\cdot\mathrm{s)}$).

The *kinematic viscosity* ν is the dynamic viscosity μ divided by the fluid density ρ ($\mathrm{m^2/s}$).

$$\nu = \frac{\mu}{\rho} \tag{2.20}$$

The kinematic viscosity can be thought of as a diffusion coefficient for momentum since:

$$\tau = \mu \frac{d\mathbf{u}}{dx} = \mu \frac{\rho}{\rho} \frac{d\mathbf{u}}{dx} = \nu \frac{d\mathbf{p}}{dx} \tag{2.21}$$

2.6 Reynolds Number

The Reynolds Number (Re) is a non-dimensional number that reflects a measure of the ratio of inertial forces and viscous forces.

$$Re = \frac{inertial\ \ forces}{viscous\ \ forces}$$

Reynolds numbers are often used in the study of dimensional analysis of fluid dynamics problems, and also to determine dynamic similitude between different experimental cases. They are also used to describe different flows:

- laminar flow, is possible at low Reynolds numbers (<2000), where is characterized by constant and smooth fluid motion
- transition flows, when the Reynolds numbers is between 2000 and 4000
- turbulent flow, is possible at high Reynolds numbers (>4000), where fluid motion tends to produce flow instabilities as chaotic eddies and vortices

Avila *et al.* (2011) used the Reynolds number as a number of different situations where a fluid is in relative motion to a surface. This definition generally includes the fluid properties of density and viscosity, plus a velocity and characteristic length or characteristic dimension. This dimension is a matter of convention – for example a radius or diameter is equally valid for spheres or circles, but one is chosen by convention. For aircraft or ships, the length or width can be used. For flow in a pipe or sphere moving in a fluid the internal diameter is generally used today. Other shapes such as rectangular pipes or non-spherical objects have an *equivalent*

diameter defined. For fluids of variable density such as compressible gases or fluids of variable viscosity such as non-Newtonian fluids, special rules apply. The velocity may also be a matter of convention in some circumstances, notably stirred vessels.

The Reynolds number is:

$$Re = \frac{\rho \mathbf{u} L}{\mu} = \frac{\mathbf{u} L}{\nu} \tag{2.22}$$

where:

- $\mathbf{u}$ is the mean *velocity* of the object relative to the fluid (m/s)
- L is a characteristic *linear dimension*, (travelled length of the fluid; hydraulic diameter when dealing with river systems; m)
- μ is the *dynamic viscosity* of the fluid (Pa · s or N · s/m^2 or kg/(m · s))
- ν is the *kinematic viscosity* of the fluid (m^2/s)
- ρ is the density of the fluid (kg/m^3)

Low velocity, high viscosity, and confined fluid conditions lead to a low Re, the dominance of viscous forces, and laminar flow. If Re <<1, the flow is known as Stokes or creeping flow. Such flow is traditionally thought to be common for liquids in many porous media due to small pore sizes. Higher velocities, larger length scales, or less viscous fluids lead to larger Reynolds numbers and the dominance of inertial forces over viscous forces. Under high Reynolds number (2400) the flow can become unstable (i.e., the onset of turbulence).

2.7 Rayleigh Number

The Rayleigh number is a dimensionless number used in fluid mechanics that is connected with buoyancy driven flow (also identified as natural convection or free convection). The Rayleigh number is below a critical value for fluid ($Ra_c = 657.511$), then heat transfer is mainly in the form of conduction; if the Ra exceeds the critical value, heat transfer is mainly in the form of convection. The value of the critical Rayleigh number depends on the particular geometry and condition.

$$Ra = \frac{g \cdot \beta}{\nu \cdot \alpha} \Delta T \cdot L^3 \tag{2.23}$$

where:

- Ra is Rayleigh number
- g is gravity constant (N/kg)
- β is thermal expansion coefficient (1/K)
- ΔT is temperature difference between surface and quiescent temperature (K)
- L is characteristic length (for a vertical wall is height) (m)
- ν is kinematic viscosity (m^2/s)
- α is thermal diffusivity ($\alpha = \lambda/(\rho \cdot c_p)$) where λ is the heat conduction coefficient ($W/(m \cdot K)$, ρ is the density (kg/m^3), c_p is the specific heat coefficient at constant pressure ($J/kg \cdot K$)

2.8 Prandtl Number

The Prandtl number is another dimensionless number which defines the ratio of kinematic viscosity (momentum diffusivity) and thermal diffusivity (White, 2006):

$$\mathrm{Pr} = \frac{\nu}{\alpha} = \frac{\mu c_p}{k} \tag{2.24}$$

where:

- ν is kinematic viscosity (m^2/s)
- α is thermal diffusivity (m^2/s)
- μ is absolute or dynamic viscosity ($kg/m \cdot s$)
- c_p is specific heat capacity ($J/kg \cdot K$)
- k is conductivity ($W/m \cdot K$)

The Prandtl number is used for heat transfer and comprises some fluid properties that can be related to the thickness of the thermal and velocity boundary layers. It is actually the ratio of the thickness velocity boundary layer to the thickness of the thermal boundary layer. When $\mathrm{Pr} = 1$, the boundary layers coincide; $\mathrm{Pr} << 1$ means thermal diffusivity dominates; and $\mathrm{Pr} >> 1$ means momentum diffusivity dominates. White (2006) founded the typical values of the Prandtl number as in the Table 2-1:

Table 2-1 Prandtl number (Withe, 2006)

MATERIAL	Pr -ranges
Gases	0.7-1.0
Water	1-10
Liquid metals	0.001-0.03
Oil	50-2000
Air 20^0C	0.71

2.9 Peclet Number

The Peclet number (Pe) is a dimensionless number used in calculations involving convective heat transfer. It is defined as the ratio of the thermal energy convected to the fluid to that conducted within the fluid, and it depends on the heat capacity, density, velocity, characteristic length and heat transfer coefficient. The Peclet number represents the product of the Reynolds number (Re) and the Prandtl number (Pr).

$$Pe = \frac{advective\ transport\ rate}{diffusive\ transport\ rate} = \mathrm{Re} \cdot \mathrm{Pr}$$

2.10 Nusselt Number

The Nusselt (Nu) number is the ratio of convective heat transfer across the boundary (Munson *et al*., 2008) compared to conduction:

$$Nu = \frac{convective\ heat\ transfer}{conductive\ heat\ transfer} = \frac{hL}{k_f}$$

where:

- L is a characteristic length
- k_f is the thermal conductivity of the fluid
- h is the convective heat transfer coefficient

A Nusselt number close to 1 is associated with laminar flows, and between 100-1000 is generally associated with turbulent flows. Munson *et al.* (2008) shows, for example, as seen in Table 2-2, the correlations for calculating the heat transfer coefficient in a pipe flow.

Table 2-2 Nusselt number for laminar and turbulent flow (Munson et al., 2008)

	Correlation	Notes
Laminar Flow	$Nu = 3.66$	For constant wall temperature
	$Nu = 4.36$	For constant heat flux at wall
Turbulent Flow	$Nu = 0.023\mathrm{Re}^{0.8}\,\mathrm{Pr}^{n}$	n=0.4 for heating n=0.33 for cooling Valid for $0.6 \le \mathrm{Pr} \le 160$
	$Nu = \frac{\frac{f}{8}(\mathrm{Re}-1000)\mathrm{Pr}}{1+12.7\sqrt{f/8}\left(\mathrm{Pr}^{2/3}-1\right)}$	Where $f = \frac{1}{\left(1.8\log_{10}\frac{\mathrm{Re}}{6.9}\right)^2}$ is the Darcy friction factor

2.11 Mach Number

The dimensionless number Mach is the ratio of speed of an object moving through a fluid and the local speed of sound (Graebel, 2001)

$$M = \frac{c}{c_s} \tag{2.25}$$

where:

- M is the Mach Number
- c is the velocity of the source relative to the medium
- c_s is the speed of sound in the medium

2.12 Poiseuille Flow

An important and simple type of flow is that which occurs in a pipe or a slit between two parallel surfaces. It is called the Poiseuille flow after the Frenchman Jean Leonard Marie Poiseuille (1797-1869) (Sutera and Skalak, 1993).

The factors that influence for flow through a pipe or a slit between two parallel surfaces are:

- the radius of the pipe or the width of the slit between the two parallel surfaces; fluid travels more easily through a wide pipe or slit than a narrow one
- the viscosity (*i. e.* air flows much more easily than oil)
- the pressure difference between the ends (at higher pressure difference the flow will be faster), or body force

The velocities at the walls are 0 (no-slip boundaries) in a slit or pipe and the maximum velocities are in the middle. In Figure 2-*4*, Sukop and Thorne (2005) showed the velocity profile of a slit of width that can be seen to be parabolic:

$$u(x) = \frac{G^*}{2\mu}\left(a^2 - x^2\right) \tag{2.26}$$

In this formula G^* can be:

- pressure gradient $\left(P_{in} - P_{aut}\right)/L$
- gravitational pressure gradient $G^* = \rho g$

The average velocity in a slit is 2/3 of the maximum; the maximum velocity is attained at $x = 0$.

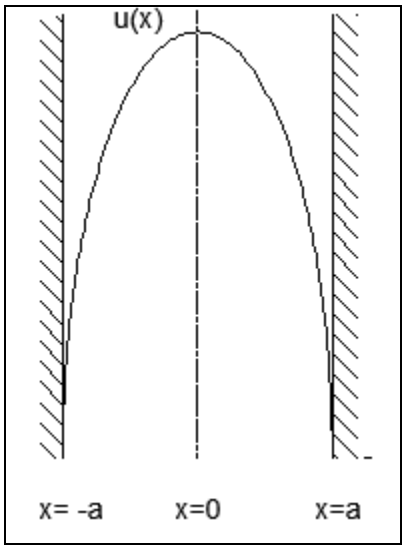


Figure 2-3 Poiseuille velocity profile (Sukop and Thorne, 2005)

3 Lattice Boltzmann Method

In this Chapter an introduction to the Kinetic theory of gases is presented. Tthe Boltzmann equation describes is explained as it the evolution of the single particle distribution function in the phase space by a partial differential equation. The evolution of the method explains how the distribution functions is updated and hence what the lattice Boltzmann method is.

3.1 Kinetic Theory

The kinetic theory was developed in relationship with the atomic structure of matter, assuming that the constituents, atoms and molecules are in a continuous motion. Krönig, and later Maxwell and Clausius postulated that heat energy is identical with the kinetic energy of atoms or molecules (Wolfgang 1973). Later Boltzmann introduced the theory of "probability of a state" and related it to the system entropy (Badino, 2006).

In the theory, the particles in a substance are the same type in a solid, liquid or gaseous state. They differ only in:

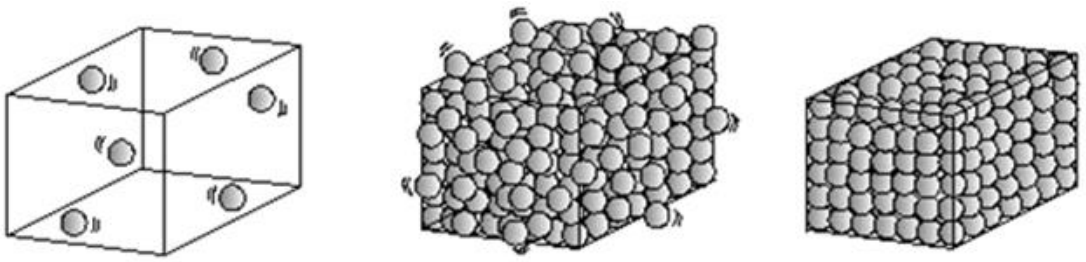
- their arrangement,
- how strongly the particles are held together, and
- the amount of kinetic energy that they possess.

Kinetic theory can be used to explain how solids, liquids and gases differ in movement and arrangement. The various states of matter are differentiated as follows:

- Gases: except for collisions, the molecular motion is force free (neglecting external force fields, such as pressure gradients)

- Liquids: because there are always a large number of molecules which interact nonuniform motion or an equilibrium configuration is a good approximation
- Solids: the atoms oscillate about an equilibrium configuration

Figure 3-1 shows the arrangement of molecules in the matter states:



(a) (b) (c)

Figure 3-1 Matter states: (a Gaseous); (b) Liquid and (c) Solid

3.1.1 Kinetic Theory of Gases

The kinetic theory of gases (also known as kinetic-molecular theory) is the study of the microscopic behaviour of molecules and their interactions which lead to relationships, like the ideal gas law and correlations between macroscopic properties and microscopic phenomena. This was the first theory to describe gas pressure in terms of collisions with the walls of a container, rather than from static forces that push the molecules apart. The kinetic theory explains why any given molecule has a definite mass (m), definite velocity ($\mathbf{e}$), and kinetic energy of $\frac{1}{2}m\mathbf{e}^2$. Kinetic theory also explains how the different sizes of particles in a gas can give them different

individual speeds. Essentially the particles follow classical mechanics laws; consequently, the Newton's second law of motion for each particle is expressed as:

$$\frac{\partial^2 \mathbf{x_a}}{\partial t^2} = \mathbf{F}_a \tag{3.1}$$

$$\frac{\partial \mathbf{x}_a}{\partial t} = \mathbf{e}_a \tag{3.2}$$

where:

- $\mathbf{x}_a$ is position vector
- $\mathbf{e}_a$ is velocity vector, $a = 1,2,...n$
- $\mathbf{F}_a$ is the sum of all forces acting on the system and, $a = 1,2,...n$, n is the number of particles

3.1.2 Bridge between Microscopic and Macroscopic Levels

The primary variable of interest is a one-particle probability distribution function (PPDF), $f(\mathbf{x},\mathbf{e},t)$. They defined that $f(\mathbf{x},\mathbf{e},t)\cdot d^3\mathbf{x}\cdot d^3\mathbf{e}$ is the number of particles which, at time t, are located within a phase-space control element ($d^3\mathbf{x}\cdot d^3\mathbf{e}$) about **x** and **e** (**x** is a particle's coordinate in physical space and **e** is a particle's velocity).

Liboff (1998) states that the density can be obtained by integrating f over only the velocity vector:

$$\int f d\mathbf{e} = \rho(\mathbf{x},t) \tag{3.3}$$

The first moment is defined as:

$$\int \mathbf{e} f d\mathbf{e} = \rho \mathbf{u} \tag{3.4}$$

For calculating the second moment, a new variable is introduced (Liboff, 1998):

$$\mathbf{c} = \mathbf{e} - \mathbf{u} \tag{3.5}$$

The velocity of the particles has two component: $\mathbf{u}$, which is the average velocity (bulk velocity) and $\mathbf{c}$, the random (peculiar) velocity.

$$\frac{1}{2}\int \mathbf{c}^2 f d\mathbf{e} = \frac{1}{2}\int \mathbf{e}^2 f d\mathbf{e} - \frac{1}{2}\rho \mathbf{u} \tag{3.6}$$

All the moments used are seen in Table 3-1:

Table 3-1 Moments used in kinetic theory

No.	Moment	Equation
1	Zeroth moment	$\int f d\mathbf{e} = \rho$
2	First moment	$\int \mathbf{e} f d\mathbf{e} = \rho \mathbf{u}$
3	Second moment	$\frac{1}{2}\int \mathbf{c}^2 f d\mathbf{e} = \frac{1}{2}\int \mathbf{e}^2 f d\mathbf{e} - \frac{1}{2}\rho \mathbf{u}$

In Eq. (3.6) on the left hand side, the term is the internal energy per unit volume. On the right hand side, the first term is the total energy per unit volume and the second term is the kinetic energy per unit volume. After considering internal energy per unit mass as e, the term on the left side became:

$$\frac{1}{2}\int \mathbf{c}^2 f d\mathbf{e} = \rho e \tag{3.7}$$

The pressure is directly related to the energy per unit mass and is known as:

$$p = \frac{2}{3}\rho e \tag{3.8}$$

The macroscopic relation between pressure and temperature is defined as:

$$p = \rho RT \tag{3.9}$$

where:

- p is pressure
- R is gas constant
- T is temperature

Combining the equations Eq. (3.8) and Eq. (3.9) then the temperature as a function of microscopic velocity can be expressed as:

$$T = \frac{2}{3}\frac{e}{R} \tag{3.10}$$

In conclusion, all the important macroscopic properties such as densities, velocities, mass, pressure, and temperature can be calculated.

3.2 History of Lattice Boltzmann Method

The lattice Boltzmann method (LBM) is the most promising method in computational fluid dynamics (Chen and Doolen, 1998). The method has its roots from the older method of lattice gas automata (LGA). The advantages for this method are ease of implementation, a potential for parallelisation, and a nature that makes expansions of the method relatively simple (Succi, 2001).

The LGA was developed from cellular automata, with a history started in 1973. Cellular automata are a discrete computer model of a grid system with an evolution defined by mathematical rules. This method was a model for the motion of single particles in fluid using behavioural rules which were as simple as possible. Hence, LGA can be seen as a simple method of molecular dynamics.

Molecular dynamics (MD) is a microscopic conceptual approach where the fluid dynamics is modeled based on the collision and other interactions between the individual molecules (Mohamad, 2011). In these models the macroscopic properties are recovered using statistical mechanics (Mohamad, 2011). The main problem in MD is excessive usage of the computing resources and their limitations in extending to bigger domains (Mohamad, 2011).

3.2.1 Lattice Gas Automata

Succi (2001) describes the lattice is a systematic arrangement of basic shapes. There are several types of lattice in two dimensions where the basic shape can be rectangle, triangle, regular hexagon, and so on. A square lattice is shown in Figure 3-2. This model was first proposed by Hardy, Pomeau and de Pazzis (named HPP) in 1973.

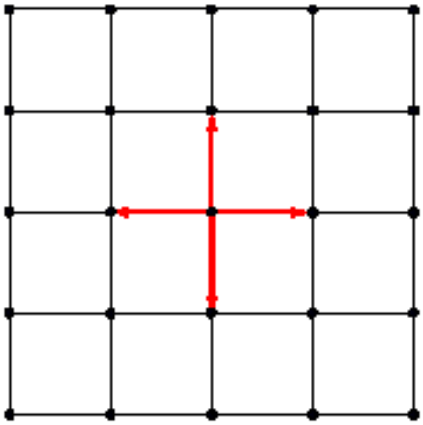


Figure 3-2 Square lattice - HPP model (Succi, 2001)

Lattice Gas Automata works with a group of particles on the lattice nodes while colliding with particles located at the neighbouring nodes while conserving the mass and the momentum. Each particle has a velocity whose direction is along the connection with one of the neighbouring nodes. All the particles possess momentum and the collision between them is governed by a set

of rules which change the velocities of the particles but conserves the total momentum of all the particles existing at a node. The particles then propagate to their close nodes according to the direction of their new velocities. In each time step iteration, at each lattice node, there is collision between particles followed by propagation.

Frisch, Hasslacher and Pomeau (1986) replace the square lattice used in HPP model with a triangular lattice because the model is insufficiently symmetric. This model is named FHP and it presents hexagonal symmetry as in Figure 3-3:

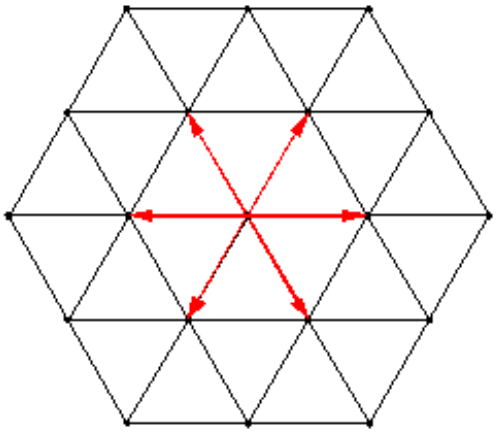


Figure 3-3 Triangular lattice - FHP model (Frisch et al., 1986)

LGA models are basically defined by the kinetic equation:

$$n_a\left(\mathbf{x}+\mathbf{e}_a\Delta t,t\right)=n_a\left(\mathbf{x},t\right)+\Delta_a \tag{3.11}$$

where:

- n_a is a Boolean in a direction
- $\mathbf{e}_a$ is a velocity
- Δ_a is the collision function, which is dependent on the LGA model

The density is calculated by summing up the total number of particles at each node:

$$\rho(\mathbf{x},t)=\sum_a n_a(\mathbf{x},t) \tag{3.12}$$

The momentum density is:

$$\rho(\mathbf{x},t)\mathbf{u}(\mathbf{x},t)=\sum_a \mathbf{e}_a n_a(\mathbf{x},t) \tag{3.13}$$

There $\mathbf{u}(\mathbf{x},t)$ is the macroscopic velocity, which means velocity of the entire particles.

McNamara and Zanetti introduced the first generation lattice Boltzmann method in 1988, where they have replaced the Boolean fields in Eq. (3.11) by the single particle distribution functions f_a .

$$f_a(\mathbf{x}+\mathbf{e}_a\Delta t,t+\Delta t)= f_a(\mathbf{x},t)+\Delta_a \tag{3.14}$$

These models are named mesoscopic models because they do not produce results on a sigle particle, but a distribution of particles.

3.2.2 Distribution Function

The distribution function (DF) is a function of seven variable $f(x,y,z,t;\mathbf{e}_x,\mathbf{e}_y,\mathbf{e}_z)$ for three dimension, where x,y,z represent the position on the Cartesian system, t represents the time and $\mathbf{e}_x,\mathbf{e}_y,\mathbf{e}_z$ represent the velocities on axis. The velocity distribution function $f(\mathbf{e}_x)$ is defined as a fraction of particles in a certain location of a container of gas or liquid with velocities between $\mathbf{e}_x$ and $\mathbf{e}_x+d\mathbf{e}_x$ in x direction. The entire fraction of particles with velocities between $\mathbf{e}_x$ and $\mathbf{e}_x+d\mathbf{e}_x$, $\mathbf{e}_y$ and $\mathbf{e}_y+d\mathbf{e}_y$, $\mathbf{e}_z$ and $\mathbf{e}_z+d\mathbf{e}_z$ is known as $f(\mathbf{e}_x)f(\mathbf{e}_y)f(\mathbf{e}_z)$. Maxwell

(Fowler, 2008) presents the applicable particle distribution function using the symmetry argument.

$$f(\mathbf{e}_x) = A\exp^{-B\mathbf{e}_x^2} \tag{3.15}$$

The magnitude of velocity in three dimensional velocity spaces is:

$$\mathbf{e}^2 = \mathbf{e}_x^2 + \mathbf{e}_y^2 + \mathbf{e}_z^2 \tag{3.16}$$

This equation represents a sphere centered at origin with the surface area $4\pi\mathbf{e}^2$.

Maxwell presents the following equations (Eqs.3.17-3.24) for the distribution function. The distribution function with the same speed $\mathbf{e}$ is:

$$f(\mathbf{e}) = 4\pi\mathbf{e}^2 f(\mathbf{e}_x) f(\mathbf{e}_y) f(\mathbf{e}_z) = 4\pi\mathbf{e}^2 A^3 \exp^{-B\mathbf{e}^2} \tag{3.17}$$

In condition where all these fractions corresponding to $\mathbf{e} = 0,1,...\infty$ add up to one:

$$\int_0^\infty f(\mathbf{e}) d\mathbf{e} = 1 \tag{3.18}$$

The constant A and B obtained after solving Eq. (3.17) are related:

$$A = \sqrt{\frac{B}{\pi}} \tag{3.19}$$

The average kinetic energy per particle is known as:

$$\overline{\frac{1}{2}m\mathbf{e}^2} = \frac{\int_0^\infty \frac{1}{2}m\mathbf{e}^2 f(\mathbf{e}) d\mathbf{e}}{\int_0^\infty f(\mathbf{e}) d\mathbf{e}} \tag{3.20}$$

and:

$$\overline{\frac{1}{2}m\mathbf{e}^2} = \frac{3m}{4B} \tag{3.21}$$

Expressing the average kinetic energy in terms of temperature T and Boltzmann constant k yields:

$$\overline{\frac{1}{2}m\mathbf{e}^2} = \frac{3}{2}kT \tag{3.22}$$

and through comparison of Eq. 3.21 and Eq. 3.22 gives the value of B:

$$B = \frac{m}{2kT} \tag{3.23}$$

Finally the distribution function of velocities obtained is:

$$f(\mathbf{e}) = \left[\sqrt{\frac{m}{2\pi kT}}\right]^3 \exp\left[-\frac{m\mathbf{e}^2}{2kT}\right] \tag{3.24}$$

3.3 The Lattice Boltzmann Method

The LBM is a numerical method for simulation of fluid flow and was first proposed by McNamara and Zanetti (1988). The LBM approximates the continuous Boltzmann equation by discretizing a physical space with lattice nodes and velocity space on a set of microscopic velocity vectors (Maier *et al*., 1997). A lattice is an infinite array of discrete points generated by a set of discrete translation rules and it corresponds to a set of nodes that represents the discrete phase of physical space, where each node corresponds to either a void or a solid. Mohammed (2005) represents that image of a porous medium in Figure 3-5(a) as solids and voids (pores), for solids are black pixels and for voids are white pixels. The lattice nodes are created at the center of each white pixel (Fig. 3-5(b)).

In the LBM fluid, particles travel between these nodes following the main LB equations, which satisfy the Navier-Stokes equations. These equations can be solved using several numerical methods (finite element and finite difference). The studies have been conducted by numerous researchers (Succi, 2001, Yoshino *et al*., 2004) who concluded that LBM is comparable in accuracy to traditional methods in relatively high Reynolds numbers and superior accuracy in low Reynolds numbers, and in the case of flow through porous media (Vafai, 1984; Koponen *et al*., 1997; Zhaoli and Guo, 2005).

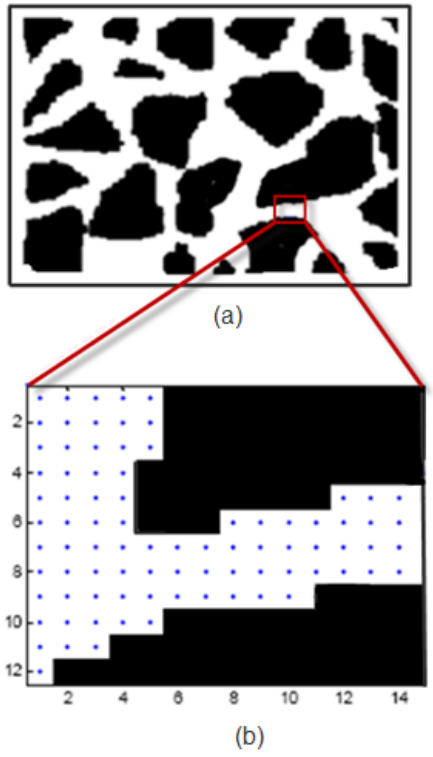
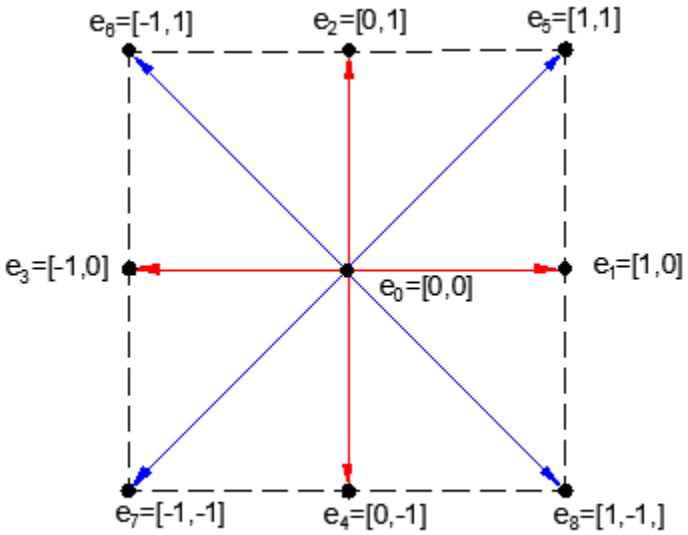


Figure 3-4 (a) Image of structure with black area represents solids and white area represents the voids, (b) Lattice nodes at the center of each white pixel (Mohammed, 2005)

The basic idea of LBM comes from the kinetic theory of fluids. LBM considers a representative volume element of fluid (a node in the lattice) composed of particles that are defined in terms of

their particle velocity distribution function $f_a(x,t)$. This function is a discrete function that has Q number of components, where Q is the number of microscopic velocity directions $(\mathbf{e}_a)$ around a lattice node (Fig. 3-6). Each component of the function $(f_a, a = 1,2....$ Q) represents a fraction of the total number of particles at each node with a microscopic velocity $\mathbf{e}_a$ (Fig. 3-6). The fluid particles travel on the lattice nodes through the use of the distribution function which determines the macroscopic velocity of fluid at each lattice node (Kutay, 2005).



(a)

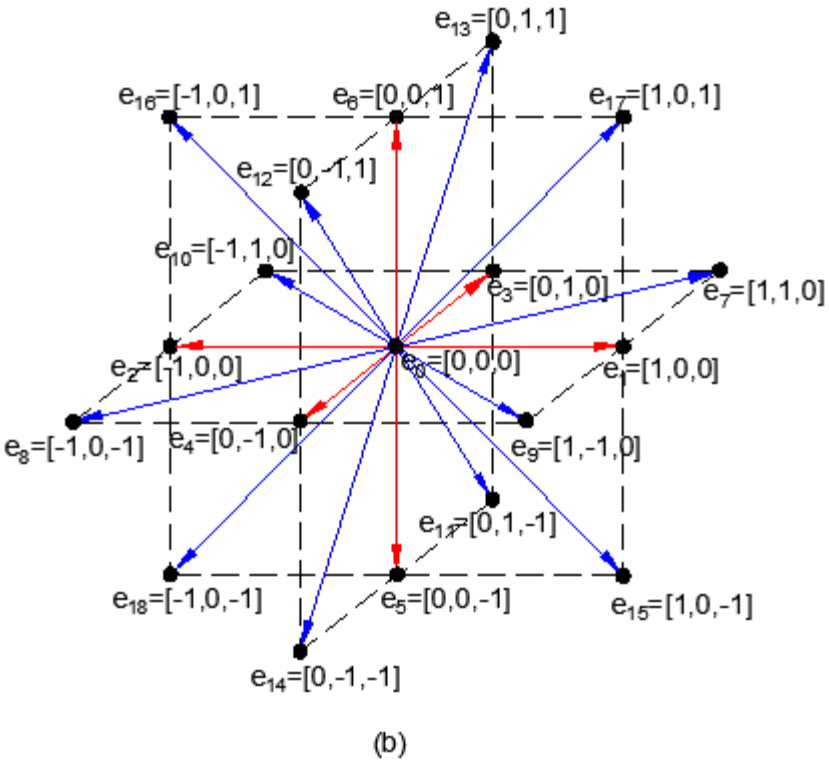


Figure 3-5 (a) D2Q9 and (b) D3Q19 lattice microscopic velocity directions (Kutay, 2005)

3.3.1 Boltzmann Equation

The Boltzmann equation (BE) is a partial differential equation which governs the transport phenomenon of the density distribution function.

The moments of the distribution function over velocity space are taken to obtain the macroscopic quantities such as density, momentum density, and internal energy as introduced by Chapman *et al.* (1970).

$$\rho(\mathbf{x},t)=\int f(\mathbf{x},\mathbf{e},t)d\mathbf{e} \tag{3.25}$$

$$\rho\mathbf{u}(\mathbf{x},t)=\int \mathbf{e}\cdot f(\mathbf{x},\mathbf{e},t)d\mathbf{e} \tag{3.26}$$

$$\rho e(\mathbf{x},t)=\frac{1}{2}\int(\mathbf{e}-\mathbf{u})^2\cdot f(\mathbf{x},\mathbf{e},t)d\mathbf{e} \tag{3.27}$$

Where $\mathbf{u}$ is the microscopic velocity vector and e is the specific internal energy.

If there are no collisions, the number of particles in this set does not change and can be expressed:

$$f\left(\mathbf{x}+\mathbf{e}\Delta t,\mathbf{e},t+\Delta t\right)=f\left(\mathbf{x},\mathbf{e},t\right) \tag{3.28}$$

If collisions between particles are counted, the number of particles in the final set will change and is given by introducing a collision function $\Omega\left(f\right)$ which defines the rate of change of the distribution function f at a fixed point $\left(\mathbf{x},t\right)$.This is the Boltzmann equation (BE)

$$f\left(\mathbf{x}+\mathbf{e}\Delta t,\mathbf{e},t+\Delta t\right)-f\left(\mathbf{x},\mathbf{e},t\right)=\Omega\left(f\right)\Delta t \tag{3.29}$$

As the BE contained a collision term and presents problems in solving it, Bhatnagar, Gross and Krook in 1954 found the solution. They used a mathematically simple relaxation term, because the collisions tend to relax the distribution function toward equilibrium:

$$\Omega\left(f\right)=-\frac{f-f^{eq}}{\lambda} \tag{3.30}$$

where:

- $\Omega\left(f\right)$ is BKG collision operator
- λ is the relaxation time which is a function of viscosity of the fluid
- f is distribution function
- f^{eq} is the equilibrium distribution function

Bhatnagar *et al.* (1954) introduced the concept that the equilibrium distribution function has a Maxwell- Boltzmann distribution of velocities and is proportional to the density. This means that the Maxwell-Boltzmann distribution can be taken to represent f^{eq} .

3.3.2 Lattice Boltzmann Equation

The lattice Boltzmann equation (LBE) is derived directly from the LGA. Finite set of velocities $\mathbf{e}_a$ along the links of the lattice is presented by the distribution function $f(\mathbf{x},\mathbf{e},t)$ which is transformed to corresponding discrete distribution functions $f_a(\mathbf{x},t)$. He and Luo (1997) noted that in order to preserve the conservation laws, when discretized, these moments must be preserved exactly, as in the following Eqs. (3.31-3.38). The moments of the distribution function are given by:

$$I = \int \Psi(\mathbf{e}) f(\mathbf{x},\mathbf{e},t) d\mathbf{e} \tag{3.31}$$

Where $\Psi(\mathbf{e})$ represent a polynomial in $\mathbf{e}$.

The moments of the distribution function I can be calculated by:

$$I = \sum_{a=1}^{n} \Psi(\mathbf{e}_a) w_a f(\mathbf{x},\mathbf{e}_a,t) \tag{3.32}$$

where:

- w_a are the weights coefficient
- n is the number of discrete velocities
- $\mathbf{e}_a$ is the discrete velocity set

Accordingly the hydrodynamic moments of Eqs. (3.25), (3.26) and (3.27) are transformed into summation:

- for $\Psi(\mathbf{e})=1$, the mass density is known as:

$$\rho(\mathbf{x},t) = \sum_{a=0}^{n} f_a = \sum_{a=0}^{n} f_a^{eq} \tag{3.33}$$

- for $\Psi(\mathbf{e})=\mathbf{e}$, the momentum density is known as:

$$\rho \mathbf{u}(\mathbf{x},t) = \sum_{a=0}^{n} \mathbf{e}_a f_a = \sum_{a=0}^{n} \mathbf{e}_a f_a^{eq} \tag{3.34}$$

- for $\Psi(\mathbf{e}) = (\mathbf{e} - \mathbf{u})^2$, the internal energy is known as:

$$\rho e(\mathbf{x},t) = \frac{1}{2}\sum_{a=0}^{n} (\mathbf{e}_a - \mathbf{u})^2 f_a = \frac{1}{2}\sum_{a=0}^{n} (\mathbf{e}_a - \mathbf{u})^2 f_a^{eq} \tag{3.35}$$

where:

$$f_a(\mathbf{x},t) = f_a = w_a f(\mathbf{x}, \mathbf{e}_a, t) \tag{3.36}$$

$$f_a^{eq}(\mathbf{x},t) = f_a^{eq} = w_a f_a^{eq} = w_a f^{eq}(\mathbf{x}, \mathbf{e}_a, t) \tag{3.37}$$

If w_a are constants, is the factor weights specific for each model chosen and the physical point $\mathbf{x}, t$ in the discretized physical/temporal space corresponds to a point on the lattice.

The discrete form of the Boltzmann equation using the BKG collision operator is known as:

$$\frac{\partial f_a}{\partial t} + \mathbf{e}_a \cdot \nabla f_a = -\frac{f_a - f_a^{eq}}{\lambda} \tag{3.38}$$

First order discretization of the above equation with lattice spacing $\delta\mathbf{x}$ and time step δt (Inamuro *et al*., 1997) leads to:

$$\frac{f_a(\mathbf{x}, t+\delta t) - f_a(\mathbf{x},t)}{\delta t} + \mathbf{e}_{ax} \frac{f_a(\mathbf{x}+\mathbf{e}_a\delta t, t+\delta t) - f_a(\mathbf{x}, t+\delta t)}{\delta \mathbf{x}} = -\frac{1}{\lambda}\left(f_a(\mathbf{x},t) - f_a^{eq}(\mathbf{x},t)\right) \tag{3.39}$$

$$f_a(\mathbf{x}+\mathbf{e}_a\delta t, t+\delta t) - f_a(\mathbf{x},t) = -\omega\left(f_a(\mathbf{x},t) - f_a^{eq}(\mathbf{x},t)\right) \tag{3.40}$$

He and Luo (1997) use the velocity $\mathbf{e}_{ax} = \delta\mathbf{x}/\delta t$ and the frequency $\omega = 1/\tau$ which is correlated to the dimensionless relaxation time $\tau = \lambda/\delta t$.

3.3.3 Equilibrium Distribution Function

The equilibrium distribution function for LBM can be derived from the Maxwell-Boltzmann distribution discussed in Section 3.2.2 :

$$f^{eq} = \rho\left(\frac{m}{2\pi kT}\right)^{\frac{D}{2}} \exp\left[-\frac{m(\mathbf{e}-\mathbf{u})^2}{2kT}\right] \tag{3.41}$$

where $D=2$ for a two dimensional model which is used in this case. The speed of sound c_s is a relation between Boltzmann constant k, temperature T and mass m:

$$c_s^2 = \frac{kT}{m} \tag{3.42}$$

He and Luo (1997) introduce the low-Mach number limit in Eq. (3.41):

$$f^{eq} = \exp\left[-\frac{\left(\mathbf{e}^2-\mathbf{u}^2\right)}{2c_s^2}\right] = \exp\left[-\frac{\mathbf{e}^2}{2c_s^2}\right]\exp\left[-\frac{\mathbf{u}^2-2\mathbf{u}\cdot\mathbf{e}}{2c_s^2}\right] \tag{3.43}$$

The Taylor series expansion for $\exp(x)$ is:

$$\exp(x) = 1+\frac{x}{1!}+\frac{x^2}{2!}+\frac{x^3}{3!}+\ldots \tag{3.44}$$

Using the Eq. (3.44) is obtained:

$$\exp\left[-\frac{\mathbf{u}^2-2\mathbf{u}\cdot\mathbf{e}}{2c_s^2}\right] \approx 1-\frac{\mathbf{u}^2-2\mathbf{u}\cdot\mathbf{e}}{2c_s^2}+\frac{\left(\mathbf{u}^2-2\mathbf{u}\cdot\mathbf{e}\right)^2}{8c_s^4}+\ldots \tag{3.45}$$

$$\exp\left[-\frac{\mathbf{u}^2-2\mathbf{u}\cdot\mathbf{e}}{2c_s^2}\right] \approx 1-\frac{\mathbf{u}^2}{2c_s^2}+\frac{\mathbf{u}\cdot\mathbf{e}}{c_s^2}+\frac{\left(\mathbf{u}\cdot\mathbf{e}\right)^2}{2c_s^4}+\mathrm{O}\left(\mathbf{u}^3\right) \tag{3.46}$$

He and Luo (1997) use the equilibrium distribution function (based on the Equations (3.45, 3.46)) for low Mach number as:

$$f^{eq} = \frac{\rho}{2\pi c_s^2} \exp\left[-\frac{\mathbf{e}^2}{2c_s^2} \right] \left[1 - \frac{\mathbf{u}^2}{2c_s^2} + \frac{\mathbf{u}\cdot\mathbf{e}}{c_s^2} + \frac{(\mathbf{u}\cdot\mathbf{e})^2}{2c_s^4} \right] + \mathrm{O}(\mathbf{u}^3) \tag{3.47}$$

For the velocity $\mathbf{e}_a$ the equilibrium distribution function is:

$$f^{eq}(\mathbf{x}, \mathbf{e}_a, t) = \frac{\rho}{2\pi c_s^2} \exp\left[-\frac{\mathbf{e}_a^2}{2c_s^2} \right] \left[1 - \frac{\mathbf{u}^2}{2c_s^2} + \frac{\mathbf{u}\cdot\mathbf{e}_a}{c_s^2} + \frac{(\mathbf{u}\cdot\mathbf{e}_a)^2}{2c_s^2} \right] \tag{3.48}$$

3.3.4 Lattice Boltzmann Models

Lattice Boltzmann models (LBMs) have an important capacity to simulate single and multiphase fluids. The model is observed from a particle perspective where collisions, streaming, and interaction between particle-particle and particle-surface create the entire theoretical framework.

Numerous LBMs exist for numerical solution of fluid flow scenarios, where each model has microscopic movement of the fluid particles. The LBMs are typically represented as $\mathrm{D}n\mathrm{Q}b$ where n is the number of dimension and b is the number of microscopic velocity direction ($\mathbf{e}_a$), as in Fig. 3-5. For example, D2Q9 represents a two-dimensional geometry with nine velocity directions, and D3Q19 represents three-dimensional geometry with nineteen velocity directions.

3.3.5 Lattice Isotropy

An explanation has been found (Wolf-Gladrow, 2005) for the condition necessary for lattice isotropy. Not every lattice is appropriate for use in the LBM. For LGA, there are numerous

conditions stipulating that a set of lattice vectors must have sufficiently isotropic behaviour to develop the Navier-Stokes equation. Latt (2007) uses the BGK model for a fluid requires that there exist a constant c_s, set of weight w_a for the lattice velocities $\mathbf{e}_a$ such that the following equations:

$$\sum_a w_a = 1 \tag{3.49a}$$

$$\sum_a w_a e_{a\alpha} = 0 \tag{3.49b}$$

$$\sum_a w_a e_{a\alpha} e_{a\beta} = c_s^2 \delta_{\alpha\beta} \tag{3.49c}$$

$$\sum_a w_a e_{a\alpha} e_{a\beta} e_{a\gamma} = 0 \tag{3.49d}$$

$$\sum_a w_a e_{a\alpha} e_{a\beta} e_{a\gamma} e_{a\delta} = c_s^4 \left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} \right) \tag{3.49e}$$

$$\sum_a w_a e_{a\alpha} e_{a\beta} e_{a\gamma} e_{a\delta} e_{a\varepsilon} = 0 \tag{3.49f}$$

Where w_a is a set of lattice vector weights and must be selected for each lattice to satisfy these conditions.

Several authors (Hardy *et al*., 1973, Frisch *et al*., 1986, Maier *et al*., 1997, Begun and Basil, 2008) illustrated the lattices which meet these conditions as D1Q3, D2Q7, D2Q9, D3Q15, D3Q19 and D3Q27 (Fig. 3-6 a, b, c, d, e, f).

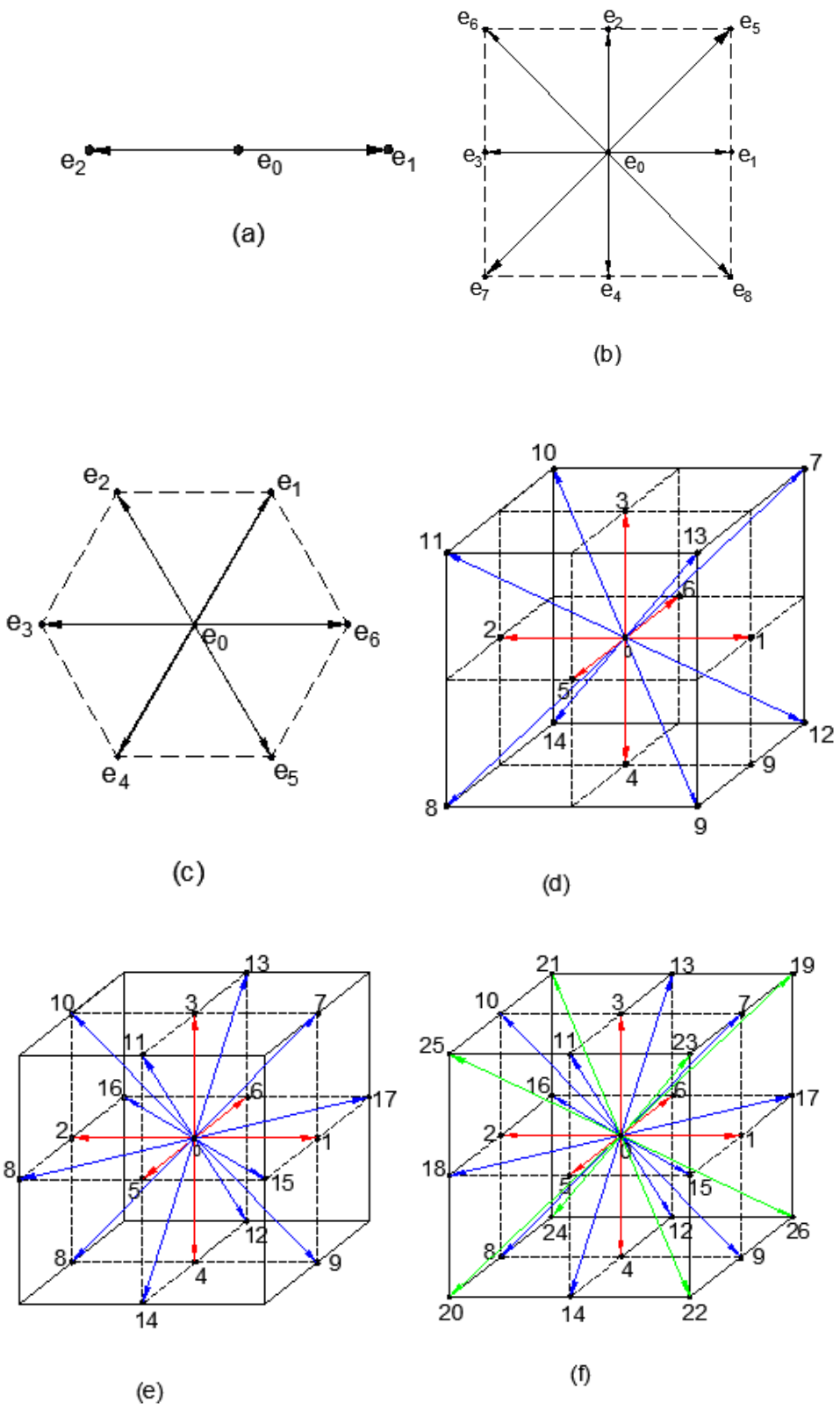


Figure 3-6 (a) D1Q3, (b) D2Q7, (c) D2Q9, (d) D3Q15, (e) D3Q19, and (f) D3Q27

For example, in two dimensions it is most common to use D2Q9 because it has a square grid of nodes and is easier to implement on a computer and in three dimensions an advantageous model is D3Q19. In the D2Q9 lattice, three different w_a weights are used:

- for w_0 , $a=0$, is for the rest vector,
- for w_s, $a=1,2,3,4,$ are for the short vectors,
- for w_l, $a=5,6,7,8,$ are for the long vectors.

Then from the Eqs.: (3.49 a, 3.49c and 3.49e) the following conditions arise:

$$w_0+4w_s+4w_l=1 \tag{3.50a}$$

$$2w_s+4w_l=c_s^2 \tag{3.50b}$$

$$2w_s+4w_l=3c_s^4 \tag{3.50c}$$

$$4w_l=c_s^4 \tag{3.50d}$$

The solution for this system and for the other lattices of Fig. 3-6 are shown in Table 3-2:

Table 3-2 Lattice weight for different models

D2Q7		**D2Q9**		**D3Q15**		**D3Q19**	
e_a	w_a	e_a	w_a	e_a	w_a	e_a	w_a
(0)	$w_0=1/2$	$(0,0)$	$w_0=4/9$	$(0,0,0)$	$w_0=2/9$	$(0,0,0)$	$w_0=1/3$
(±1)	$w_s=1/12$	$(\pm1,0)$	$w_s=1/9$	$(\pm1,0,0)$	$w_s=1/9$	$(\pm1,0,0)$	$w_s=1/18$
		$(\pm1,\pm1)$	$w_l=1/36$	$(\pm1,\pm1,\pm1)$	$w_l=1/72$	$(\pm1,\pm1,0)$	$w_l=1/36$
$c_s^2=1/2$		$c_s^2=1/3$		$c_s^2=1/3$		$c_s^2=1/3$	

Based on observations D1Q3 is the projection of the D2Q9 lattice on to one dimension, and D2Q9 lattice also is the projection of the D3Q15 lattice on to two dimensions.

3.3.6 A Two-Dimensional Model

Frisch, Hasslacher, and Pomeau (1986) introduced the first lattice gas model that can accurately simulate the two dimensional Navier-Stokes equations. The name is the FHP model; it is constructed on an equilateral triangular lattice (Fig. 3-6c).

The distance between every two points is 1 *lattice unit* (*lu*) and all particles have only one speed 1*lu/time step* (*luts*$^{-1}$). At every lattice point $\mathbf{x}$ may be up to six particles, one for every possible velocities defined by the particle speed and one of the six possible directions:

$$\mathbf{e}_a = (\cos \pi a/3, \sin \pi a/3) \tag{3.51}$$

Where $a = 1,2...,6$ and the velocities vectors start from the origin (0, 0) to the Cartesian coordinate. As noted (Rothman and Zaleski, 1997) a string of Boolean variables $\mathbf{n} = (n_1, n_2, ..., n_6)$ contains the states ($n_a = 0$ or 1) indicating the presence (1) or absence (0) of particles moving from a lattice site at $\mathbf{x}$ to a neighbouring site $\mathbf{x} + \mathbf{e}_a$.

In recent years, for simulation of fluid flow the D2Q9 model has been more commonly used.

The D2Q9 model is a two dimensional model with square lattice and nine velocity vectors. Shokouhmand *et al*. (2009) used this model, and the lattice velocities are presented in Fig. 3-5a:

The velocity vectors are $\mathbf{e}_a$, with $a = 0,1,2,3,4,5,6,7,8$.

The equilibrium distribution function for this model is given by (Eq. 3.48):

$$f_a^{(eq)} = w_a\left[1+\frac{\mathbf{e}_a\cdot\mathbf{u}}{c_s^2}+\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{2c_s^4}-\frac{\mathbf{u}^2}{2c_s^2}\right] \tag{3.52}$$

And the weight factors for D2Q9 are:

$$w_a=\begin{cases}\frac{4}{9}, & a=0\\ \frac{1}{9}, & a=1,2,3,4\\ \frac{1}{36}, & a=5,6,7,8\end{cases} \tag{3.53}$$

For D2Q9 model the lattice velocities are defined as:

$$\mathbf{e}_a=\begin{cases}0, & a=0\\ (1,0),\ (0,1),\ (-1,0),\ (0,-1) & a=1,2,3,4\\ (1,1),\ (-1,1),\ (-1,-1),\ (1,-1) & a=5,6,7,8\end{cases} \tag{3.54}$$

The important parameters in this model are:

- Density $\rho=\sum_a f_a$,
- Macroscopic velocity $\rho\mathbf{u}=\sum_a f_a\mathbf{e}_a$,
- Pressure $p=\rho c_s^2$,
- Kinematic viscosity $\nu=c_s^2\left(\tau-\frac{1}{2}\right)\delta_t$.

With this model the value of particle speed c is chosen as $c=\delta x/\delta t$, where δx is the spacing and δt is the time step. The speed of sound is related to the lattice velocity and is $c_s=c/\sqrt{3}$. Then the equilibrium distribution function for the finite set of velocities of the D2Q9 model is:

$$f_a^{(eq)} = w_a\rho\left[1+\frac{3\mathbf{e}_a\cdot\mathbf{u}}{c^2}+\frac{9}{2}\frac{(\mathbf{e}\cdot\mathbf{u})^2}{c^4}-\frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \qquad a=0,1,...8 \tag{3.55}$$

3.3.7 A Three-Dimensional Model

The models in three dimensions are several, but the most representative is the D3Q19 model (Peng *et al*., 2003). The D3Q15 model has a small range of stability, and the D3Q27 model requires the highest computational effort; the D3Q19 model is thus a good compromise between the two models.

Peng *et al.* (2003) represent the D3Q19 models (Fig.3.5b), based on a three-dimensional regular cubic lattice, with a set of nineteen discrete velocity vectors. That set has one velocity vector at rest (in center), six velocities at nearest neighbours and twelve velocities at next-nearest neighbours.

For D3Q19 model the lattice velocities are defined as:

$$\mathbf{e}_a = \begin{Bmatrix} 0, & a = 0 \\ (\pm 1,0,0),\ (0,\pm 1,0),(0,0,\pm 1), & a = 1,2,...6 \\ (\pm 1,\pm 1,0),\ (\pm 1,0,\pm 1),(0,\pm 1,\pm 1) & a = 7,8,...18 \end{Bmatrix} \tag{3.56}$$

the weight factors are:

$$w_a = \begin{cases} \dfrac{1}{3}, & a = 0 \\ \dfrac{1}{18}, & a = 1,2,...6 \\ \dfrac{1}{36}, & a = 7,8,...18 \end{cases} \tag{3.57}$$

and the equilibrium distribution function is given by:

$$f_a^{(eq)} = w_a \rho \left[1 + \frac{3\mathbf{e}_a \cdot \mathbf{u}}{c^2} + \frac{9}{2}\frac{(\mathbf{e}_a \cdot \mathbf{u})^2}{c^4} - \frac{3}{2}\frac{\mathbf{u}^2}{c^2} \right] \qquad a = 0,1,...18 \tag{3.58}$$

3.4 Lattice Boltzmann Model for Convection Heat Transfer in Porous Media

The LBM is a methodology for fluid flow in porous media based on the Navier-Stokes equations. The first LBM simulation of the fluid flow in porous media in two dimensions was introduced by Rhotman (1988). For three dimensions, Foti and Succi (1989) adopted the convenient result of porous media permeability being a function of porosity. Cancelliere *et al.* (1990) later developed a better representation of Succi's study using the microgeometry of the materials. Nield and Bejan (1992) have studied theoretically and experimentally the problem of fluid flow and heat transfer in porous media. The fluid flow and heat transfer in porous media have been studied by Guo and Zhao (2005). The conventional numerical method of computational fluid dynamics (CFD) is based on discretization of macroscopic continuum equations. The LBM is based on microscopic models or mesoscopic kinetic equations for fluids and has been shown to have good results with complex geometries of the porous media at pore level (Zhang *et al*., 2000), also with continuum or slip flow (Chen *et al*., 1998).

In a natural porous medium, the distribution of pores is not regular in shape and size. Examples of natural porous media include sandstone, beach sand, wood, and even the human lung. The fluids flow through connected pores in the porous medium.

3.4.1 Macroscopic Equation in Porous Medium

Niathiarasu *et al*. (1997) considered that if the Boussinesq limit holds and there is local thermal equilibrium between the fluid and the solid, then the governing equation of the generalized model for incompressible fluid flow and convection heat transfer in porous media can be defined as:

$$\nabla \cdot \mathbf{u} = 0 \tag{3.59a}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\left(\frac{\mathbf{u}}{\varepsilon}\right) = -\frac{1}{\rho f}\nabla(\varepsilon p) + \nu_e \nabla^2 \mathbf{u} + \mathbf{F} \tag{3.59b}$$

$$\sigma \frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla \cdot (\alpha_m \nabla T) \tag{3.59c}$$

The coefficient σ is the ratio between the heat capacities of the solid and fluid phases and is given by:

$$\sigma = \varepsilon + (1-\varepsilon)\rho_s c_{ps} / \rho_f c_{pf} \tag{3.60}$$

$$\alpha_m = k_m / \rho_f c_{pf} \tag{3.61}$$

$$k_m = k_e + k_d \tag{3.62}$$

where:

- $\mathbf{u}$ is the volume averaged velocity
- p is the pressure
- ν_e is the effective viscosity parameter
- $\mathbf{F}$ is the total body force due to the presence of a porous medium and the other external force fields
- T is the temperature of the fluid
- ε is the porosity of the porous medium
- σ is the ratio between the heat capacities of the solid and fluid phases
- ρ_s and ρ_f are the solid and fluid densities
- c_{ps} and c_{pf} are the solid and fluid specific heats at constant pressure
- α_m is the effective thermal diffusivity

- k_m is the effective thermal conductivity
- k_e is the stagnant thermal conductivity
- k_d is the thermal conductivity due to thermal dispersion

Guo and Zhao (2005) defined the force $\mathbf{F}$:

$$\mathbf{F} = -\frac{\varepsilon\nu}{K}\mathbf{u} - \frac{\varepsilon F_\varepsilon}{\sqrt{K}}|\mathbf{u}|\mathbf{u} + \varepsilon\mathbf{G} \tag{3.63}$$

where:

- ν is viscosity of the fluid and could be different then effective viscosity ν_e
- $\mathbf{G}$ is the gravitational force, and is given by:

$$\mathbf{G} = -g\beta(T - T_0) + \mathbf{a} \tag{3.64}$$

where:

- g is the gravitational acceleration
- β is the thermal expansion coefficient
- T_0 is the reference temperature
- $\mathbf{a}$ is the acceleration due to other external force fields

Ergun (1952) developed the geometric function F_ε and the permeability K of the porous medium related to the porosity ε, based on experimental investigations. These can be expressed as the following functions:

$$F_\varepsilon = \frac{1.75}{\sqrt{150\varepsilon^3}}; \qquad K = \frac{\varepsilon^3 d_p^2}{150(1-\varepsilon)^2} \tag{3.65}$$

where d_p is the diameter of the solid particle.

After introduced the F_ε in Eq. (3.66) become:

$$\mathbf{F} = -\frac{\varepsilon\nu}{K}\mathbf{u} - \frac{1.75}{\sqrt{150\varepsilon K}}|\mathbf{u}|\mathbf{u} + \varepsilon\mathbf{G} \tag{3.66}$$

If $\varepsilon \to 1$, the governing Equations (3.62) are for fluid flows in a simple medium. Also the flow governed by Eq. (3.62) is characterized by dimensionless numbers: Darcy number (Da), the Prandtl number (*Pr*), the Raylegh number (Ra -for natural convection), the Reynolds number (*Re*-for forced or mixed convection), and the viscosity ratio (Je).

$$Da = \frac{K}{L^2}; \quad Je = \frac{\nu_e}{\nu}; \quad \mathrm{Pr} = \frac{\nu}{\alpha}; \quad Ra = \frac{g\beta\Delta TL^3}{\nu\alpha}; \quad \mathrm{Re} = \frac{LU}{\nu} \tag{3.67}$$

3.4.2 Lattice Boltzmann Equation for the Velocity Field

Guo and Zhao (2002) extended the LBE for fluid flows in a plain medium to flows in porous media; this can be expressed as:

$$f_a(\mathbf{x} + \mathbf{e}_a\delta_t, t + \delta_t) - f_a(\mathbf{x}, t) = -\frac{1}{\tau_v}\left[f_a(\mathbf{x}, t) - f_a^{(eq)}(\mathbf{x}, t)\right] + \delta_t F_a \tag{3.68}$$

where:

- $f_a(\mathbf{x}, t)$ and $f_a(\mathbf{x} + \mathbf{e}_a, t+1)$ are the non-equilibrium distribution function in a^{th} lattice direction at node $\mathbf{x}$, at time t, and node $\mathbf{x} + \mathbf{e}_a$, at time $t+1$
- $\mathbf{e}_a$ is the velocity at position $\mathbf{x}$ and time t
- δ_t is the time increment
- τ_v is the dimensionless relaxation time
- $f_a^{(eq)}$ is the equilibrium distribution function

- $\delta_t F_a$ is the total force due to the presence of the porous medium and the other external force fields

Guo and Zhao (2002) defined the equilibrium distribution function (EDF) in the DnQb model as:

$$f_a^{(eq)} = w_a \rho \left[1 + \frac{\mathbf{e}_a \cdot \mathbf{u}}{c_s^2} + \frac{\mathbf{uu} : \left(\mathbf{e}_a \mathbf{e}_a - c_s^2 \mathbf{I} \right)}{2\varepsilon c_s^4} \right] \tag{3.69}$$

where:

- w_a is the weight for the specific model
- c_s is the speed of sound for fluid, and $c_s = c/\sqrt{3}$
- $c = \delta_x / \delta_t$, and δ_x is the lattice spacing

A suitable formulation for total force F_a is:

$$F_a = w_a \rho \left(1 - \frac{1}{2\tau_v} \right) \left[\frac{\mathbf{e}_a \cdot \mathbf{F}}{c_s^2} + \frac{\mathbf{uF} : \left(\mathbf{e}_a \mathbf{e}_a - c_s^2 \mathbf{I} \right)}{\varepsilon c_s^4} \right] \tag{3.70}$$

The fluid velocity $\mathbf{u}$ is defined as:

$$\rho \mathbf{u} = \sum_a \mathbf{e}_a f_a + \frac{\delta_t}{2} \rho \mathbf{F} \tag{3.71}$$

Following to the above equations (Guo and Zhao, 2002), $\mathbf{F}$ is connected to $\mathbf{u}$ and the Eq. (3.71) is nonlinear for the velocity. Guo and Zao (2005) presented the fluid density and velocity as:

$$\rho = \sum_a f_a \tag{3.72}$$

$$\mathbf{u} = \frac{\mathbf{v}}{c_0 + \sqrt{c_0^2 + c_1 |\mathbf{v}|}} \tag{3.73}$$

where $\mathbf{v}$ is the auxiliary velocity and is given by:

$$\rho \mathbf{v} = \sum_a \mathbf{e}_a f_a + \frac{\delta_t}{2} \varepsilon \rho \mathbf{G} \tag{3.74}$$

and the parameters c_0 and c_1 are defined by:

$$c_0 = \frac{1}{2}\left(1 + \varepsilon \frac{\delta_t}{2} \frac{\nu}{K}\right) \quad ; \qquad c_1 = \varepsilon \frac{\delta_t}{2} \frac{\mathrm{F}_\varepsilon}{\sqrt{\mathrm{K}}} \tag{3.75}$$

3.4.3 Lattice Boltzmann Equation for the Temperature Field

In a fluid flow with heat transfer in a plain medium, LBMs are classified in two categories:

- multispeed models (MS)
- double-distribution function models (DDF)

The MS models are basic extensions of the isothermal LBM, which introduce a larger set of discrete velocities. The equilibrium distribution functions include higher order terms for velocity. The DDF models using the temperature field obey a simpler convection-diffusion equation. This model for fluid flows in a plain medium was studied by many researchers such as Bartoloni *et al*. (1993), Shan (1997) and Guo *et al*. (2002). Guo and Zhao (2002) modeled the evolution of the velocity field by the general lattice Boltzmann equation (GLBE) of a density distribution function and the evolution of the temperature field by the new LBE for temperature distribution function. They proposed the LBE for the thermal energy distribution as:

$$g_a\left(\mathbf{x} + \mathbf{e}_a \delta_t, t + \delta_t\right) - g_a\left(\mathbf{x}, t\right) = -\frac{1}{\tau_c}\left[g_a\left(\mathbf{x}, t\right) - g_a^{(eq)}\left(\mathbf{x}, t\right)\right] \tag{3.76}$$

where:

- $g_a(\mathbf{x}, t)$ is the thermal distribution function
- τ_c is the dimensionless relaxation time

- $g_a^{(eq)}$ is the thermal distribution function at equilibrium

He *et al*. (1998) discovered for the LBM thermal models the equilibrium distribution function for the internal energy density distribution function can be expressed as:

$$g^{(eq)} = \frac{\rho e}{(2\pi RT)^{D/2}} \exp\left(-\frac{\mathbf{e}^2}{2RT}\right)\left[\frac{\mathbf{e}^2}{DRT} + \left(\frac{\mathbf{e}^2}{DRT} - \frac{2}{D}\right)\frac{(\mathbf{e}\cdot\mathbf{u})}{RT} + \frac{(\mathbf{e}\cdot\mathbf{u})^2}{2(RT)^2} - \frac{\mathbf{u}^2}{2RT}\right] +$$
$$+ \frac{\rho e}{(2\pi RT)^{D/2}} \exp\left(-\frac{\mathbf{e}^2}{2RT}\right)\left[\left(\frac{\mathbf{e}^2}{DRT} - \frac{D+4}{D}\right)\frac{(\mathbf{e}\cdot\mathbf{u})^2}{2(RT)^2} - \left(\frac{\mathbf{e}^2}{DRT} - \frac{D+2}{D}\right)\frac{\mathbf{u}^2}{2RT}\right] \tag{3.77}$$

After eliminating the term with zeroth-through second order moment of the second term of the equation, they choose the same lattice models for the energy density distribution function as the models for the density distribution function.

Thus, for example, in different models the equations are:

- For D2Q9 - Seta *et al*. (2006) use the following equations for the discrete internal energy density equilibrium distribution:

$$g_0^{(eq)} = -\frac{2\rho e}{3}\frac{\mathbf{u}^2}{c^2} \qquad (a=0)$$

$$g_{1-4}^{(eq)} = \frac{\rho e}{9}\left[\frac{3}{2} + \frac{3}{2}\frac{\mathbf{e}_a\cdot\mathbf{u}}{c^2} + \frac{9}{2}\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{c^4} - \frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \qquad (a=1,2,3,4) \tag{3.78}$$

$$g_{5-8}^{(eq)} = \frac{\rho e}{36}\left[3 + \frac{6\mathbf{e}_a\cdot\mathbf{u}}{c^2} + \frac{9}{2}\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{c^4} - \frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \qquad (a=5,6,7,8)$$

- For D3Q19- Grucelski (2012) use the following equations for the discrete internal energy density equilibrium distribution:

$$g_0^{eq} = -\frac{\rho e}{2}\frac{\mathbf{u}^2}{c^2} \qquad (a=0)$$

$$g_{1-6}^{eq} = \frac{\rho e}{18}\left[1+\frac{\mathbf{e}_a\cdot\mathbf{u}}{c^2}+\frac{9}{2}\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{c^4}-\frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \qquad (a=1,2,3,4,5,6) \tag{3.79}$$

$$g_{7-18}^{eq} = \frac{\rho e}{36}\left[2+4\frac{\mathbf{e}_a\cdot\mathbf{u}}{c^2}+\frac{9}{2}\frac{(\mathbf{e}_a\cdot\mathbf{u})^2}{c^4}-\frac{3}{2}\frac{\mathbf{u}^2}{c^2}\right] \qquad (a=7,8,9,...18)$$

The internal energy is related to the temperature by: $e = 3RT/2$, where R is the gas constant ($R = 8.314$ [J/moll-K]). The macroscopic density, velocity and temperature are calculated by:

$$\rho = \sum_a f_a \tag{3.80}$$

$$\rho\mathbf{u} = \sum_a \mathbf{e}_a f_a \tag{3.81}$$

$$\rho(2RT/2) = \sum_a g_a \tag{3.82}$$

3.4.4 Lattice Units

In Lattice Boltzmann simulation there are three kinds of systems:

- physical system (real system)
- dimensionless system
- lattice system

Lattice Boltzmann methods used several dimensionless parameters, such as: Reynolds number, Rayleigh number, Prandtl number, etc. The real parameters (viscosity, length, temperature, etc.) have to be transformed into dimensionless parameters and then into lattice units.

A graphical representation of this relationship between systems is:

Physical System (P) ⟷ *Dimensionless System (D)* ⟷ *Lattice System (LB)*

There are two approaches that are frequently used to convert between lattice units and physical units:

a) Direct conversion

Latt (2008) noted that in direct conversion the first approach the lattice units are related to physical units through the time step Δt and the node spacing Δx. The physical flow velocity at the node is given by:

$$\mathbf{u}_P = \mathbf{u}_{LB} \frac{\Delta x}{\Delta t} \tag{3.83}$$

There $\mathbf{u}_P$ is the velocity in physical units and $\mathbf{u}_{LB}$ is the velocity in lattice units.

For physical speed of sound:

$$c_{s,P} = c_{s,LB} \frac{\Delta x}{\Delta t} \tag{3.84}$$

The kinematic viscosity in physical units is:

$$\nu_P = \nu_{LB} \frac{\Delta x^2}{\Delta t} = c_{s,LB}^2 \left(\tau - \frac{1}{2} \right) \frac{\Delta x^2}{\Delta t} = c_{s,P}^2 \left(\tau - \frac{1}{2} \right) \Delta t \tag{3.85}$$

From this the time step is:

$$\Delta t = \frac{\nu_P}{c_{s,P}^2 (\tau - 1/2)} \tag{3.86}$$

In the same way from Eq. (3.90) can have the space step:

$$\Delta x = \frac{\nu_P}{c_{s,LB} c_{s,P} (\tau - 1/2)} \tag{3.87}$$

Because at isothermal ideal gas, the pressure is proportional to the density $p = c_s^2 \rho$, that means when ρ_0 is at equilibrium (or atmospheric) density and p_0 is the corresponding pressure:

$$\frac{p}{p_0} = \frac{\rho}{\rho_0} \tag{3.88}$$

Both sides of this equation hold in both physical and lattice units. The physical pressure at any point is given as:

$$p_P = p_{0,LB} \frac{\rho_{LB}}{\rho_{0,LB}} \tag{3.89}$$

b) Dimensionless formulation

Latt (2008) described the dimensionless formulation where a physical system (P) is converted to a continuous dimensionless system (D) and then converted to a lattice system (LB). In an incompressible fluid, the density has a constant value $\rho = \rho_0$, and is not changed in time and space.

The equations need to be into a dimensionless structure. For this, Latt (2008) introduced notations l_0 as a length scale and t_0 as time scale. The physical variables for time t_P and the position vector r_P are defined as dimensionless:

$$t_D = \frac{t_P}{t_{0,P}} \quad \text{and} \quad r_D = \frac{r_P}{l_{0,P}} \tag{3.90}$$

For other variables a unit conversion is introduced based on a dimensional analysis:

$$\mathbf{u}_P = \frac{l_{0,P}}{t_{0,P}} \mathbf{u}_D, \qquad \partial_{tP} \frac{1}{t_{0,P}} \partial_{tD}, \qquad \nabla_P = \frac{1}{l_{0,P}} \nabla_D, \qquad p_P = \rho_0 \frac{l_{0,P}^2}{t_{0,P}^2} p_D \tag{3.91}$$

Using these variables into Eqs. (3.96) and (3.97), the dimensionless Navier-Stokes equations are:

$$\partial_{tD} \mathbf{u}_D + (\mathbf{u}_D \cdot \nabla_D) \mathbf{u}_D = -\nabla_D p_D + \frac{1}{\mathrm{Re}} \nabla_D^2 \mathbf{u}_D \tag{3.92}$$

$$\nabla_D \cdot \mathbf{u}_D = 0 \tag{3.93}$$

Where the dimensionless Reynolds number is given by:

$$\mathrm{Re} = \frac{l_0^2}{t_0 \nu} \tag{3.94}$$

The reference variables in the dimensionless system for $l_{0,D} = 1$ and $t_{0,D} = 1$, thus the viscosity in dimensionless system is:

$$\nu_D = 1/\mathrm{Re} \tag{3.95}$$

In the dimensionless system (D), the characteristic length and characteristic time of the system are normalized both with 1. The dimensionless system is divided into a grid with N_{ch} nodes used to resolve its characteristic length and T_{ch} is the time step used to resolve the system its characteristic time. Then space and time are given by:

$$\delta_x = 1/N_{ch} \tag{3.96}$$

$$\delta_t = 1/T_{ch} \tag{3.97}$$

The same technique can be used for velocity and viscosity to be converted between dimensionless system and lattice system.

$$\mathbf{u}_D = \frac{\delta_x}{\delta_t}\mathbf{u}_{LB} \qquad \text{so:} \qquad \mathbf{u}_{LB} = \frac{\delta_t}{\delta_x}\mathbf{u}_D \tag{3.98}$$

and:

$$\nu_D = \frac{1}{\mathrm{Re}} = \frac{\delta_x^2}{\delta_t}\nu_{LB} \qquad \text{so:} \qquad \nu_{LB} = \frac{\delta_t}{\delta_x^2}\frac{1}{\mathrm{Re}} \tag{3.99}$$

Using the reference velocity $\mathbf{u}_0 \equiv \frac{l_0}{t_0}$, can be written by definition:

$$\mathbf{u}_{0,D} = 1 \tag{3.100}$$

$$\mathbf{u}_{0,LB} = \frac{\delta_t}{\delta_x} \tag{3.101}$$

4 The Main Calculation Cycle and Boundary Conditions

LBM can be considered to be composed of four different parts: collision, streaming, application of boundary conditions and calculation of macroscopic properties. These four processes constitute the basic operating loop of a LBM program, and are discussed in detail in the following four subsections. Appendix 2 gives further details on how the various equations are specifically applied in a 'long hand' numerical example. The boundary conditions described in this chapter are specific to the velocity and temperature.

4.1 Collision Step

The collision step is considered as a redistribution of the distribution functions towards the local discretized equilibrium distribution functions and the local mass and momentum are constant. The equilibrium distribution function f^{eq} for D2Q9 is defined by Eq.3.58 in Chapter 3.

The local density ρ and velocity $\mathbf{e}$ are conserved, and the distribution functions change according to the relaxation Maxwellian rule (Chirila, 2010).

4.2 Streaming Step

Chirila (2010) explained that in the LBM, if all particles are entering the same node at the same time from different directions, there is a collision step and a new distribution of particles results.

In Lattice Boltzmann dynamics a collision step is presented as:

$$f_a^{out}(\mathbf{x},t) = f_a^{in}(\mathbf{x},t) + \Omega_a\left(f^{in}(\mathbf{x},t)\right) \tag{4.1}$$

The streaming step (propagation step) is:

$$f_a^{in}\left(\mathbf{x}+\mathbf{e}_a\cdot\delta_t, t+\delta_t\right)= f_a^{out}\left(\mathbf{x},t\right) \tag{4.2}$$

where:

- f_a^{out} is the distribution values after collision
- f_a^{in} is the values after collision and propagation, values entering the neighbouring cell
- Ω_a is the collision term, a model specific function describing the outcome of the particle collision.

Combining Eq. (4.8) and Eq. (4.9) a LBM can also be defined as:

$$f_a\left(\mathbf{x}+\mathbf{e}_a\cdot\delta_t, t+\delta_t\right)= f_a\left(\mathbf{x},t\right)+\Omega_a\left(f\left(\mathbf{x},t\right)\right) \tag{4.3}$$

In LBM the conservation laws is very important. If the number of particle is conserved in the collision process, then it is required to have:

$$\sum_{a=0}^{n} f_a^{out}\left(\mathbf{x},t\right)=\sum_{\mathbf{a=0}}^{\mathbf{n}} f_a^{in}\left(\mathbf{x},t\right) \tag{4.4}$$

The collision term must be:

$$\sum_{a=0}^{n}\Omega_a = 0 \tag{4.5}$$

Also, if the momentum is conserved, it is required to have:

$$\sum_{a=0}^{n}\mathbf{e}_a f_a^{out}\left(\mathbf{x},t\right)=\sum_{a=0}^{n}\mathbf{e}_a f_a^{in}\left(\mathbf{x},t\right) \tag{4.6}$$

Therefore:

$$\sum_{a=0}^{n}\mathbf{e}_a\Omega_a = 0 \tag{4.7}$$

Romana *et al*. (2008) give example of streaming and collision in two dimensions:

- for hexagonal lattice with seven possible velocities as in Fig. 4-2 (the arrows represent the particles, their direction correspond to the velocity , and their length is proportional f_a)

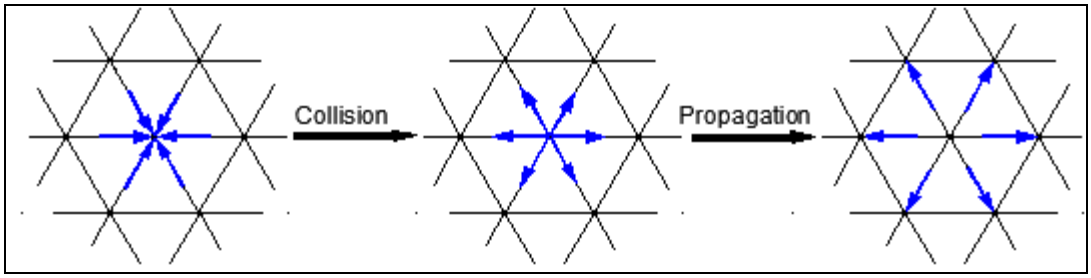


Figure 4-1 Collision and propagation phases in a LB model defined on a 2D hexagonal lattice (Romana et al., 2008)

- for square lattice with nine possible velocities as in Fig.4-1:

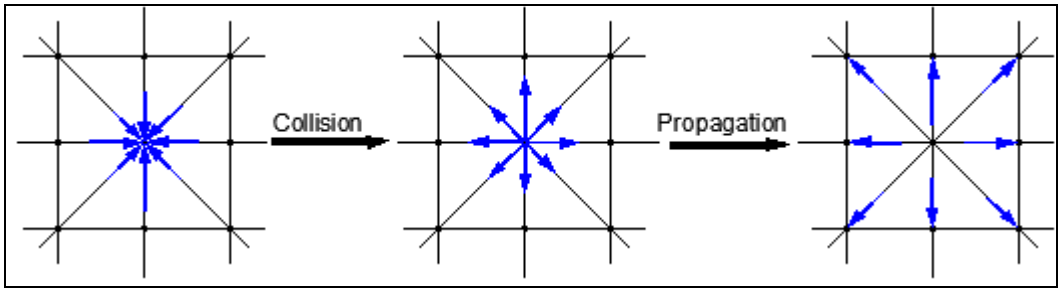


Figure 4-2 Collision and propagation phases in a LB model defined on a 2D square lattice (Romana et al., 2008)

4.3 Boundary Condition

Boundary conditions are very important part of LBM and they are necessary before computing. Many researchers such as Inamuro *et al*. (1995), Noble *et al*. (1995), Maier *et al*. (1996), Zou and He (1997) etc., proposed and investigated various boundary conditions for different kinds of simulations. In practical applications, boundary conditions can be obtained in terms of macroscopic physical variables such as ρ and $\mathbf{u}$ from mass and momentum conservations at the boundaries:

$$\rho = \sum_{a=0}^{n} f_a \tag{4.8}$$

$$\rho \mathbf{u} = \sum_{a=0}^{n} \mathbf{e}_a f_a \tag{4.9}$$

In the LBM these conditions have to be implemented through the distribution function f_a at boundary nodes according to the macroscopic boundary conditions.

The bounce-back rule has the incoming specific densities f_a at a wall node being reflected back to the original fluid node, and the direction rotated at 180^0. The boundary condition has been demonstrated to be only for first-order accuracy in time and space (Pan *et al*., 2006).

First order bounce-back is the boundary condition which defines the unknown distribution function at the boundary with the first order accuracy. It does not provide accurate results for developing flow inside the channel. Muhammed (2011) represented this as in Fig. 4-3 where the first order bounce-back type of boundary condition applied.

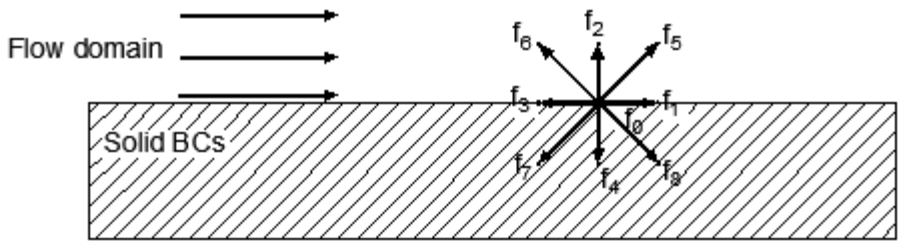


Figure 4-3 First order bounce-back of boundary condition (Muhammed, 2011)

The distribution function is reflected inside the flow as shown in the figure and the unknown distribution functions are f_5, f_2 and f_6.

$$f_5 = f_7, \quad f_2 = f_4 \quad \text{and} \quad f_6 = f_8 \tag{4.10}$$

The distribution functions f_7, f_4 and f_8 are known; they can be evaluated from streaming step and collision step.

The velocity lattice and thermal lattice arrangements at boundaries and inside the flow are represented for D2Q9 model. Almalowi and Oztekin (2012) represented the boundary condition for the velocity and the temperature; those are shown in the Fig. 4-4 and Fig. 4-5:

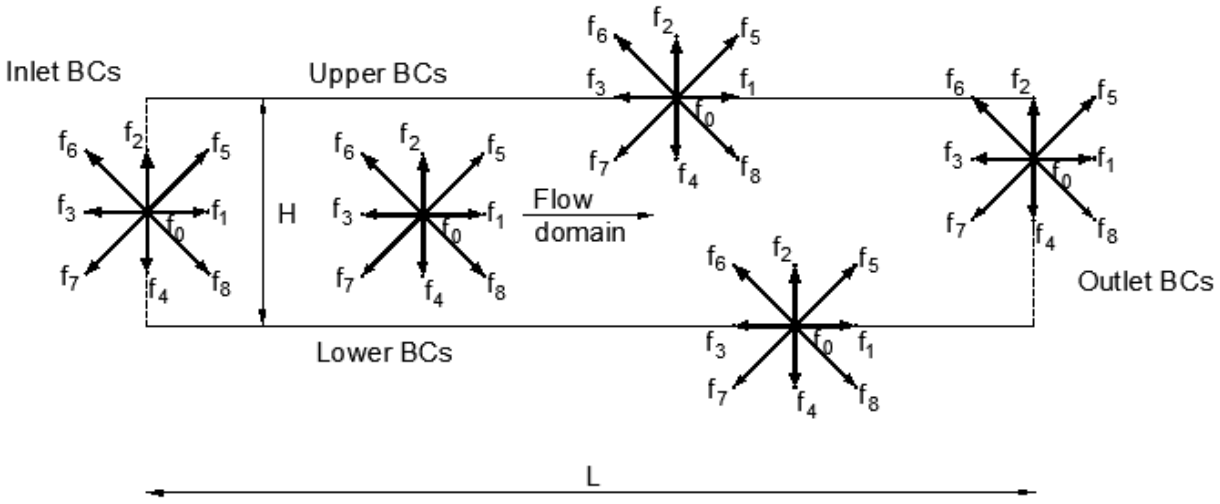


Figure 4-4 D2Q9 velocity lattice arrangements at boundaries and inside the flow domain (Almalowi and Oztekin, 2012)

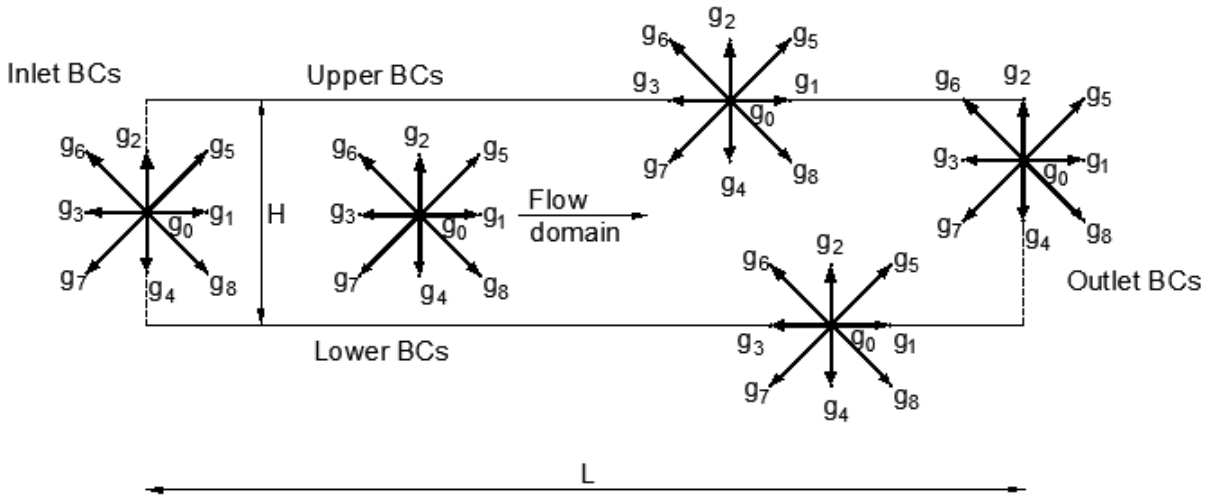


Figure 4-5 D2Q9 thermal lattice arrangements at boundaries and inside the flow domain (Almalowi and Oztekin, 2012)

Figure 4-6 represents the boundary condition for a 2D Channel flow which is used for simulation:

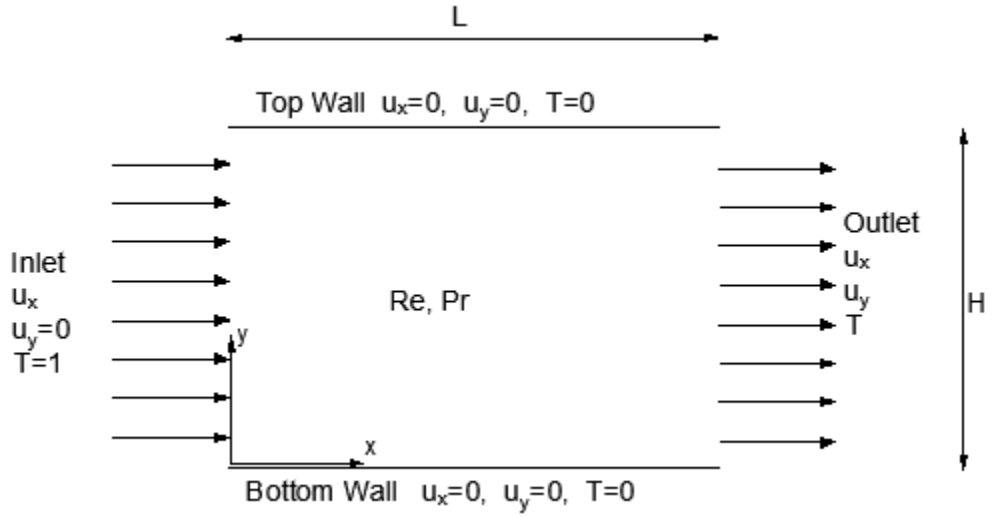


Figure 4-6 Boundary condition for 2D channel (Almalowi and Oztekin, 2012)

4.3.1 Boundary Condition for the Velocity

Inlet Velocity Boundary Condition. In this case (Fig. 4-4), it is assumed that there is no velocity component in y direction and is a non-zero velocity in x direction. There are some distribution functions unknown after the streaming step. The unknown distribution functions are f_1, f_5 and f_8. They are on the left boundary. Also the density ρ is unknown. We use the equation of the non-equilibrium function and then Eq. (4.8) and Eq. (4.9) as:

$$f_1 - f_1^{eq} = f_3 - f_3^{eq} \tag{4.11}$$

$$\rho = f_0 + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8 \tag{4.12}$$

$$\rho u_x = f_1 - f_3 + f_5 - f_6 - f_7 + f_8 \tag{4.13}$$

$$\rho u_y = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 = 0 \tag{4.14}$$

The solution for these equations offers the following results for the unknowns at the inlet boundary:

$$\rho = \frac{f_2 + f_4 + f_9 + 2(f_3 + f_6 + f_7)}{1 - u_x} \tag{4.15}$$

$$f_1 = f_3 + \frac{2}{3}\rho u_x \tag{4.16}$$

$$f_5 = f_7 + \frac{1}{2}(f_4 - f_2) + \frac{\rho u_x}{6} \tag{4.17}$$

$$f_8 = f_6 - \frac{1}{2}(f_4 - f_2) + \frac{\rho u_x}{6} \tag{4.18}$$

where u_x is the known inlet velocity.

Bottom Wall Velocity Boundary Condition. A non-slip boundary condition at the wall (Fig. 4-4), is assumed to be in equilibrium distribution function with a counter slip velocity at the

wall, equal to the wall velocity. This boundary condition can be obtained by equating the x and y velocities to zero for the nodes on required boundaries. The non-equilibrium parts of the distribution functions need to be used as:

$$f_2 - f_2^{eq} = f_4 - f_4^{eq} \tag{4.19}$$

$$\rho = \sum_{a=0}^{8} f_a = f_0 + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8 \tag{4.20}$$

$$\rho u_x = f_1 - f_3 + f_5 - f_6 - f_7 + f_8 = 0 \tag{4.21}$$

$$\rho u_y = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 = 0 \tag{4.22}$$

When u_x and u_y are set to zero, the unknown ρ multiplies with zero, then in Eq. (4.30) and Eq. (4.31) ρ becomes unimportant. In Eq. (4.28), Eq. (4.29) and Eq. (4.30) the solution gives a relation for the unknown distribution function of the bottom wall:

$$f_2 = f_4 \tag{4.23}$$

$$f_5 = f_7 - \frac{f_1 - f_3}{2} \tag{4.24}$$

$$f_6 = f_8 + \frac{f_1 - f_3}{2} \tag{4.25}$$

Top Wall Velocity Boundary Condition. Similarly with the bottom wall a boundary condition (Fig. 4-4), need to be used:

$$f_4 - f_4^{eq} = f_2 - f_2^{eq} \tag{4.26}$$

$$\rho = \sum_{a=0}^{8} f_a = f_0 + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8 \tag{4.27}$$

$$\rho u_x = f_1 - f_3 + f_5 - f_6 - f_7 + f_8 = 0 \tag{4.28}$$

$$\rho u_y = f_4 - f_2 - f_5 - f_6 + f_7 + f_8 = 0 \tag{4.29}$$

The solutions are:

$$f_4 = f_2 \tag{4.30}$$

$$f_7 = f_5 + \frac{f_1 - f_3}{2} \tag{4.31}$$

$$f_8 = f_6 - \frac{f_1 - f_3}{2} \tag{4.32}$$

Outlet Velocity Boundary Condition. The velocity at the outlet (Fig. 4-4), is unknown, contrary to the inlet boundary. The unknown distribution functions are f_3, f_6 and f_7. They are on the left outlet boundary. Also the density ρ is unknown. In this case the velocity in the y direction is assumed no more varying and the equations are:

$$f_3 - f_3^{eq} = f_1 - f_1^{eq} \tag{4.33}$$

$$\rho = f_0 + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8 \tag{4.34}$$

$$\rho u_x = f_3 - f_1 - f_5 + f_6 + f_7 - f_8 \tag{4.35}$$

$$\rho u_y = f_2 - f_4 + f_5 + f_6 - f_7 - f_8 = 0 \tag{4.36}$$

The solution for these equations offers the following results for the unknowns at the outlet boundary:

$$\rho = \frac{f_2 + f_4 + f_9 + 2(f_1 + f_5 + f_8)}{1 + u_x} \tag{4.37}$$

$$f_3 = f_1 + \frac{2}{3}\rho u_x \tag{4.38}$$

$$f_6 = f_8 + \frac{1}{2}(f_4 - f_2) - \frac{\rho u_x}{6} \tag{4.39}$$

$$f_7 = f_5 - \frac{1}{2}(f_4 - f_2) - \frac{\rho u_x}{6} \tag{4.40}$$

4.3.2 Boundary Condition for the Temperature

In the model (Fig. 4-5) is considered the temperature from outside as hot temperature (notation is 1 in the code) and at the walls as cold temperature (notation is 0 in the code).

$$T = \sum_{g=0}^{8} g_a = g_0 + g_1 + g_2 + g_3 + g_4 + g_5 + g_6 + g_7 + g_8 \tag{4.41}$$

Inlet Temperature Boundary Condition. The temperature boundary conditions (Fig. 4-5) applied use the equality of non-equilibrium distribution functions. The equations are:

$$g_1 = T(w(1) + w(3)) - g_3 \tag{4.42}$$

$$g_5 = T(w(5) + w(7)) - g_7 \tag{4.43}$$

$$g_8 = T(w(8) + w(6)) - g_6 \tag{4.44}$$

Bottom Wall Temperature Boundary Condition. The equations are:

$$g_2 = T(w(2) + w(4)) - g_4 \tag{4.45}$$

$$g_5 = T(w(5) + w(7)) - g_7 \tag{4.46}$$

$$g_6 = T(w(6) + w(8)) - g_8 \tag{4.47}$$

Top Wall Temperature Boundary Condition. The equations are:

$$g_4 = T(w(2) + w(4)) - g_2 \tag{4.48}$$

$$g_7 = T(w(5) + w(7)) - g_5 \tag{4.49}$$

$$g_8 = T(w(6) + w(8)) - g_6 \tag{4.50}$$

Outlet Temperature Boundary Condition. The equations are:

$$g_{3,X} = 2g_{3,X-1} - g_{3,X-2} \tag{4.51}$$

$$g_{6,X} = 2g_{6,X-1} - g_{6,X-2} \tag{4.52}$$

$$g_{7,X} = 2g_{7,X-1} - g_{7,X-2} \tag{4.53}$$

4.4 Definition of Macroscopic Properties

The properties of densities and velocities for flow are calculated using the mass and momentum conservation at each node:

$$\rho(\mathbf{x},t) = \sum_{a=0}^{8} f_a(\mathbf{x},t) \tag{4.54}$$

$$\mathbf{u}(\mathbf{x},t) = \frac{\sum_{a=0}^{8} e_a f_a(\mathbf{x},t)}{\rho(\mathbf{x},t)} \tag{4.55}$$

The temperature is calculated using the thermal distribution function:

$$T(\mathbf{x},t) = \sum_{a=0}^{8} g_a(\mathbf{x},t) \tag{4.56}$$

The LBM worked example for D2Q9 model is presented in Appendix 2.

5 Physical Experiments for Verification

To verify the simulation results three physical model experiments were designed and executed: rectangular box model, circular plastic pipe model, and circular aluminum pipe model. In these experiments gravel as a porous medium was used. The temperature evolution during the experiments were analysed from the images taken by a thermal camera. Taken together, the three experiments comprised a progression towards the boundary conditions that could be accommodated by the code, at this stage in its development. This is why the experiments are presented before the simulations.

5.1 Rectangular Box with Varying Boundary Temperature

An experiment was conducted to evaluate and validate the LBM model built with MATLAB code. The experiment aimed to show the effects of air flow through on heat transfer with porous media.

The experiment consisted of a box filled with granite gravel, an inlet region, an outlet region, a flow meter, a hair dryer and a thermal camera. The hair dryer was used to supply hot air. The box had cardboard walls with dimensions 12 cm x 12 cm x 32 cm, as in Fig. 5-1. In this experiment, gravel has been used as the porous medium, and was assumed as an isotropic and homogeneous porous media. From inspection of Fig. 5-2, the average size of gravel used in the experiment was estimated at 2 cm.

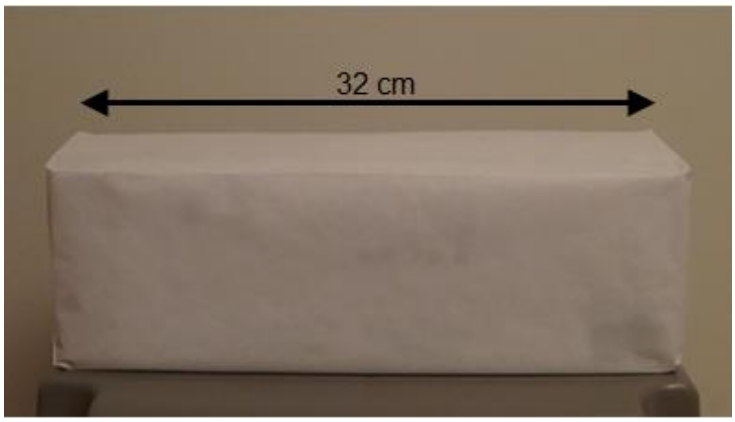


Figure 5-1 Cardboard box filled with gravel

Figure 5-2 Granite gravel

At the beginning of the experiment, all the walls were the same temperature, considered 'cold temperature', and the hot air flow was at the inlet region. The initial temperature of the box and gravel was 18.8° C. The hair dryer blew air at 52° C and the air velocity was approximately 0.2 m/s. The fluid flowed through the inlet with assumed uniform velocity, passed through porous media and exited through the outlet. During the transit of the air flow through the gravel, a heat transfer occurred between the two media.

The schematic of the experiment is presented in the Figure 5-3:

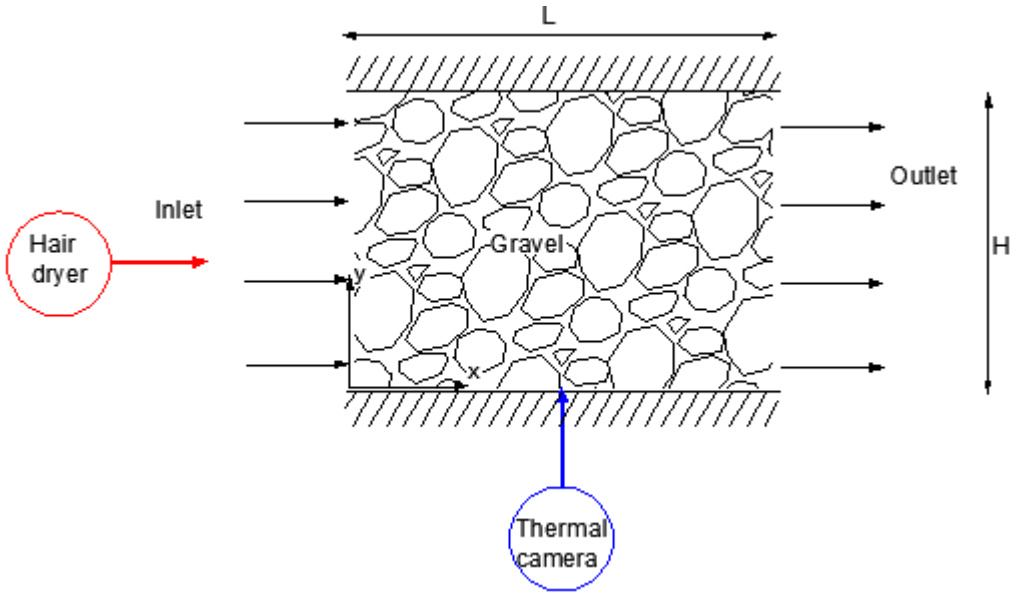


Figure 5-3 Validation experiment with rectangular box

The temperature measurements in experiments were done with the thermal camera model Flir E50. The emissivity (ε-is the relative ability of its surface to emit energy by radiation) of surface was set at 0.83. The camera viewed the side of the box, normal to flow direction. The images (Fig. 5-4a to f) were taken during the experiment at different times (time stamped on the images) with a thermal camera. In this imagery, the temperatures were represented by a palette of colours varying from light yellow to dark purple. The coldest temperature was represented by the dark purple colour. With increases in temperature of the media over time, the colours of thermal image became brighter to yellow. After the pictures were taken, the images were analysed and the temperature field examined in detail.

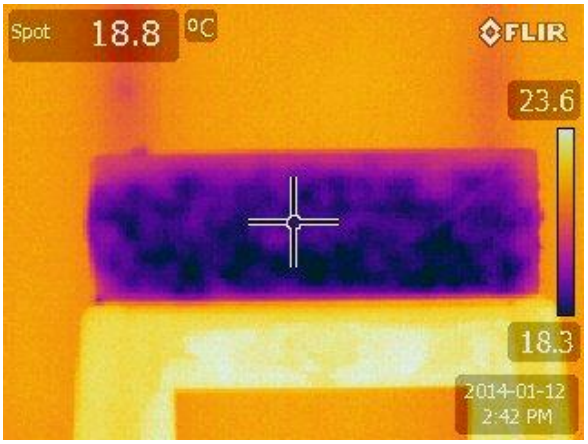
On the top left corner of the images, the spot temperature of the box surface was presented (in the cross hairs). The spot was located approximately on the middle of box. A legend of temperature was presented on the right hand side of the pictures. Unfortunately, the temperature scale changes for each time, making it harder to interpret the temperature field. The temperatures

change because the hot air was gone through inside the box; there was a heat transfer phenomenon, leading to a change in temperature of the gravel, and then the air exited through the outlet. In the bottom right hand corner of each image the time when the pictures had been taken was presented. In Fig. 5-4a, the high temperature (23.6°C) was at the plastic table where the box was placed. The temperature of the room in which the experiment took place started at 20°C and rose to 22.4°C by the end of the experiment (Fig. 5-4f). The box filled with gravel was colder (18.3°C) than the room at the beginning of the experiment because was taken from cold room. All those temperature were correlated with the colours of the legend. The temperatures of the spot during the experiment ranged between 18.8° C and 36.6° C.

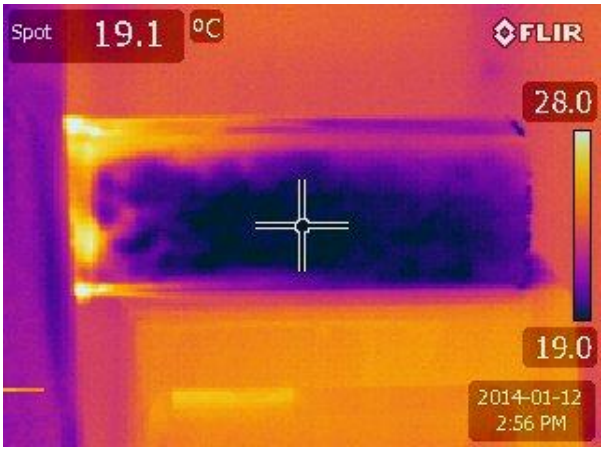
The pictures presented in Fig. 5-4a and Fig. 5-4b were taken at 14 minutes apart and the differences of the spot temperatures was 0.3° C. The temperature legend displayed in Fig. 5-4a ranged between 18.3°C and 23.6°C with a span of 5.3 °C. In Fig. 5-4b the temperature ranged between 19°C and 28°C with a span of 9°C. In Fig. 5-4b the heat transfer process was observed to start to progress along the surface of the box. After a further 32 minutes (Fig. 5-4 c), the hot surface temperature was visible progressing to the right side of the box toward the outlet. The temperature difference of the spot ranged between 21°C and 38.7 °C with a span of 17.7 °C.

In Fig. 5-4d, the difference ranged between 21.9°C and 51.4°C with a span of 29.5 °C after 20 more minutes. In the second last image (Fig. 5-4 e) the hot temperature was present over almost the whole surface of the box, the last cold zone was situated close the exit. The spot temperature was at 30.6°C.

In Fig. 5-4f shows the temperature distribution on the surface, after it had become steady.

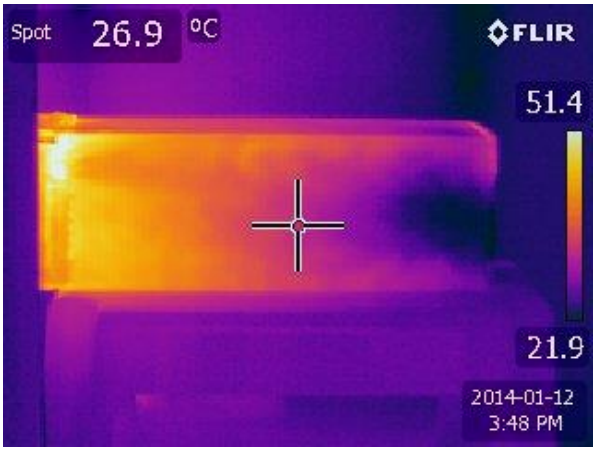


(a)

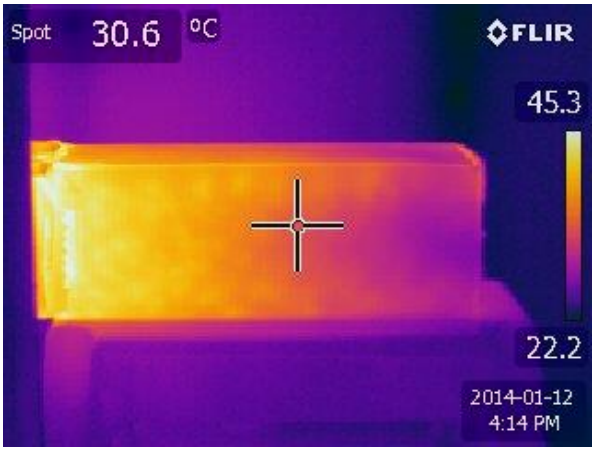


(b)

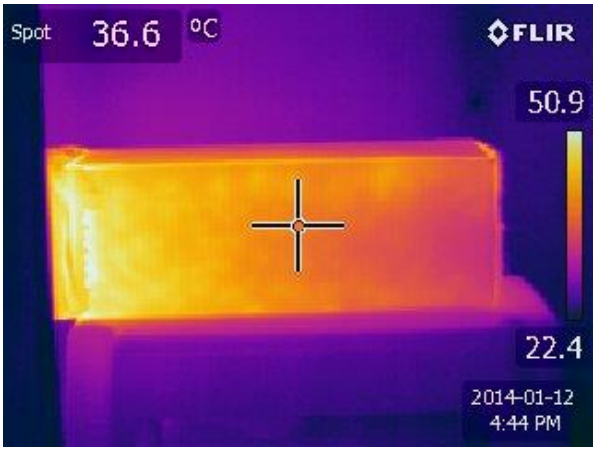
(c)



(d)



(e)



(f)

Figure 5-4 Thermal camera images for rectangular box experiment

The air was warmer along the edges of box because the pores were coupled at the contact of gravel with rectangular edges, leaving an open channel, through which more warm air could flow, compared with the circular edges (Fig. 5-5a and Fig. 5-5b; blue colour).

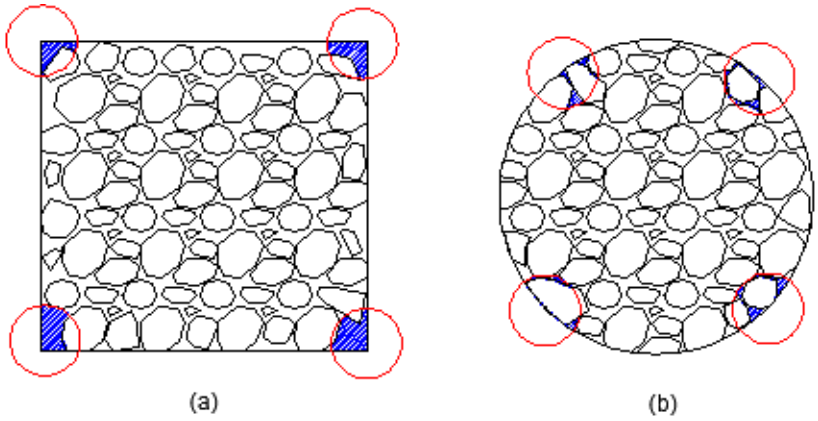


Figure 5-5 (a)-Pores at rectangular edges; (b)-Pores at circular edges

The evolution temperature in time was presented in Fig. 5-6.

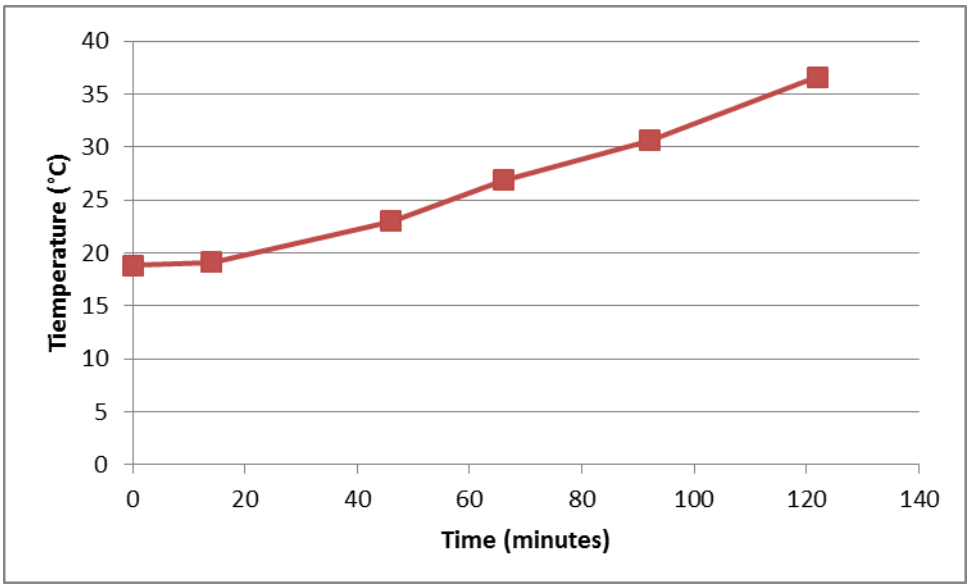


Figure 5-6 Temperature-time graph for rectangular box experiment

The temperature was increased in time, but the temperature was not right distribution temperature for the porous media, because the thermal camera measured just the surface temperature. The isothermal condition at the walls, in this experiment, was not represented well. For this reason it was required to redesign another experiment with appropriate condition as in the simulation.

5.2 Circular Plastic Pipe with Constant Boundary Temperature

In this experiment we used a plastic pipe filled with gravel. In an attempt to obtain the isothermal condition at the boundary, the pipe was immersed in water with melting ice, which maintained a temperature at the walls to 0°C. The temperature measured was for the porous media in this experiment.

This experiment consisted of a plastic pipe filled with gravel, plastic pipe used to blow air flow, an inlet region, an outlet region, a flow meter, a container filled with mixture of water and ice, a hair dryer and a thermal camera. The plastic pipe was ABS type. The gravel used in this experiment with fragment dimensions of approximately 2 cm, was considered as a continuum with an estimated porosity 0.44. The dimensions of pipe, filled with gravel, were 10 cm diameter and 32 cm length, as presented in Fig. 5-7 and in Fig. 5-8. The thermal camera and hair dryer were used as before. The position of the thermal camera, in this experiment, was over the exit of the flow (outlet), shown in Fig. 5-9, with the view of the camera parallel to, and against, the direction of the flow.

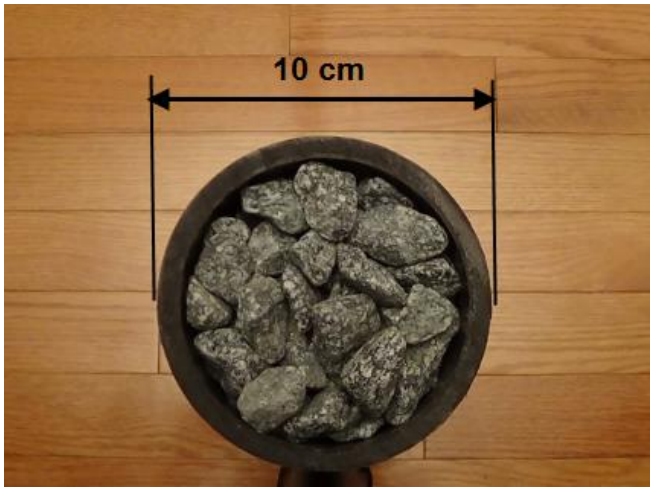


Figure 5-7 Outlet of the circular plastic pipe experiment

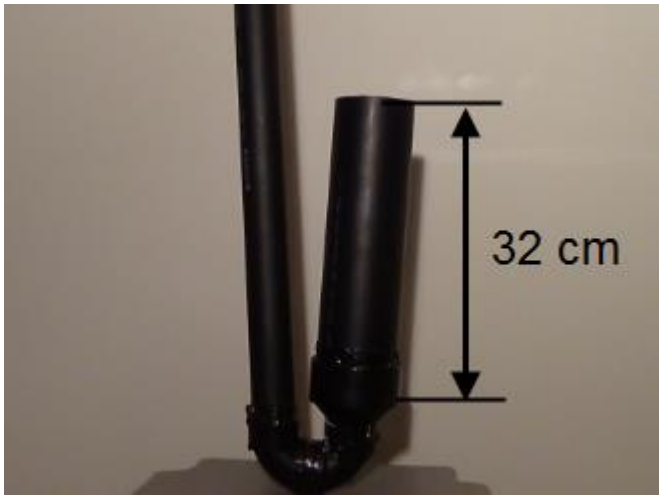


Figure 5-8 Plastic pipe filled with gravel, before immersion in water tank

The assembly of pipe and porous medium was immersed in a container filled with water and continuing melting of ice maintained the water at 0° C, the idea being to maintain the pipe walls isothermal. The temperature of the air flow was 35° C. The air velocity was measured approximately as 0.2 m/s and it was assumed uniform. The fluid flowed through the inlet and passed through porous media, and then exited through the outlet. A schematic of the experiment is presented in the Fig. 5-9.

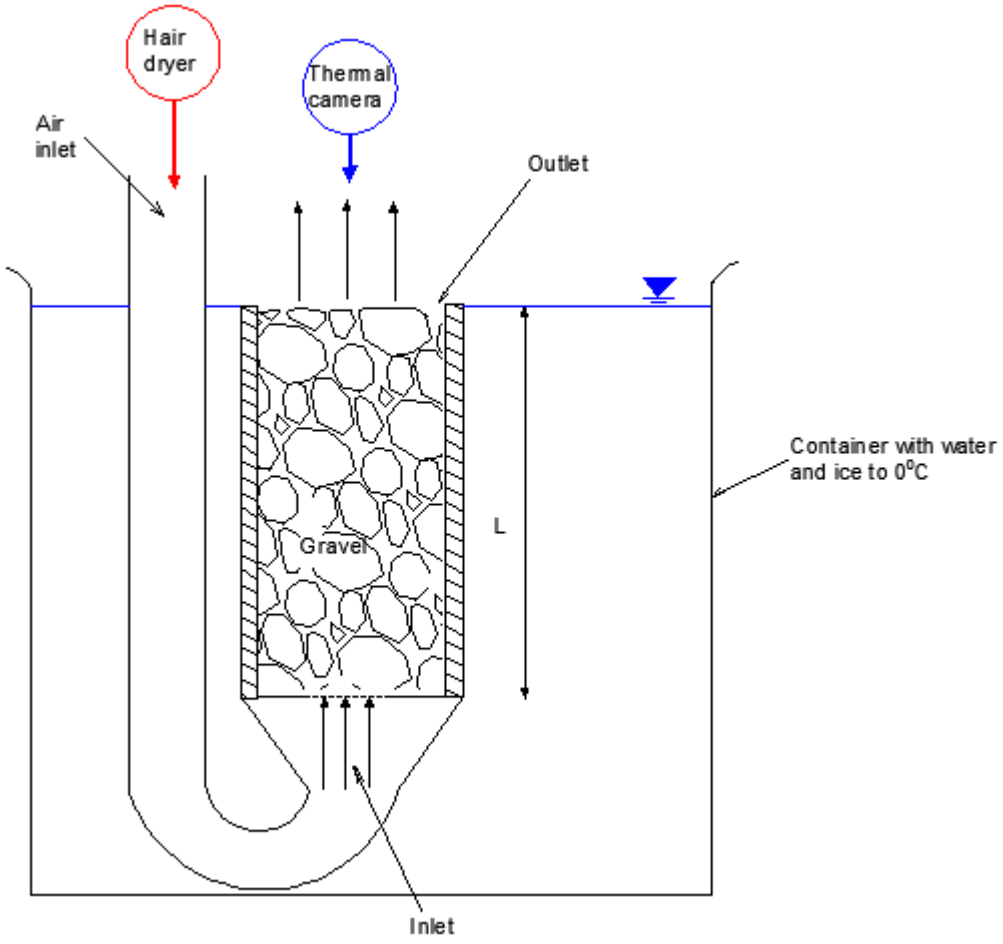
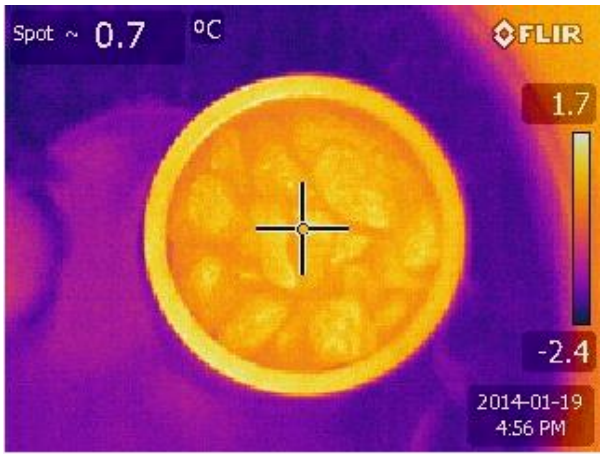


Figure 5-9 Validation of the circular pipe experiment

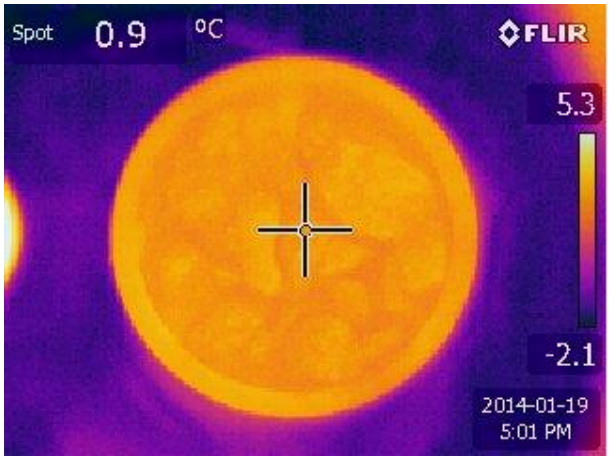
Further thermal imagery (Fig. 5-10) was taken of the outlet at different times (which are shown in each image in the lower right hand corner). The model of thermal camera was again the Flir E50. The emissivity of surface was setup at 0.83. For the measurement of velocity we used the smoke tube method. In this method a puff of smoke was released at the exit of pipe and it was timed until it traveled 50 cm. The velocity was calculated having the time and distance.

The gravel temperature was 0°C at the beginning of the experiment. In the container, the water was maintained at 0°C temperature close the pipe. The lower temperature was from the ice

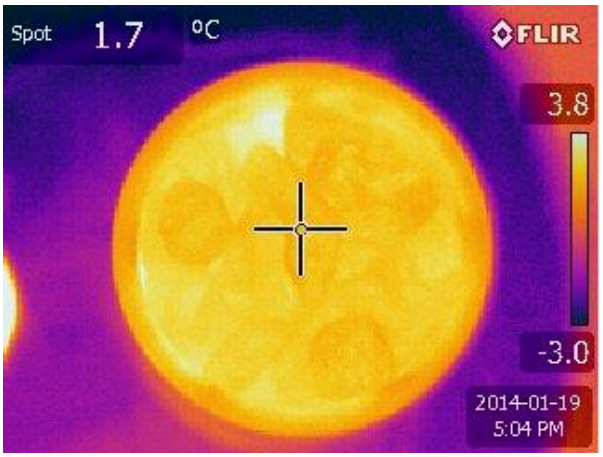
represented as dark purple colour in the picture taken by the camera (Fig. 5-10a, b, c, d, e, f, g). The temperatures of the spot during the experiment were ranging between 0.7° C and 19.5° C.



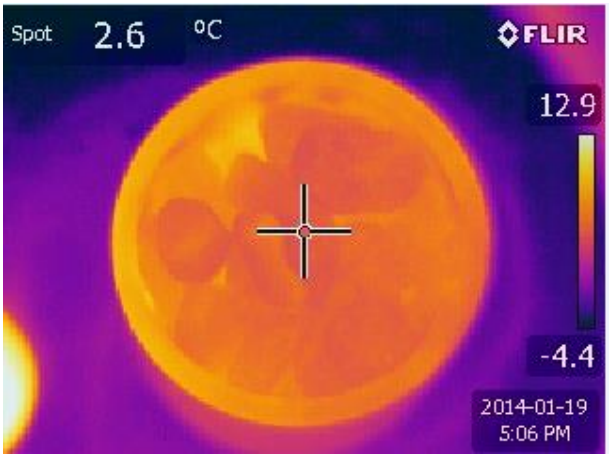
(a)



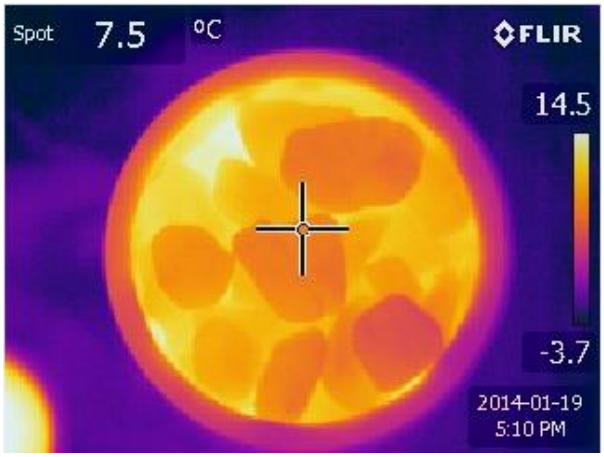
(b)

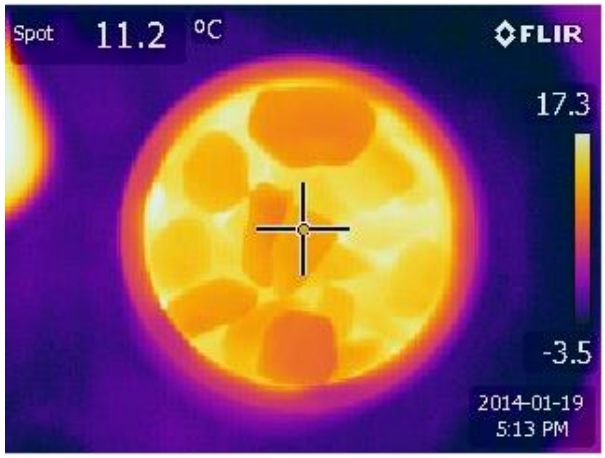


(c)

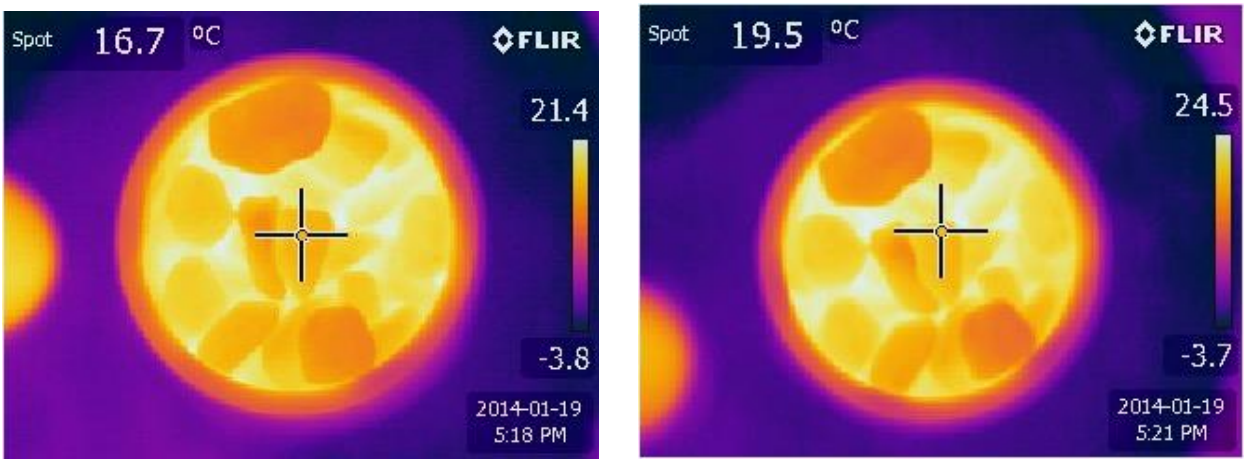


(d)





(e) (f)



(g) (h)

Figure 5-10 Thermal camera images at the outlet at plastic pipe experience

The duration of the experiment was 25 minutes. The temperatures were represented by the same palette of colours as in the previous experiment's pictures. The coldest temperature was represented by dark purple colour and the hottest temperature become brighter to yellow colour. On the top left hand corner of pictures the spot temperature was displayed. The temperature measured by the thermal camera was located approximately in the center of the pipe at the exit (outlet). A legend of temperature is presented on the right hand side of the images.

In Fig. 5-10a the temperature in the centre of outlet pipe was indicated by the spot temperature as 0.7°C. In the water were presented pieces of ice, so the coldest temperature was -2.4°C (represented by temperatures scale). The highest temperature was at the room wall, 1.7°C.

The temperature measured after 4 minutes increased to 0.9°C (Fig. 5-10b). There the ice was melting and the cold temperature changed at -2.1°C. On the scale was presented the highest temperature at 5.3°C, which was for the air inlet pipe on the left hand side. In the Fig. 5-10c the temperature increased to 1.7°C. The image presented in Fig. 5-10b and Fig. 5-10c were taken at 3 minutes interval. The corresponding difference of temperature was 0.8°C. In Fig. 5-10d was

presented a temperature increase with 0.9°C in a 2 minutes period. In Fig. 5-10e was observed a spot temperature of 7.5°C, which means an increase of 4.9°C than in Fig. 5-10d. In Fig. 5-10f the spot temperature increased with 3.7°C in 3 minutes interval to 11.2°C. In Fig. 5-10g the spot temperature was 16.7°C, which means an increase with 5.5°C in 5 minutes. In the last image (Fig. 5-10f), after 3 more minutes the spot temperature was 19.5°C; and the highest temperature in the field of view was 24.5°C. During the experiment, ice was added to maintain the temperature of the water at 0°C.

The temperature-time dependence for this experiment is represented in Fig. 5-11:

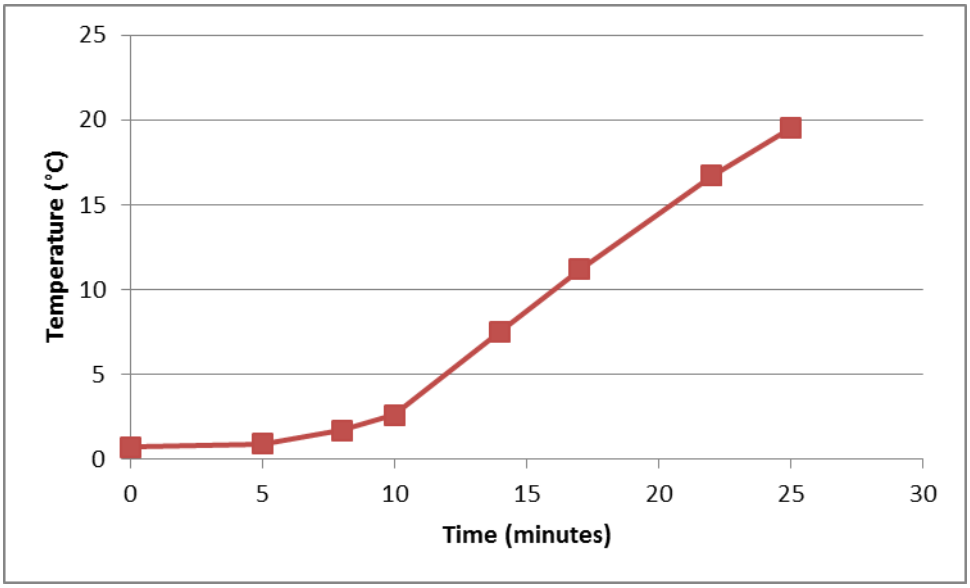


Figure 5-11 Temperature-time graph for circular plastic pipe experiment

The advantage of this experiment is that the porous media in the pipe are not open channels as in the square box. The other advantage is that in this case the spot temperature was located in the center of the section of porous media not on the external wall.

However, because the ABS pipe has thermally insulating properties, the temperature at the boundary confining the gravel was not maintained at 0°, despite adding the ice and having the

pipe in a waterbath (Fig. 5-9 and Fig. 5-10). The temperature of the inner wall varied throughout the experiment, and was clearly not isothermal.

5.3 Circular Aluminum Pipe with Constant Boundary Temperature

The next experiments used an aluminum pipe to retain the gravel because aluminum has very good thermal conductor properties. The experiments expected to show the effects of air flow through and heat transfer in porous media with isothermal boundaries. The assembly was immersed in water with ice and the temperature at the aluminum pipe was maintained at 0°C and better approximated isothermal conditions. The experimental procedure was carried out in the same manner as in previous cases. In this experiment a plastic pipe was used (with insulate properties) to blow hot air at the inlet region of the aluminum pipe filled with gravel. The aluminum pipe dimensions were 32 cm length and 10 cm diameter, as in Fig. 5-12 and Fig. 5-13. In all experiments, gravel with the same physical properties was used. The hair dryer blew the hot air inside the pipe. The thermal camera was used to take the pictures at the outlet where all changes of the temperatures in time through porous media are observed. The thermal camera model was the same Flir E50. The emissivity of surface was setup at 0.92. The reason for different emissivity used in this experiment compared to previous ones is that in the room was lower illumination, and the emissivity needed to be adjusted.

The schematic experiment was the same as before (Fig. 5-9).

The general assembly of the gravel-filled aluminum pipe, before immersion in the water tank, is presented in Fig. 5-13:

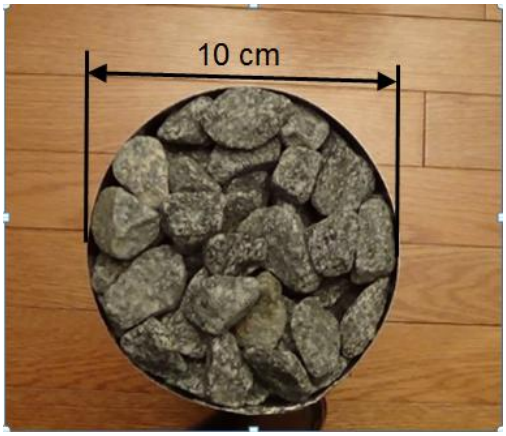


Figure 5-12 Outlet of the circular aluminum pipe experiment

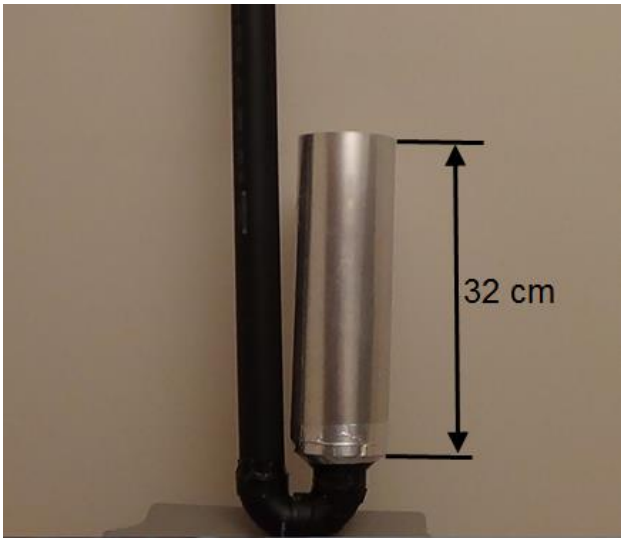


Figure 5-13 Aluminum pipe filled with gravel, before immersion in water tank

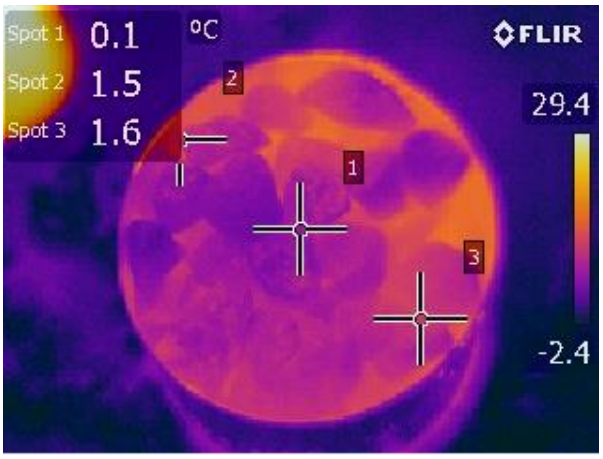
Three different experiments were conducted using the aluminum pipe filled with gravel.

5.3.1 Circular Aluminum Pipe - Experiment 1

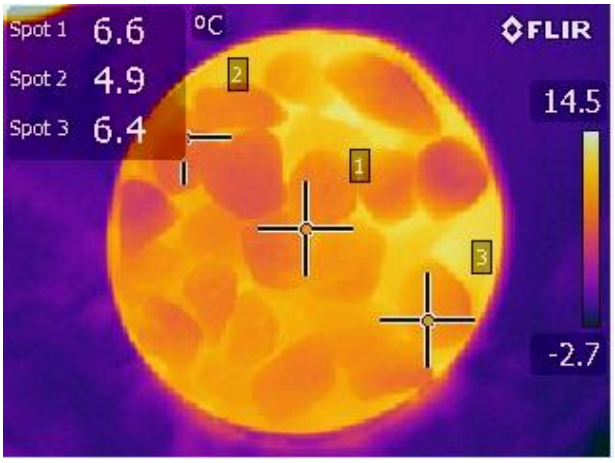
The images, taken by the thermal camera in this experiment, were presented in Fig. 5-14. Thermal camera was setup to measure temperatures for three different spots; spot 1 placed in the centre exit pipe, and the other two, spot 2 and spot 3 are placed symmetrical distance of the

spot1. The distance between spots were 4cm. All spots were presented on the top left hand corner of pictures. The emissivity (ε) was introduced in the images in the bottom left hand corner and was chosen from the thermal camera at value of 0.92 for the gravel (the same as before experiment). The legend of temperature was presented on the right hand side of the pictures as before. During the experiment, the water was maintained at 0°C. After the images were taken, the temperatures were examined in detail. The gravel temperature was at the beginning of the experiment under 0°C and blew little hot air, and started to take the pictures when the gravel had approximately 0°C.

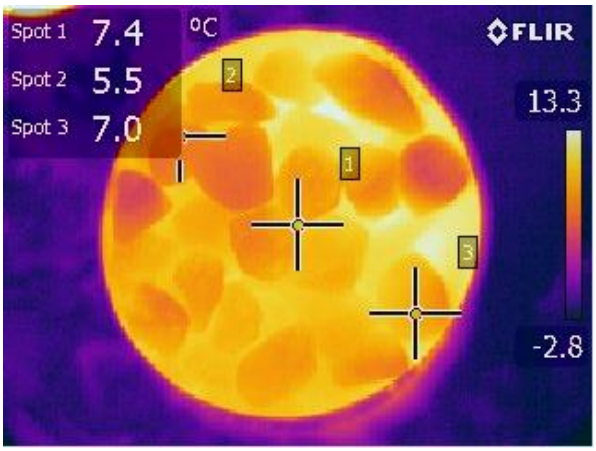
The imagery of experiment are presented in Fig. 5-14a, b, c, d, e, f, g, h, i, j:



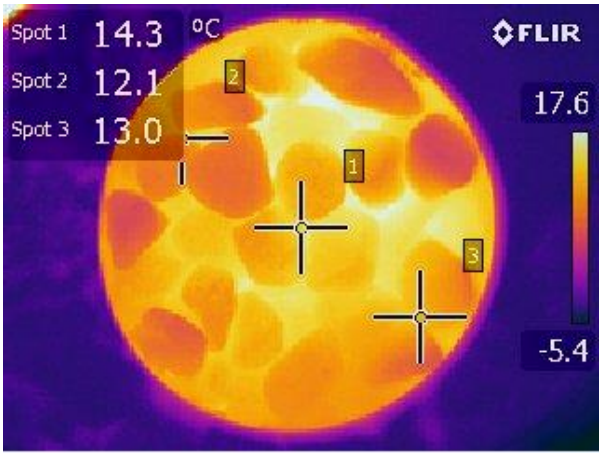
(a) time=0



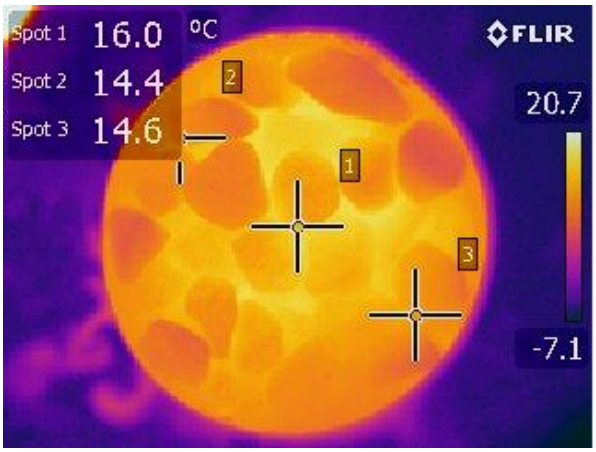
(b) time=4



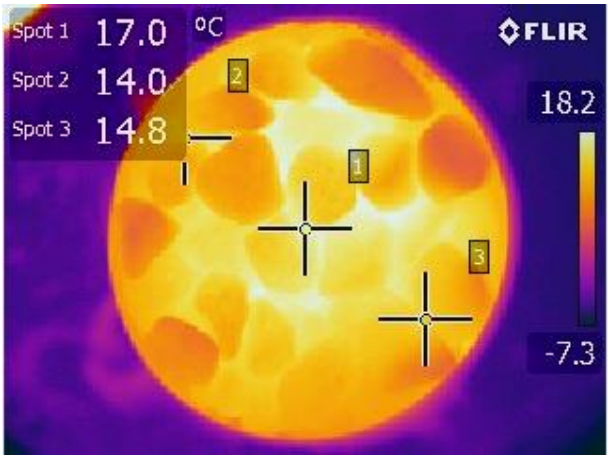
(c) time=6



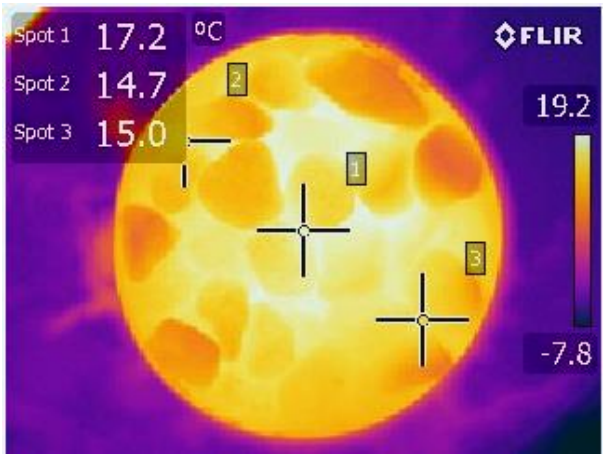
(d) time=10



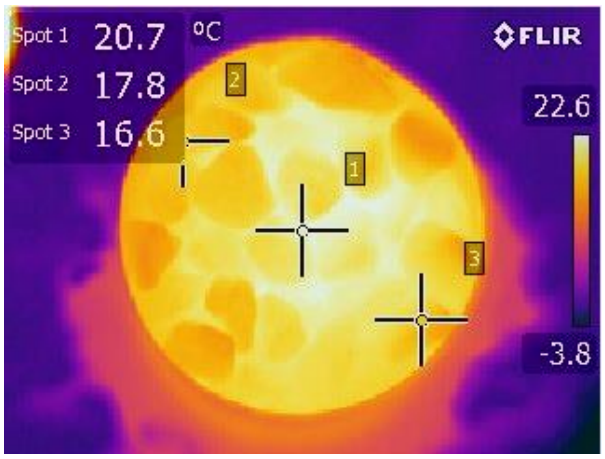
(e) time=12



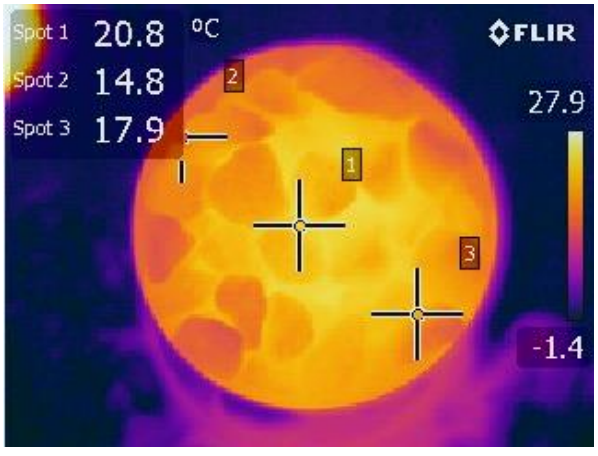
(f) time=14



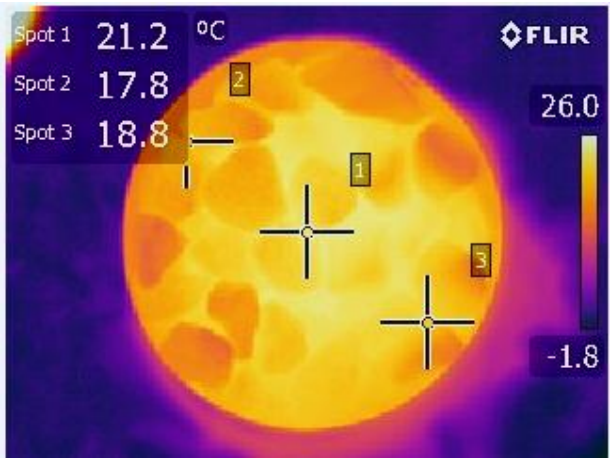
(g) time=15



(h) time=18



(i) time=20



(j) time=24

Figure 5-14 Thermal camera images at the outlet for aluminum pipe-experiment 1

In Fig. 5-14a the temperature spot 1 was to 0.1°C, the spot 2 was to 1.5°C, and the spot 3 was to 1.6°C. The coldest temperature as seen on the temperature scale was -2.4°C that represent melting ice floating on water surface. The hair dryer blew air at 45°C temperature. The scale indicated the highest temperature, which come from the air inlet pipe. After 4 minutes (Fig. 5-14b), the temperature increased to 6.6 °C at spot 1, 4.9 °C at spot 2, and 6.4 °C at spot 3. In Fig. 5-14c the central temperature was increased at 7.4 °C. On the scale of temperature also increased at 13.3°C for highest temperature. The lower temperature come from the melting ice in the water, and on the scale of temperature was represented by the dark purple colour. The difference in the next image (Fig.5-14d), taken by the thermal camera after more 4 minutes, were the temperatures: 14.3°C at spot 1, 12.1°C at spot 2, and 13°C at spot 3. In addition, the highest temperature showed by the temperature scale increased at 17.6°C. The temperature continued to increase, in Fig. 5-14e the temperature of the spots showed that; to 16°C at spot 1, to 14.4°C at spot 2, and to 14.8°C at spot 3. The scale of temperature showed the highest temperature to 20.7°C. In Fig. 5-14f the central temperature displayed at the spot 1 was 17°C, and the other two temperatures at the spot 2 and spot 3 were 14°C and respectively 14.8°C. These temperatures were taken after 2 minutes than before. After 2 other minutes the temperatures increased to 17.2°C at spot 1, to 14.7°C at spot 2, and to 15°C at spot 3 (Fig. 5-14g). The highest temperature displayed by the scale of temperature was 19.2°C. In the Fig. 5-14h, the increased value of the temperatures were at spot 1 with 3.5°C, at spot 2 with 3.1°C, and at spot 3 with 1.6°C during the 3 minutes interval (between Fig. 5-14g and Fig. 5-14h). Also the highest temperature was increased to 22.6°C. The picture presented in Fig. 5-14i and Fig. 5-14j were taken at 4 minutes interval. The maximum temperature at the central spot reached to 21.2°C (Fig. 5-14j), and to

17.8°C at spot 2, respectively to 18.8°C at spot 3. The image showed the highest temperature at 26°C.

The evolution of temperature distribution in the experiment 1 was presented in Fig. 5-15. There was observed that the central temperatures (showed by the temperature spot 1) had a higher value than the peripheral temperatures (showed by the temperatures spot 2 and spot 3).

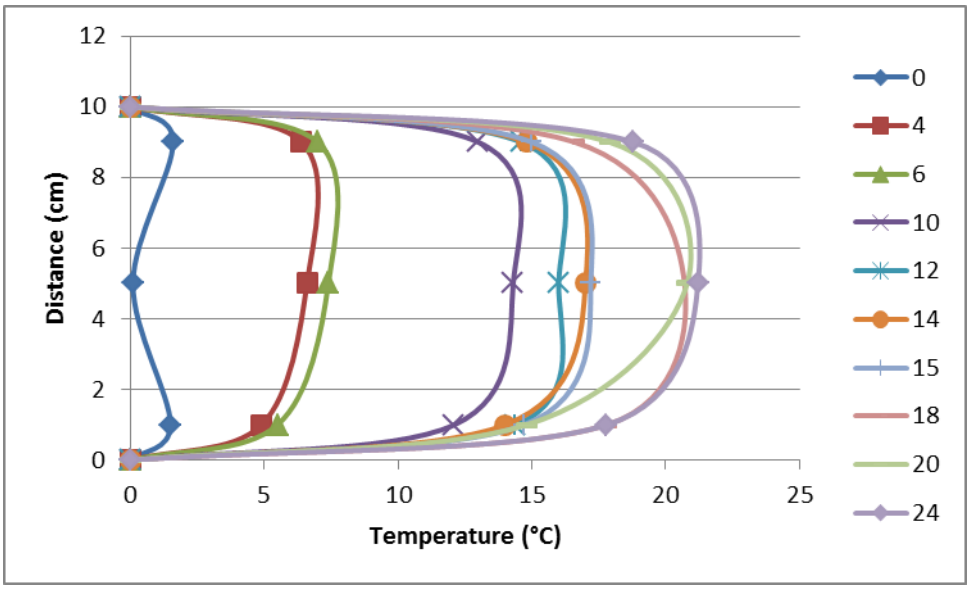


Figure 5-15 Temperature-profile graph for circular aluminum pipe –experiment 1

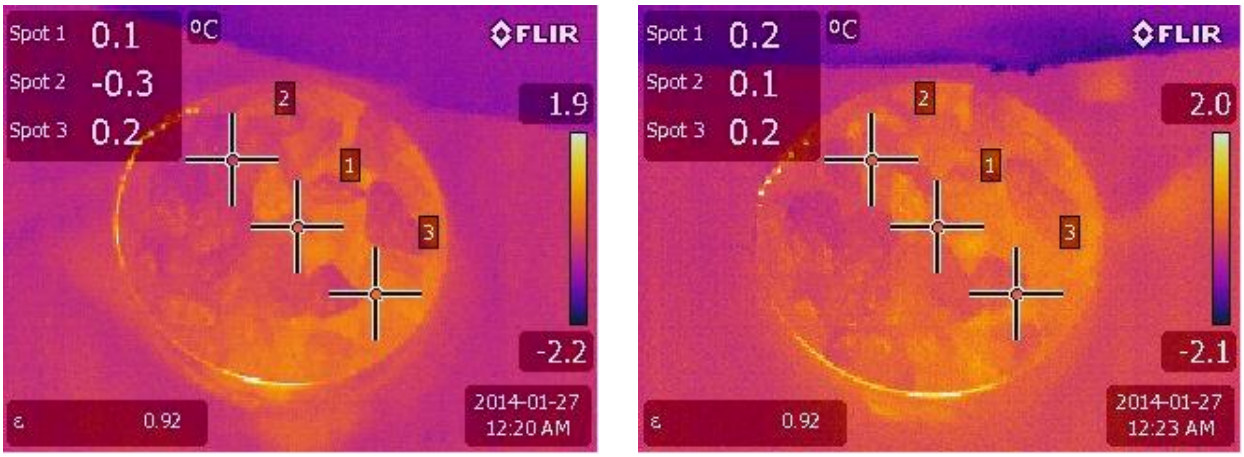
In this experiment, the temperatures varied between 0°C to 21.2°C in 25 minutes interval. After 25 minutes porous media started to steam because a small water leakage . Inside the plastic pipe was a small quantity of water and the temperatures became unstable.

5.3.2 Circular Aluminum Pipe - Experiment 2

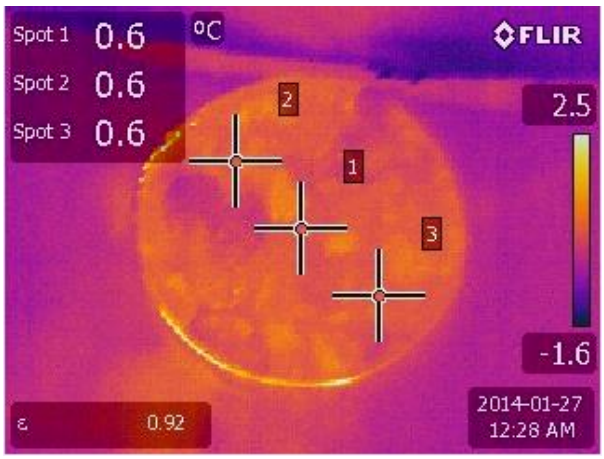
The second experiment was done for a longer period and with no leaking inside the pipe. In this experiment we used PVC cement which glued the pieces of plastic pipe together. The plastic and

aluminum pipes were bonded using silicon and duck tape. The distances between the temperatures of spots were changed at 3 cm, for a better distribution along the diameter. The temperature of the gravel at beginning was 0°C. In the bottom right hand corner of each image the time when the pictures had been taken was presented. In the container filled with water and melting ice was immersed the assembly of pipes. The hair dryer blew the hot air through the inlet of pipe at 37°C. The air velocity was approximately measured at 0.2 m/s. The temperature measurements were done with the same thermal camera model Flir E50. In the bottom left hand corner of each image was presented the emissivity of the gravel (0.92). A legend of temperature was presented in the right side of the images, with lower temperature as dark purple colour, and higher temperature light yellow.

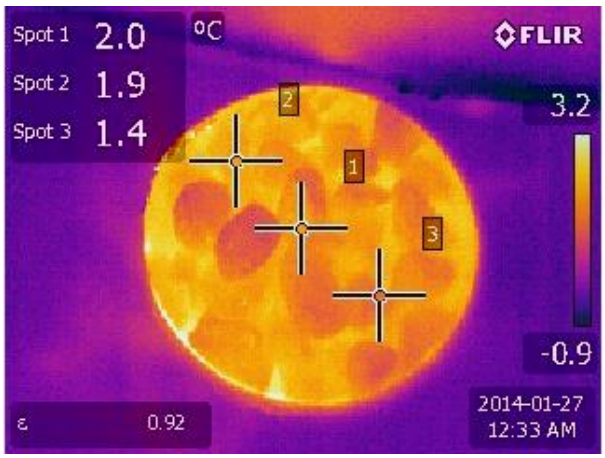
Fig. 5-16a, b, c, d, e, f, e, f show the evolution of temperature in time through porous media:



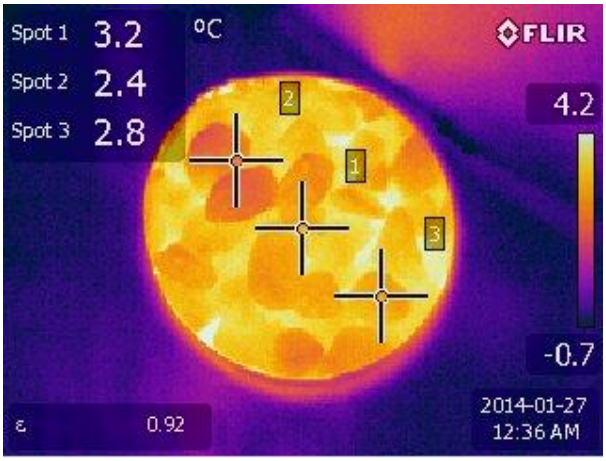
(a) (b)



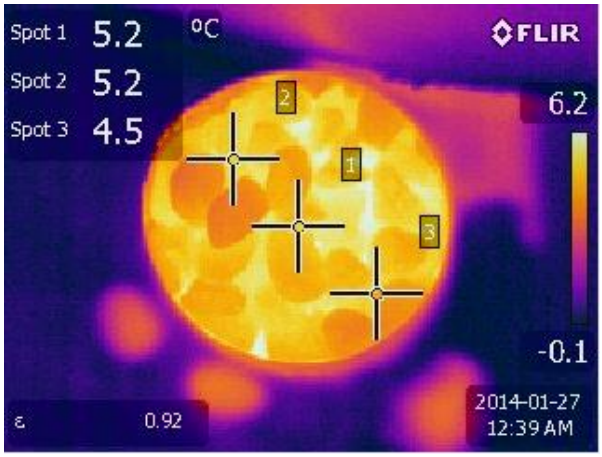
(c)



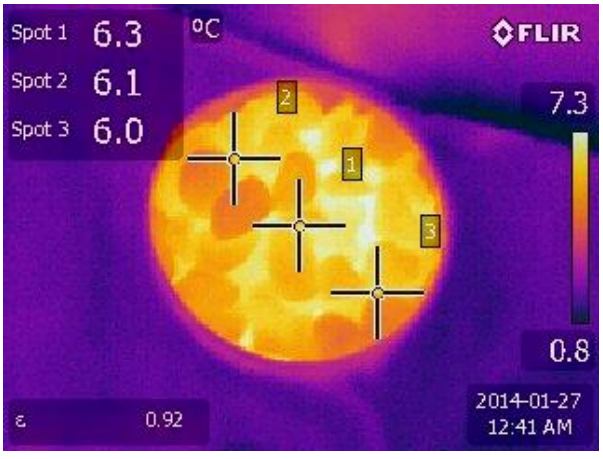
(d)



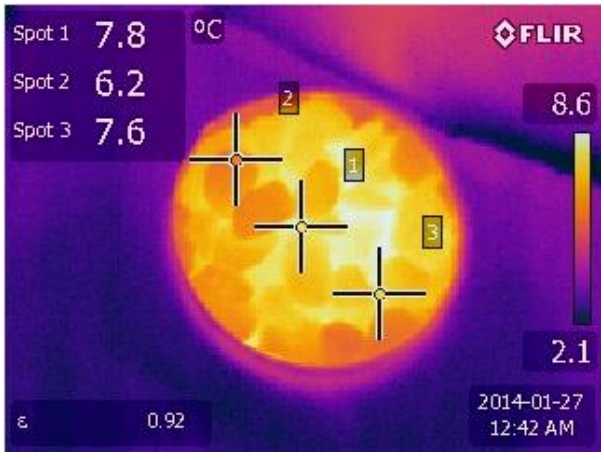
(e)



(f)



(g)



(h)

Figure 5-16 Thermal camera images at the outlet for aluminum pipe-experiment 2

The temperature distribution in time was analysed in every image taken by the thermal camera during the experiment. In the first image (Fig. 5-16a) the predominant colour was for 0°C; the temperature spots indicate 0.1°C at spot 1, -0.3°C at spot 2, and 0.2°C at spot 3. The lowest temperature was -2.2°C, and the highest temperature 1.9°C. The aluminum wall had the water temperature (0°C) and in general cannot be seen in the images. In the second image (Fig. 5-16b), the temperature spots increase the value in three minutes when the air from the hair dryer is passed through the porous media. Hence, the temperature spots were 0.2°C (central spot) at spot 1, 0.1°C at spot 2, and 0.2°C at spot 3. Also the temperature scale displayed an increasing value, the lowest temperature -2.1°C, and the highest temperature 2°C. After five minutes, the temperatures were increased to 0.6°C at all three spots. The highest temperature was at 2.5°C (Fig. 5-16c). The temperature continued to increase over the time reaching 2°C at spot 1, 1.9°C at spot 2, and 1.4°C at spot 3 as showed in Fig. 5-16d. The highest temperature was displayed at 3.2°C. In Fig. 5-16e the central temperature was increased with 1.2°C after 3 minutes. The other two temperatures of the spots also increased with 1.5°C, and with 1.4°C respectively. The temperature scale indicated the highest temperature at 4.2°C. Between Fig. 5-16e and Fig. 5-16f was a three minute interval. The temperatures were 5.2°C at spot 1 and spot 2, and 4.5°C at spot 3. The lowest temperature was at -0.1°C, and the highest at 6.2°C. The next image taken by the thermal camera (Fig. 5-16g) was after a further 3 minutes. This showed increased temperatures to 6.3°C at spot 1, to 6.1°C at spot 2, and to 6°C at spot 3. The highest temperature increased with 1.1°C. In the last image (Fig. 5-16h) the central temperature displayed was 7.8°C. The temperature at spot 2 was 6.2°C, and at spot 3 was 7.6°C. The highest temperature in this experiment was 8.6°C.

The central temperatures displayed by the temperature spot 1 were a fraction higher than the other two temperatures displayed by the temperature spot 2 and spot 3, because the temperature distributions are different. In the Fig. 5-17 are presented the evolution of temperature distribution in the experiment 2:

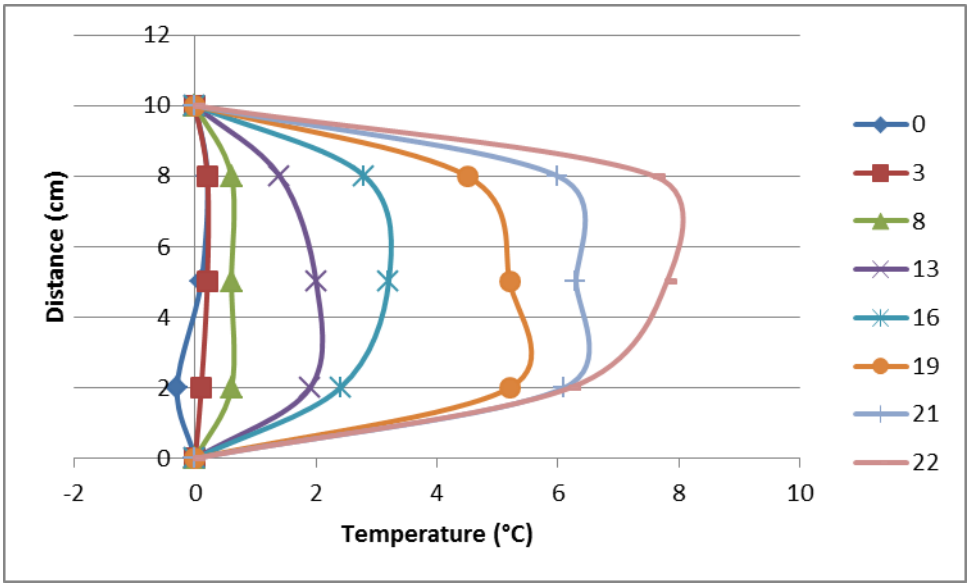


Figure 5-17 Temperature-profile graph for circular aluminum pipe –experiment 2

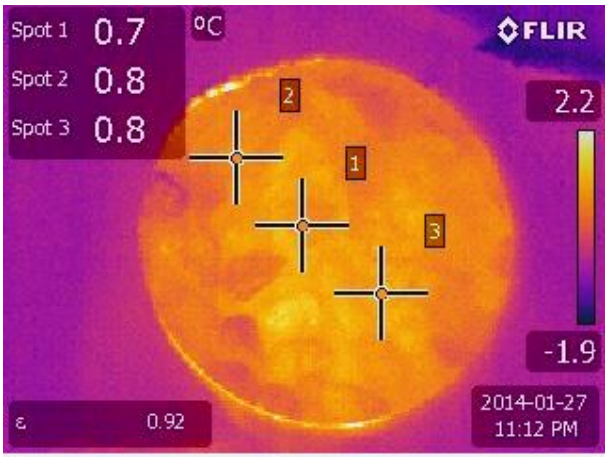
After 22 minutes, this experiment presented the same problem as in the first experiment: there was water leakage in the pipe, because of a hole between plastic and aluminum pipes. The occurrence of steam caused the temperature to drop.

5.3.3 Circular Aluminum Pipe - Experiment 3

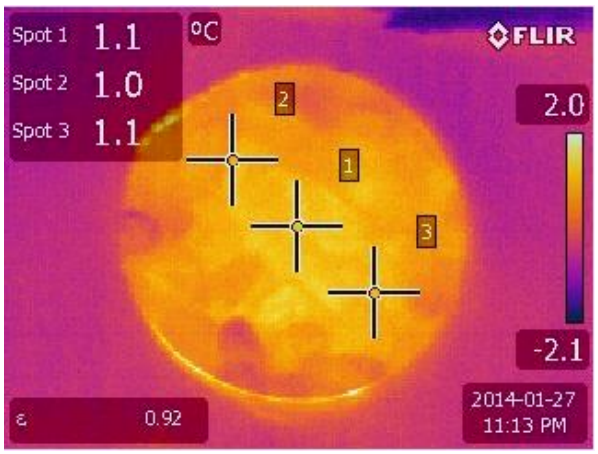
Another experiment was done, after fixing the problem. This was with the same goal, to increase the time after starting. In this experiment, more coating of silicon to seal the joint inside the aluminum pipe was used and duck tape was used outside. The assembly of filled aluminum pipe with gravel was immersed in water at 0°C. The hair dryer blew the air at 37°C, and the velocity

was approximated at 0.1m/s as before experiment. The emissivity was setup by the thermal camera at 0.92 for the gravel. The air flowed through the inlet with assumed uniform velocity, passed the porous media and exited the outlet. During the transit of the air flow through the gravel, a heat transfer occurred between the two media.

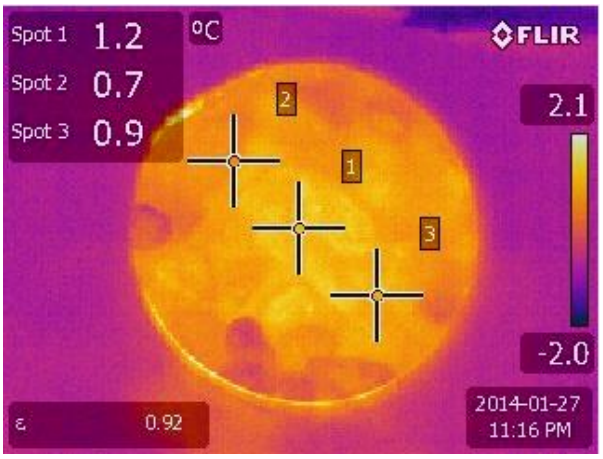
The same thermal camera was used to take the images at the outlet aluminum pipe. The distances between the temperature spots were at 3 cm. Every image presented the temperature spots, placed in the top left hand corner; the emissivity placed in the bottom left hand corner; the time when the pictures was been taken, placed in the bottom right hand corner. A legend of temperature was presented on the right hand side of the images as before, where the scale changes from each time. The images taken by the thermal camera during the experiment 3 were presented in Fig. 5-18:



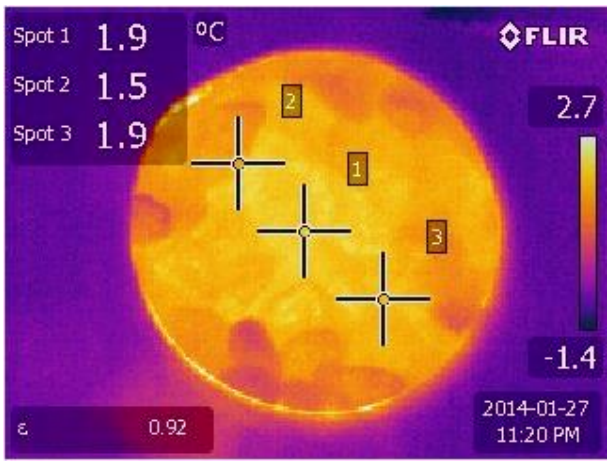
a)



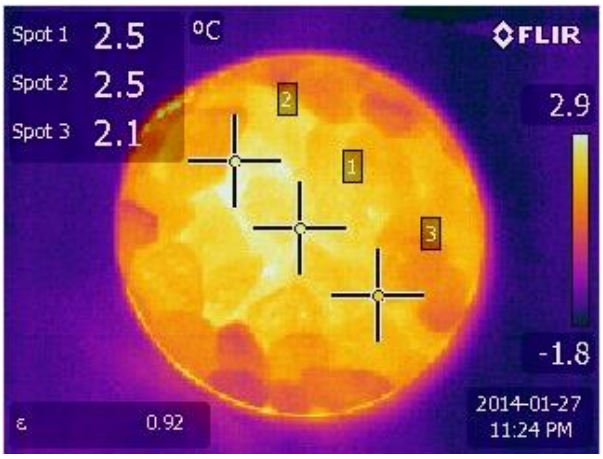
b)


Spot 1 1.2
Spot 2 0.7
Spot 3 0.9
°C
FLIR
2.1
-2.0
ε 0.92
2014-01-27
11:16 PM

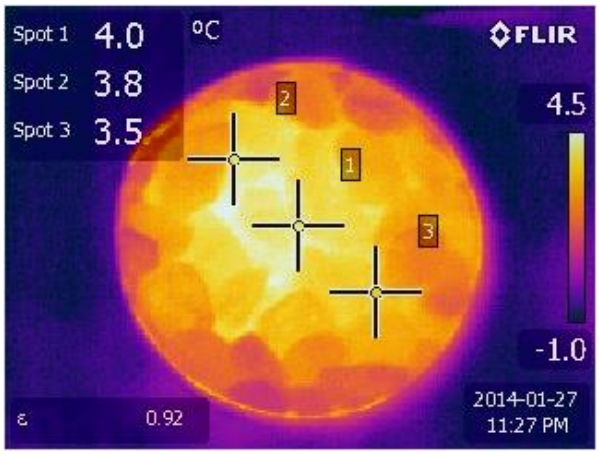
c)


Spot 1 1.9
Spot 2 1.5
Spot 3 1.9
°C
FLIR
2.7
-1.4
ε 0.92
2014-01-27
11:20 PM

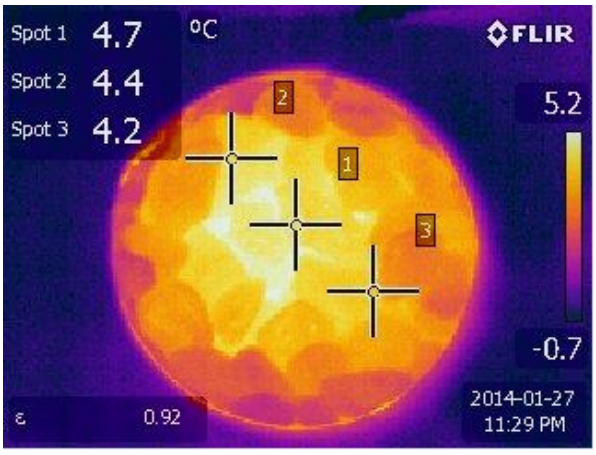
d)


Spot 1 2.5
Spot 2 2.5
Spot 3 2.1
°C
FLIR
2.9
-1.8
ε 0.92
2014-01-27
11:24 PM

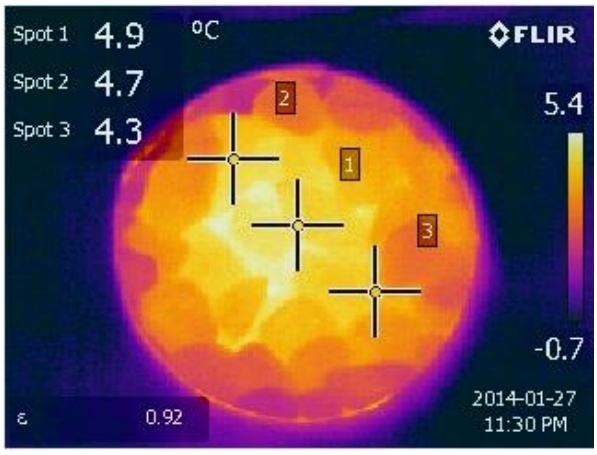
e)


Spot 1 4.0
Spot 2 3.8
Spot 3 3.5
°C
FLIR
4.5
-1.0
ε 0.92
2014-01-27
11:27 PM

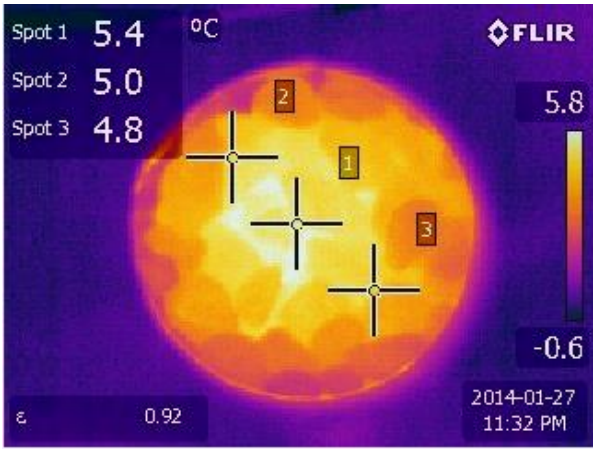
f)


Spot 1 4.7
Spot 2 4.4
Spot 3 4.2
°C
FLIR
5.2
-0.7
ε 0.92
2014-01-27
11:29 PM

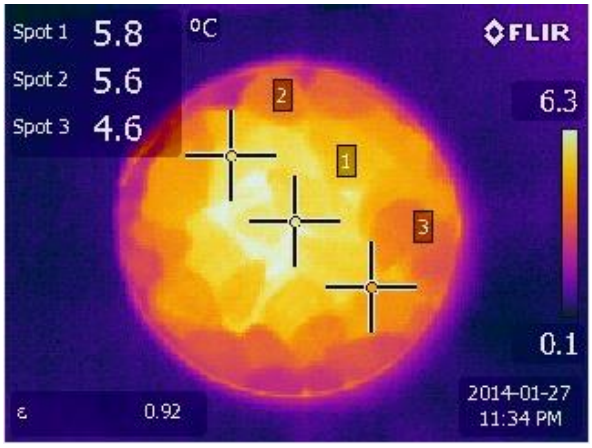
g)


Spot 1 4.9
Spot 2 4.7
Spot 3 4.3
°C
FLIR
5.4
-0.7
ε 0.92
2014-01-27
11:30 PM

h)



i)



j)

Figure 5-18 Thermal camera images at the outlet for aluminum pipe -experiment 3

The images were analysed and the temperature evolution examined in detail. In the first image (Fig. 5-18a), the temperatures displayed by the temperature spots were at the beginning 0.7°C for spot1, 0.8°C for spot 2 and spot 3. The highest temperature showed by the temperature scale was 2.2°C. After a further 1 minute the temperature increased with 0.4°C at spot1, 0.2°C at spot 2, and 0.3°C at spot 3 as showed in Fig. 5-18b. The images presented in Fig. 5-18c and Fig. 5-18d were taken at 4 minutes interval time. The corresponding difference of temperatures were 0.7°C at spot 1, 0.8°C at spot 2, and 1°C at spot 3. The temperature scale displayed the highest temperature at 2.7°C. In the Fig. 5-18e the temperature increasing to 2.5°C at spot 1 and spot 2, and 2.1°C at spot 3. All these occurred in 4 minutes. The highest temperature also increased at 2.9°C. After other more three minutes the temperature of the spots rose to 4°C at spot 1, 3.8°C at spot 2, and 3.5°C at spot 3 (Fig. 5-18f). In this image, the highest temperature displayed was 4.5°C, and the lowest temperature was -1°C. The next image taken by the thermal camera (Fig. 5-18g) was after two minutes than before and the temperatures increased with 0.7°C at spot 1, with 0.6°C at spot 2, and with 0.7°C at spot 3. The temperature scale displayed the highest temperature at 5.2°C, and the lowest at -0.7°C. The corresponding time between Fig. 5-18h and

Fig. 5-18i was two minutes. The temperatures rose in this time to 5.4°C at spot1, to 5°C at spot 2, and 4.8°C at spot 3. For highest temperature, the scale showed at 5.8°C in this image. The last image (Fig. 5-18j) was taken after two minutes than before, and the temperature were displayed to 5.8°C at spot 1, to 5.6°C at spot 2, and 4.6°C at spot 3. In this experiment was observed how the temperature distributions evolve with a similarity than central temperatures.

Fig. 5-19 presented the evolution of temperatures distribution across the diameter of the pipe in experiment 3. The temperatures evolution's profile was similar with the other two experiments.

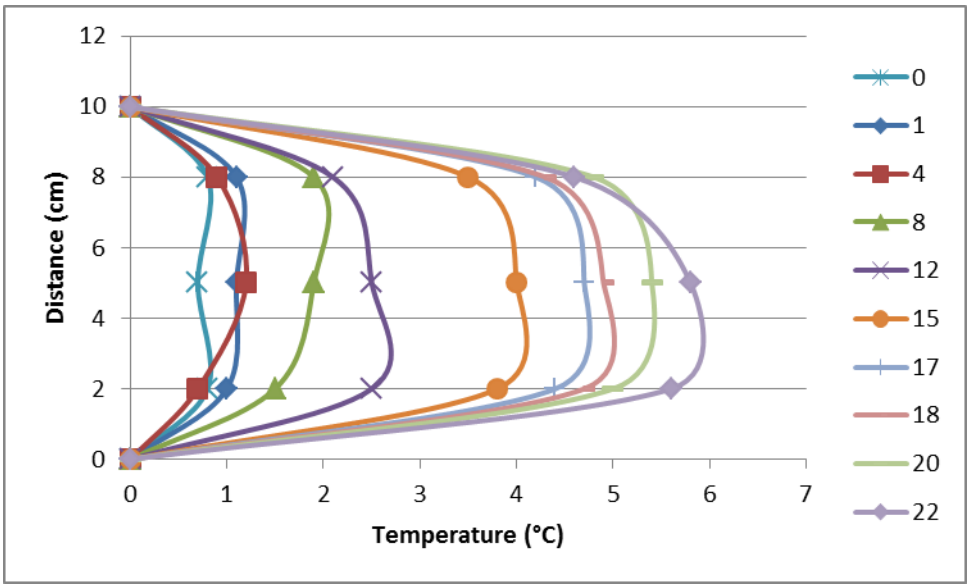


Figure 5-19 Temperature profile graph for circular aluminum pipe–experiment 3

The same issue of water leakage in the porous media influenced this experiment results. The duration of the experiment was 22 minutes and temperature varied from 0°C to 5.8°C.

A comparison between experiments results was made in Fig. 5-20, where the temperature-time evolution at spot 1 are presented:

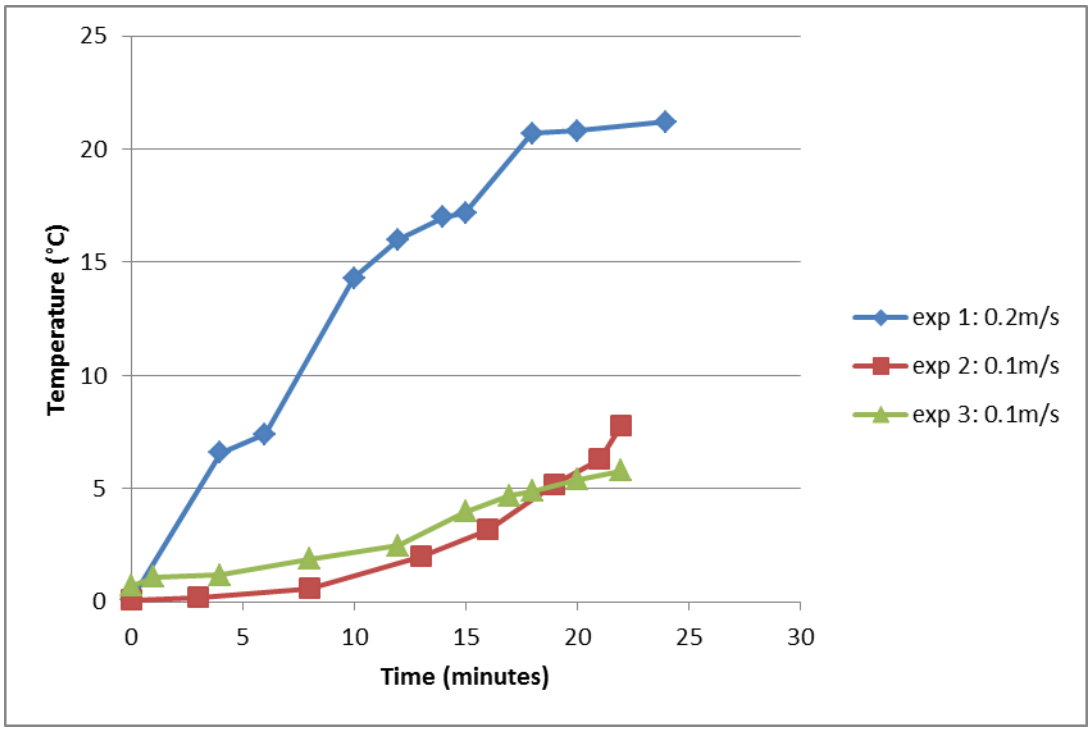


Figure 5-20 Temperature-time evolution for temperature spot 1, experiments 1, 2, 3

The temperature increased until close to the steady state. In the graph we show the temperatures evolution at central spot in time for the three experiments. For the first experiment, the air flow speed and temperature provided by the hair dryer, had higher values (0.2m/s), than the other two experiments (0.1m/s). This was because the speed of blown air by the hair dryer was changed.

5.4 Discussion of Physical Experimental Results

All those experiments are valuable for a comparison between the physical results and LBM simulation results. The experiments intended to show the effects of air flow and heat transfer through porous media. The first experiment consisted of a square box filled with gravel, as porous medium; the boundary condition was not the same with boundary condition used in

simulation. The main reason for the difference was the thermal conductivity of the cardboard box which influenced the heat temperature. Also was presented a warmer air along the edges of box; there was coupled pores at the contact of gravel with rectangular edges. The thermal camera had the capability to measure the surface temperature, not the inside of the porous media. The duration of the experiment, which was intended to produce heat transfer through porous media contained in the box, was for two hours. In comparison with the LBM simulation, the experience duration was significantly longer. Another experiment was needed to be designed in order to reproduce the isothermal condition of the simulation.

For this condition we conducted more experiments. In the first experiment we use a plastic pipe filled with gravel as porous media. In this experiment, the thermal camera measured the temperature at the exit of porous media. The evolution of temperature of the air flow through porous media showed a better heat transfer in time. The initial condition for the experiment was different from initial condition for the LBM model, because the walls presented thermally insulating properties. In a third set of experiments an aluminum pipe we used, which has very good thermal conductor properties, and keeping the walls at 0°C the isothermal condition for boundary was respected. This experiment had the same boundary condition as in LBM model. The evolution of temperature distribution for central spot was presented in the graphs (Section 5.3). In comparison the two models, physical and lattice, was different for the necessary time to travel the entire domain. The experiments where the isothermal boundary was used (circular aluminum pipe) were intended to provide the same boundary condition as in lattice simulation. The experiments did not reach the steady state because of water steam influence. The temperature measured during the experiments, until the steam altered the results, showed a resemblance to the distribution as in the LBM simulation. The different thermal expansion

coefficients for the material used to build the assembly (aluminum=$22.2x10^{-6}$m/mK; ABS=$30.4x10^{-6}$m/mK) caused leakage to occur over time. A construction of experiment assembly using the same materials or materials with the same expansion properties could solve the issues. In addition, using bounding glues with more elastic properties may be a solution.

6 LBM Simulations of Physical Experiments

The code developed in MATLAB by Adam Turcotte was based on material presented in Chapter 3 and Chapter 4, for the D2Q9 model, and evolved from a code written by Celik (2012). In this thesis for the simulations, various models were established: model 1, flow in a duct that tracked temperature, density, particle speed, and velocity vectors; model 2, with porous media, and horizontal velocity, vertical velocity, and velocity vectors tracked; model 3, with porous media, density, particle speed, and velocity vectors tracked. The simulation was designed to be similar to the physical experiment of the circular aluminum pipe. For the work presented in this thesis, I needed to learn and to familiarise myself with MATLAB software in order to run the code, and to work with the programmer to complete the task. The resulting MATLAB codes are presented in Appendix 1.

In these simulations, the boundary condition for temperature was considered isothermal, and all parameters used are given in lattice units. This means that a preliminary step in simulations is to convert physical quantities from the real world to the lattice world. This conversion is done based on the fact that several dimensionless numbers computed for fluid flow and heat transfer need to be identical in both 'worlds'.

6.1 Example of other Problem used for Verification

An example used for verification in this thesis is a master thesis by S. B. Celik (2012).

Fluid flow and heat transfer were studied numericaly for two-dimensional channels in Celik's work using the same computer code as subsequently used in this thesis for Model 1. Celik

compared velocity profiles and Nusselt numbers from the LBM with the Navier-Stokes based analytical and numerical results available in the literature. The LBM code had two parts: a velocity and a heat transfer calculation, with collision, streaming, boundary condition and macroscopic property calculation subsections as in this work. The Poiseulle flow domain in macroscale had no slip boundary conditions at the walls. A uniform velocity was specified at the inlet and the flow at the outlet was assumed to be fully developed. The non-dimensional temperatures at the inlet and at the walls are 1 and 0, as in Fig. 4-12. Figure 6-1, produced by Celik, with the code used in this thesis is compared to results from COMSOL (COMSOL, 2012) the widely accepted Finite Element Method, and verifies correct operation of the code used in Model 1.

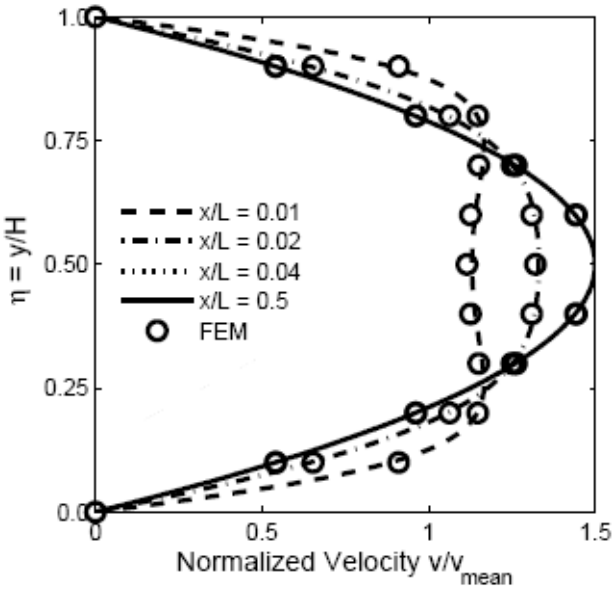


Figure 6-1 FEM and LBM comparison of developing velocity at macro-scale (Celik, 2012)

6.2 Linking the Lattice Boltzmann Method to Reality

The LBM simulation uses lattice units. Every quantity was defined in native physical units and thus needs to be converted to lattice units. The technique used to do this is to define equality of certain dimensionless numbers that define the problem in both the physical domain and the LBM domain. After simulation, lattice quantities need to be converted back to physical units. The following dimensionless numbers were used: Nusselt, Reynolds, Peclet, Prandtl, Rayleigh and Darcy all of which have been discussed in Chapter 2.

The dimensionless numbers were used to calculate the following coefficients in lattice units:

- characteristic length L
- kinematic viscosity ν
- thermal diffusivity α
- gravity and expansion coefficient $g\ \beta$
- permeability K

A tool using the Solver application in Microsoft Excel software was developed in order to calculate these coefficients such that the physical and lattice dimensionless number equalities are solved simultaneously. The solution was found by minimizing the sum of squared errors between the physical and lattice dimensionless numbers (or their logarithms).

In the following explanations, note that subscript P means the physical system and subscript LB means lattice system.

The calculation for kinematic viscosity is ν_{LB} given by Reynolds number:

$$\mathrm{Re}_P = \mathrm{Re}_{LB} \tag{6.1}$$

$$\frac{\mathbf{u}_P L_P}{\nu_P} = \frac{\mathbf{u}_{LB} L_{LB}}{\nu_{LB}}, \tag{6.2}$$

then the viscosity in lattice system is:

$$\nu_{LB} = \frac{\mathbf{u}_{LB} L_{LB} \nu_P}{\mathbf{u}_P L_P}. \tag{6.3}$$

The calculation for the thermal diffusivity is α_{LB} which appears in the Peclet number

$$Pe_P = Pe_{LB} \tag{6.4}$$

$$\frac{\mathbf{u}_P L_P}{\alpha_{\mathbf{P}}} = \frac{\mathbf{u}_{LB} L_{LB}}{\alpha_{\mathbf{LB}}}, \tag{6.5}$$

then thermal diffusivity in lattice system is:

$$\alpha_{LB} = \frac{\mathbf{u}_{LB} L_{LB} \alpha_P}{\mathbf{u}_P L_P}. \tag{6.6}$$

The calculation for the term $g_{LB}\beta_{LB}$ is given by the Rayleigh number:

$$Ra_P = Ra_{LB} \tag{6.7}$$

$$\frac{g_P \beta_P}{\nu_P \alpha_P} \Delta T_P L_P^3 = \frac{g_{LB} \beta_{LB}}{\nu_{LB} \alpha_{LB}} \Delta T_{LB} L_{LB}^3, \tag{6.8}$$

then this term in lattice system is:

$$g_{LB}\beta_{LB} = \frac{g_P \beta_P \nu_{LB} \alpha_{LB} \Delta T_P L_P^3}{\nu_P \alpha_P \Delta T_{LB} L_{LB}^3}. \tag{6.9}$$

The calculation for permeability is given by the Darcy number:

$$Da_P = Da_{LB} \tag{6.10}$$

$$\frac{K_P}{L_P^2} = \frac{K_{LB}}{L_{LB}^2}, \tag{6.11}$$

so that the permeability in lattice units is:

$$K_{LB} = \frac{K_P L_{LB}^2}{L_P^2}. \tag{6.12}$$

Table 6-1 summarised the thermal and physical properties of porous media and the corresponding calculated dimensionless numbers. The numbers in the yellow fields were inputs and the numbers in the blue fields were calculated.

In this example, the air velocity selected was 0.2 m/s and the other values were calculated using the tool referred in this section. Roberson and Crowe (1997) expressed the value for the air density, the kinematic viscosity, the thermal conductivity of the fluid, the gas constant, the air heat capacity, thermal diffusivity, and acceleration due to gravity. The value for the Boltzmann constant, the particle hard shell diameter, the temperature of particle dynamics, and the pressure of particle dynamics was used by Pace (2007). Bear (1972) selected the value for the permeability of the medium.

Table 6-1 Thermal and physical parameters of porous media

Description	Physical world	UM	Lattice world
Heat transfer coefficient (porous substrate to air)	0.01	W/m2K	
Characteristic length (diameter/span of duct)	0.10	m	55
Thermal conductivity (of air boundary layer)	0.026	W/mK	
Nusselt number	0.039		0.039
Boltzman constant	1.38×10^{-23}	J/K	
Particle hard shell diameter	6.2×10^{-10}	m2	
Temperature of particle dynamics	298.14	K	
Pressure of particle dynamics	101,325	Pa	
Characteristic length (diameter/span of duct)	0.10	m	
Mean free path (lambda)	2.38×10^{-8}	m	1.31×10^{-5}
Knudsen number	2.38×10^{-7}		2.38×10^{-7}
Air velocity	0.20	m/s	0.1
Kinematic viscosity	1.58×10^{-5}	m2/s	4.29×10^{-4}
Reynolds number	1276		1276
Thermal conductivity	0.0257	W/mK	
Gas constant	287.15	J/kgK	
Air density	1.18	kg/m3	
Air heat capacity	1005	J/kgK	
Thermal diffusivity	2.16×10^{-5}	m2/s	5.95×10^{-4}
Peclet number	926		924
Prandtl number	0.726		0.72
Gravity acceleration multiplied by thermal expansion	9.81x3.43	m/s2K	1.46×10^{-6}
Bulk air temp	298.14	K	
Diff between wall temp and bulk air temp	10	K	
Boundary surface temperature	308.14	K	
Film temperature	303.14	K	
Thermal expansion coefficient of air	2.30×10^{-3}	1/K	
Rayleigh number	9.55×10^{5}		9.55×10^{5}
Permeability of medium	1.00×10^{-9}	m2	3.03×10^{-4}
Darcy number	1.00×10^{-7}		1×10^{-7}

6.3 Simulation of Circular Pipe Experiment

For the simulation were used three models.

6.3.1 Model 1: without Porous Media, with Nusselt Number, Temperature, Density, Particle Speed, and Velocity Vectors

Fig. 6-2 to Fig. 6-6 presented results of the model's Nusselt number, temperature, density, particle speed, and velocity vectors. In this simulation the porosity was not introduced. The description of each set of results was:

- in the top left hand corner up the Nusselt number was plotted (calculated from the code) , which represented the ratio of convective to conductive heat transfer across the boundary, and along the duct.
- in the top right hand corner up the temperature of the air flow through the domain was plotted.
- in the middle left hand side the density was plotted.
- in the middle right hand side the particle speed was plotted.
- in the lower bottom the velocity vectors was ploted.

Each set of results were taken at different time steps: 100, 10000, 50000, 75000, and 175000. The maximum number of time steps was 200000. For the temperature plot, the top and the bottom walls were considered cold along the domain. In the simulation code was notated with 0, which was represented cold temperature, and the hot temperature was notated with 1.

In Fig. 6-2 at the beginning, the temperature was represented by the colour at the entrance of domain. In the rest of domain was the black colour, which means there is nothing, because there

was no flow in this time of simulation. At the density contour the presence of density in small part on the left hand side of domain was showed. The particle speed was started to go through the domain and the value also to increase, on the same small part of domain. At the walls (top and bottom) and the rest of domain, the value of particle speed was zero, which means no air flow there (blue colour). The velocity vectors were not influenced by the other forces and the senses was straight ahead, through the domain. At time step 100 the velocity was presented at the short part in the left hand side of domain.

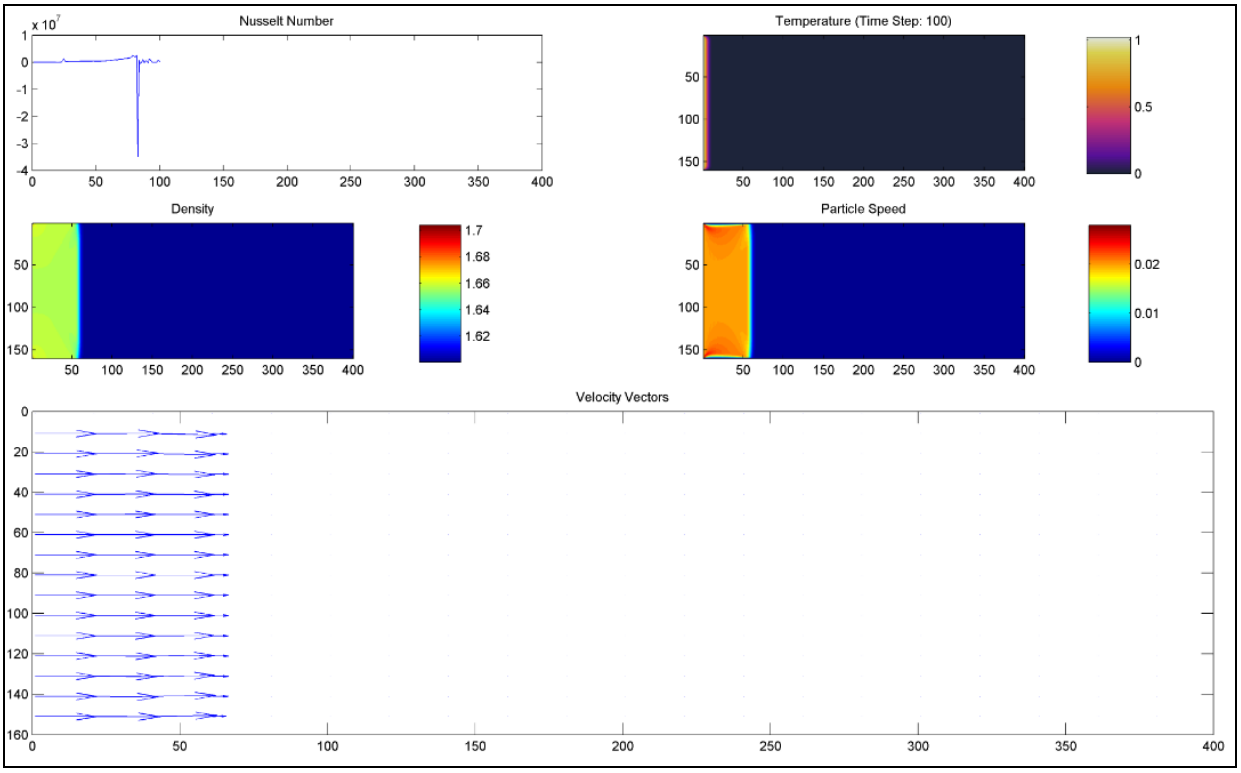


Figure 6-2 LBM model 1 plots at time step 100

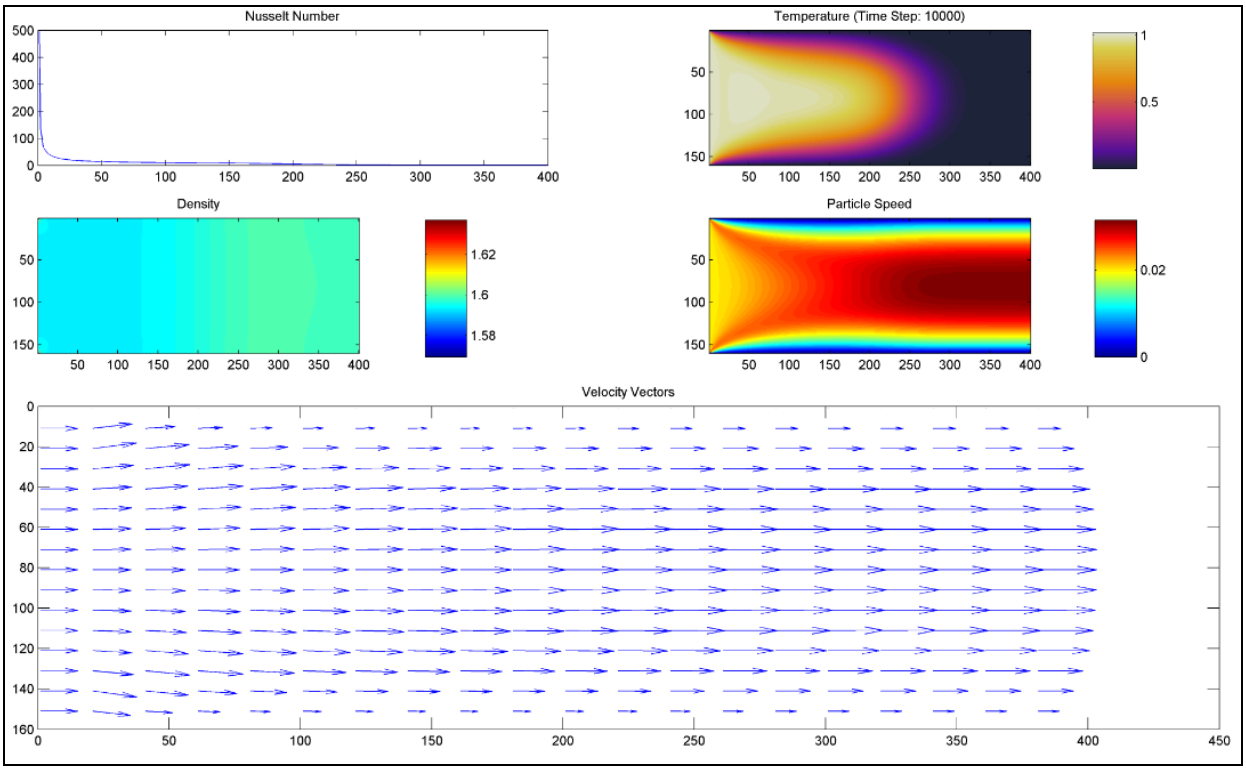


Figure 6-3 LMB model 1 plots at time step 10000

The Fig. 6-3 was taken at timesteps 10000. In Fig. 6-3 to Fig. 6-6 Nusselt number (over 400) was dropped at the entrance in domain (left hand side) and then, was fluctuating close the value one across the right hand side of domain. This meant the heat transfer mechanism was more convective than conductive for short time. The temperature plot presented shows heat transfer occured over the half way of the domain. The highest value of temperature was given by light yellow colour, placed in the middle, and then the colour was changing becoming orange and darker through black, which means there was reduced temperature (near the walls was kept the black colour, thus the cold temperature with value zero). The density contour showed that the density was presented in the entire domain. The values for the density were presented by the legend of colour. The particle speeds were presented in the entire domain at the difference of value. The value for particle speed at the walls was zero, because at the walls the velocity had

value zero. In this figure, the velocity vectors were passed entire domain. The velocity profile was parabolic, where the maximum value for velocity was in the middle; and near the walls, value was zero.

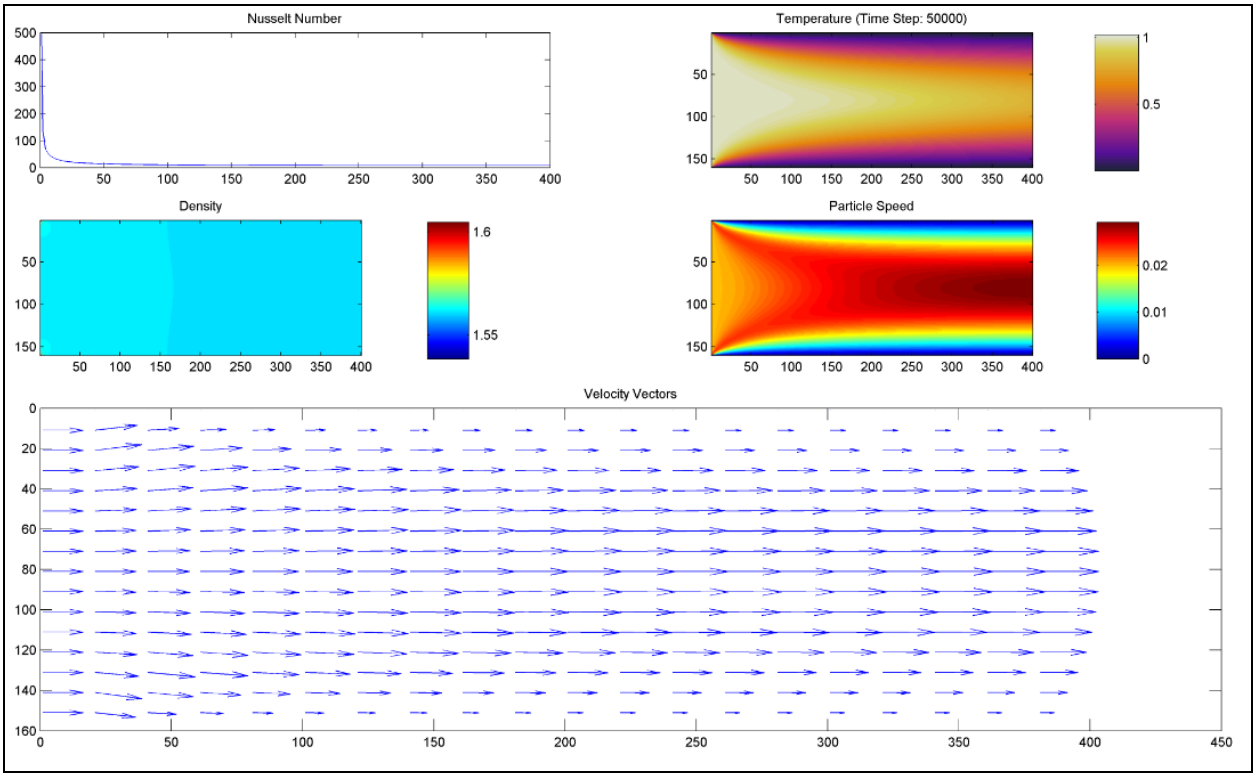


Figure 6-4 LBM model 1 plots at time step 50000

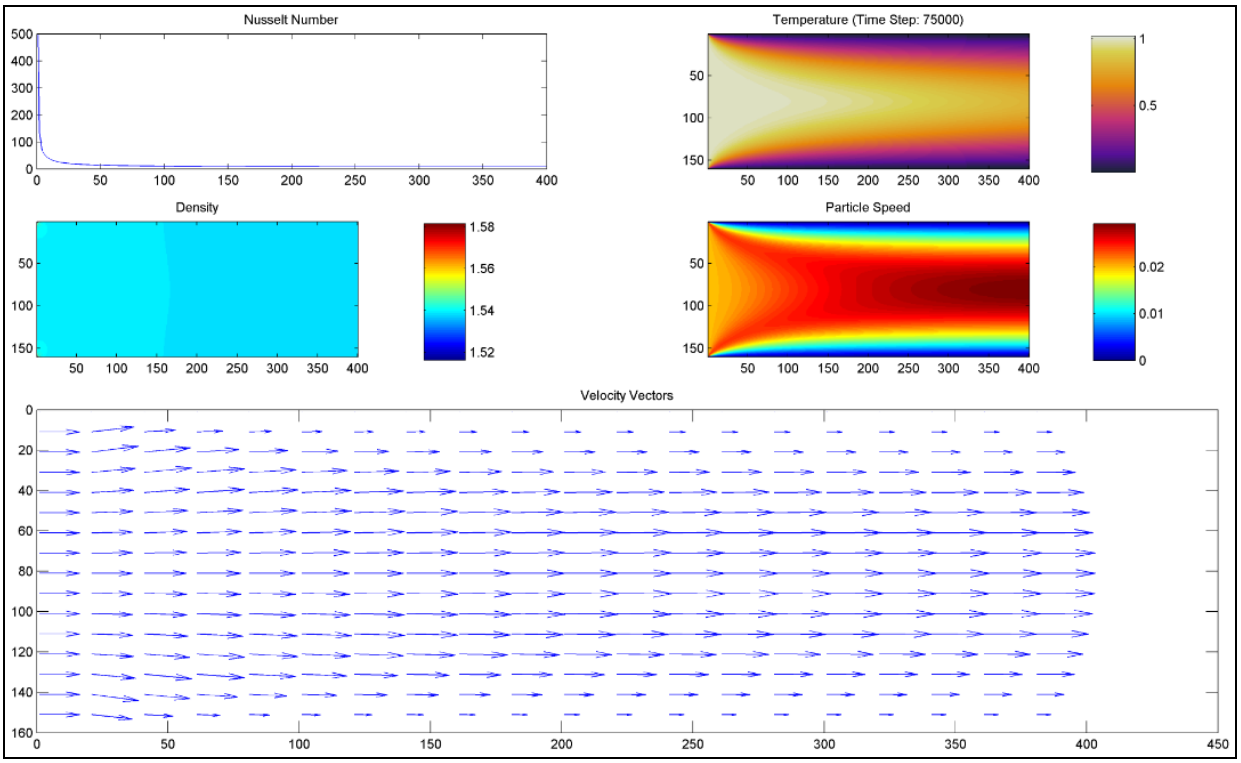


Figure 6-5 LBM model 1 plots at time step 75000

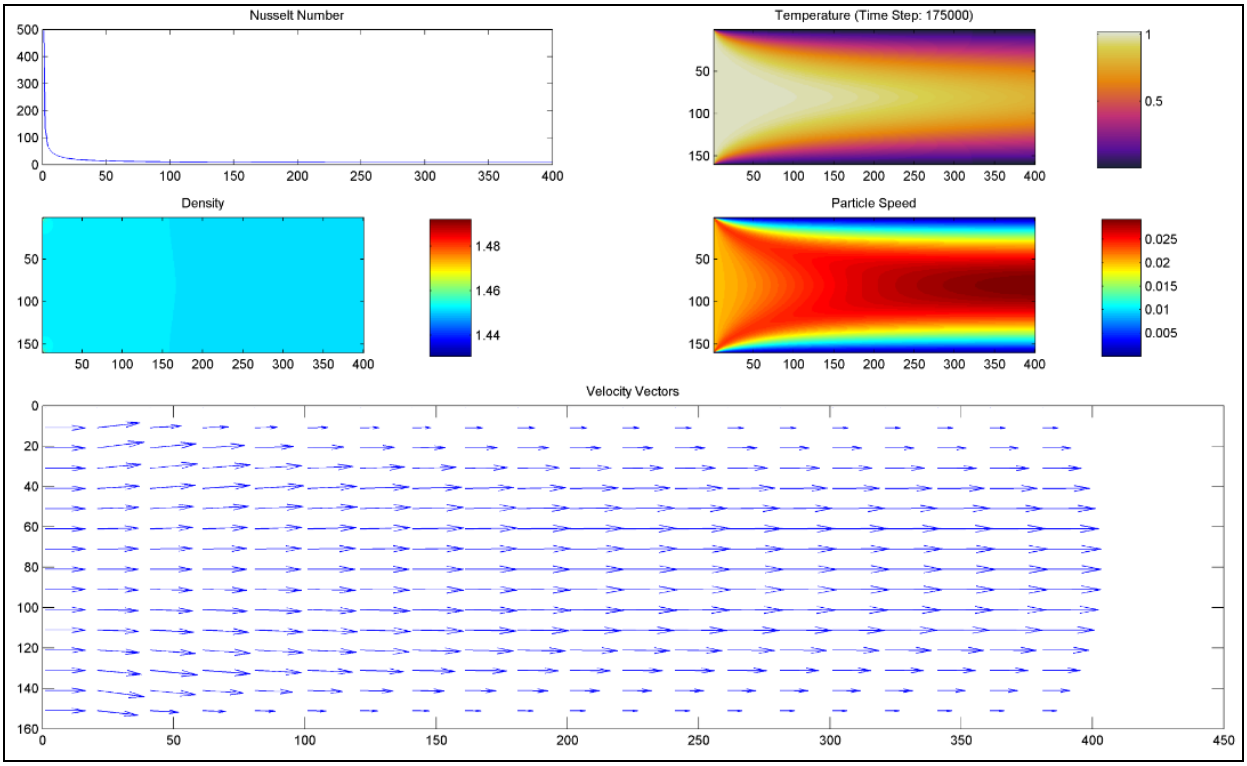


Figure 6-6 LBM model 1 plots at time step 175000

The Fig. 6-4 was taken at timstep 50000; the Fig. 6-5 was taken at timestep 75000; the Fig. 6-6 was taken at timestep 175000. All of them were presented the same results. The temperature plot showed the steady state, with the value of temperature near the walls still zero (black colour). The density was presented in entire domain, with higher value for the first half of domain than the second half of domain. The particle speed was also presented in entire domain with concentrated value at the centre and near the walls with zero. At wall, the particle speed had value of zero, all the time, represented by black colour. The velocity profile was comparable with velocity profile in a pipe, where the maximum value was in the middle, and decreased at the walls. At the walls, the value was zero.

6.3.2 Model 2: with Porous Media, Nusselt Number, Temperature, Horizontal and Vertical Velocity, and Velocity Vectors

Fig. 6-7 to Fig. 6-10 presented results of the model's Nusselt number, temperature and velocities evolution in time, along the duct.

The description of each set of results is similar as in Model 1; for the middle left hand side a contour plot of the horizontal velocity was plotted instead the density and in the middle right hand side a contour plot the vertical velocity was plotted instead the particle speed.

Each set of results applies for increasing numbers of time steps of the lattice Boltzmann model: 20300; 65450; 92,000; and 160000, when the model was close to the steady state. The maximum number of time steps is 160000. The velocity was as input 0.2 m/s. The porosity used in this run was 0.486, and the permeability was 10^{-8} m^2. The top and bottom walls were isothermal along the duct; in the boundary condition, the temperature was considered the same, cold temperature.

The first image, Fig.6-7, was taken at time step 20300.

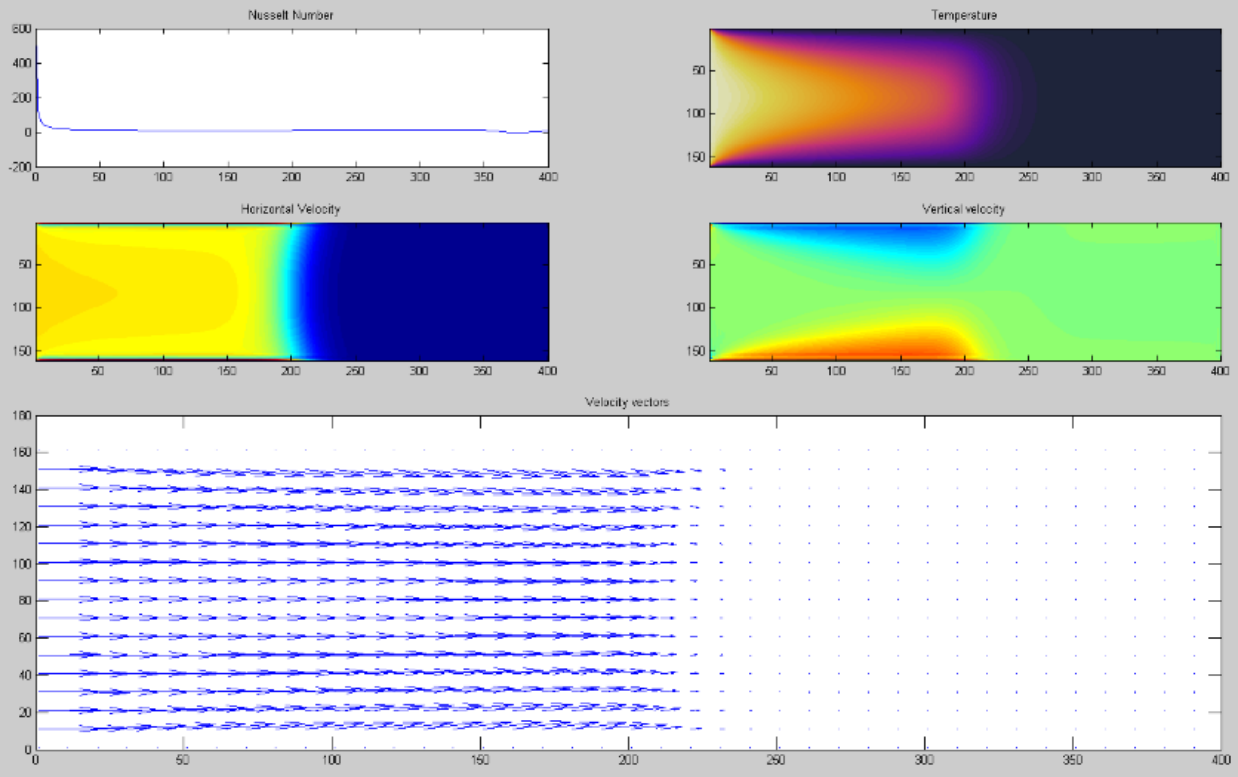


Figure 6-7 LBM model 2 plots at time step 20300, eps=0.486, K=1e-8

In Fig. 6-7 to Fig. 6-10 Nusselt number dropped from a high value (over 300) at the right hand side where there was a constant velocity and temperature, and then boundary condition applied fluctuated close to one as one traverses to the right hand side. That means the heat transfer mechanism was more convective than conductive on the extreme left hand side. The Nusselt number has had initially a high value (over 300), which it is interpreted that the fluid flow is more convective than conductive. A value of Nusselt number close to one is namely convection and conduction of similar magnitude and it is characteristic of laminar flow. The hot temperature in this figure was over the half way of domain. It was represented as light yellow colour in the middle, then become orange and changed darker through black. In the right half of domain, the dark colour showed that there was not heat transfer between the air flow and porous media,

principally because there is no flow on the right hand side of the lattice at this particular time of the simulation.

Non-zero horizontal velocity was present in more than half domain (yellow colour). The dark blue colour represents zero velocity. At the location of the steepest velocity gradient, the spatial trend appears curved, and this is due to the buoyancy of the air (the force given of acceleration due to gravity and thermal expansion). The vertical velocity presented in this figure, was the same within the right hand side of the domain. On the left hand side the plots have different colours at the top and bottom because the velocity senses were opposite. The magnitude and direction of velocity are presented in the lower vector plot. On the right hand side the length of velocities are so short that they seems to disappear.

Fig. 6-8 was taken at time step 65450.

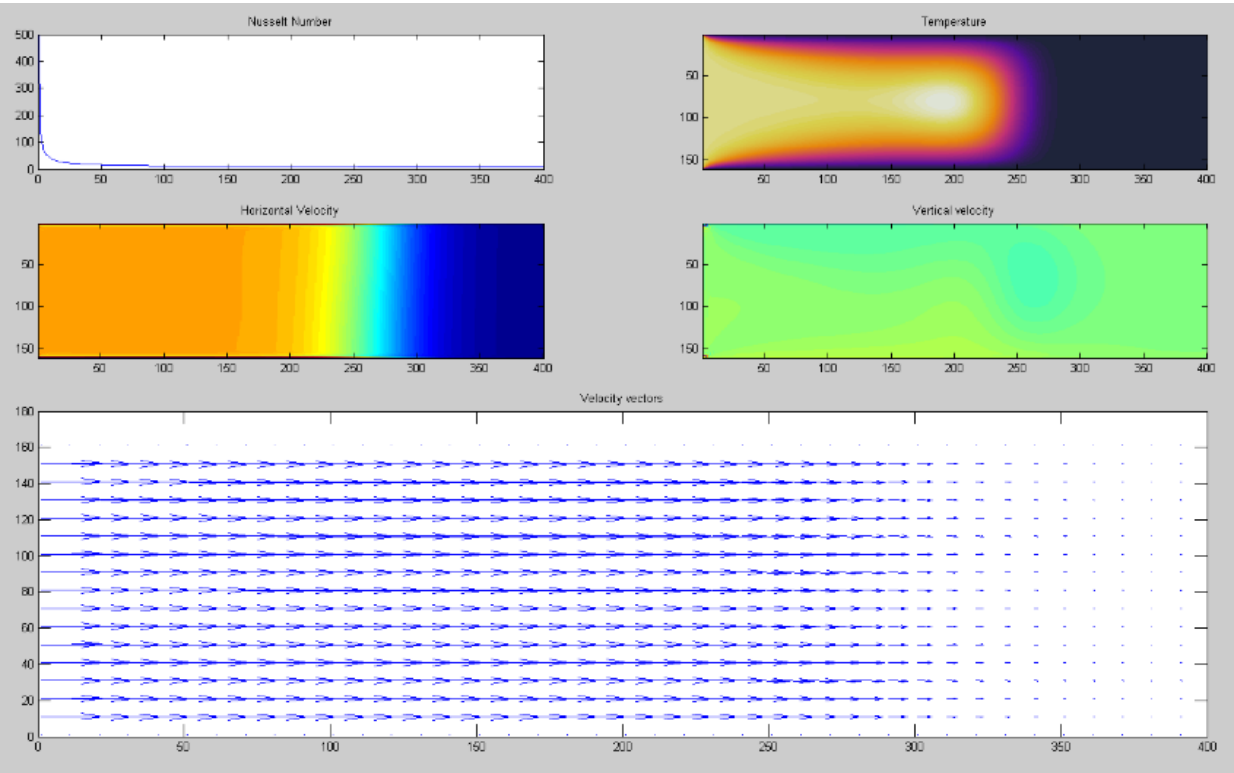


Figure 6-8 LBM model 2 plots at timestep 65450, eps=0.486, K=1e-8

The hot temperature plot in this figure was over two thirds of the domain, and was showed as light yellow colour in the middle and changed the edge through orange to blue and then black. The heat transfer was not in the entire domain, because there was no flow. The horizontal velocity was progressing at the right hand side, reaching two thirds of the domain. The intensity of colour was different which means that the velocities are different. At the location of the steepest velocity gradient, the spatial trend appears not symmetrical, because there were other forces (buoyancy of the air). In this image, the blue colour displayed that was not velocity in this time of the simulation. The vertical velocity was observed as not uniformly developed, as shown by the colour distribution in the two thirds of domain, indicated some circulation. In the rest of the domain, the colour was presented uniform, indicating virtually constant vertical velocity components throughout the lattice.

The velocity vectors were plotted in the lowest diagram in the bottom and confirm the left hand side to right hand side motion expected. The length of vectors becomes shorter in last third of the domain from left to right, and then they seem to disappear.

The Fig. 6-9 was taken at timestep 92000, where a clear progression of temperature and velocity distribution on the lattice is observable. The Nusselt number profile was similar to that at 65500 time steps, suggesting that heat transfer modes remain stationary through the evolution. The 'Warm front' in the temperature distribution passed completely from left to right across the lattice. The temperature distribution shows consistency with isothermal boundaries. The horizontal velocity becomes non-zero throughout the domain. Some circulation in the vertical velocity remains apparent.

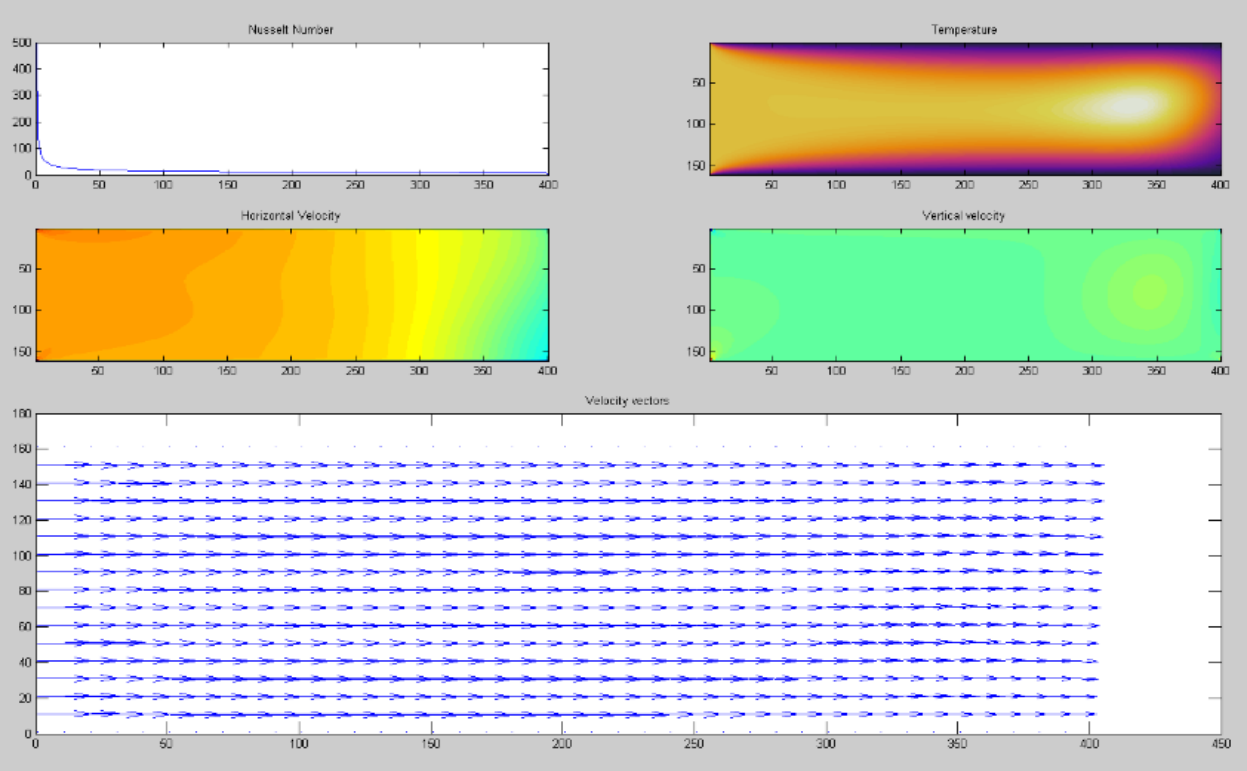


Figure 6-9 LBM model 2 plots at timestep 92000, eps=0.486, K=1e-8

The last figure (Fig. 6-10) was presented the LBM model very close to the equilibrium state, determined through comparison of temperature and velocity from 92000 time steps; not much changed. Isothermal upper and lower boundaries clearly remain the same. The horizontal velocity and vertical field were practicaly identical to those of 92000 time steps, after 60000 more time steps and they are not symmetrical because the effect of buoyancy.

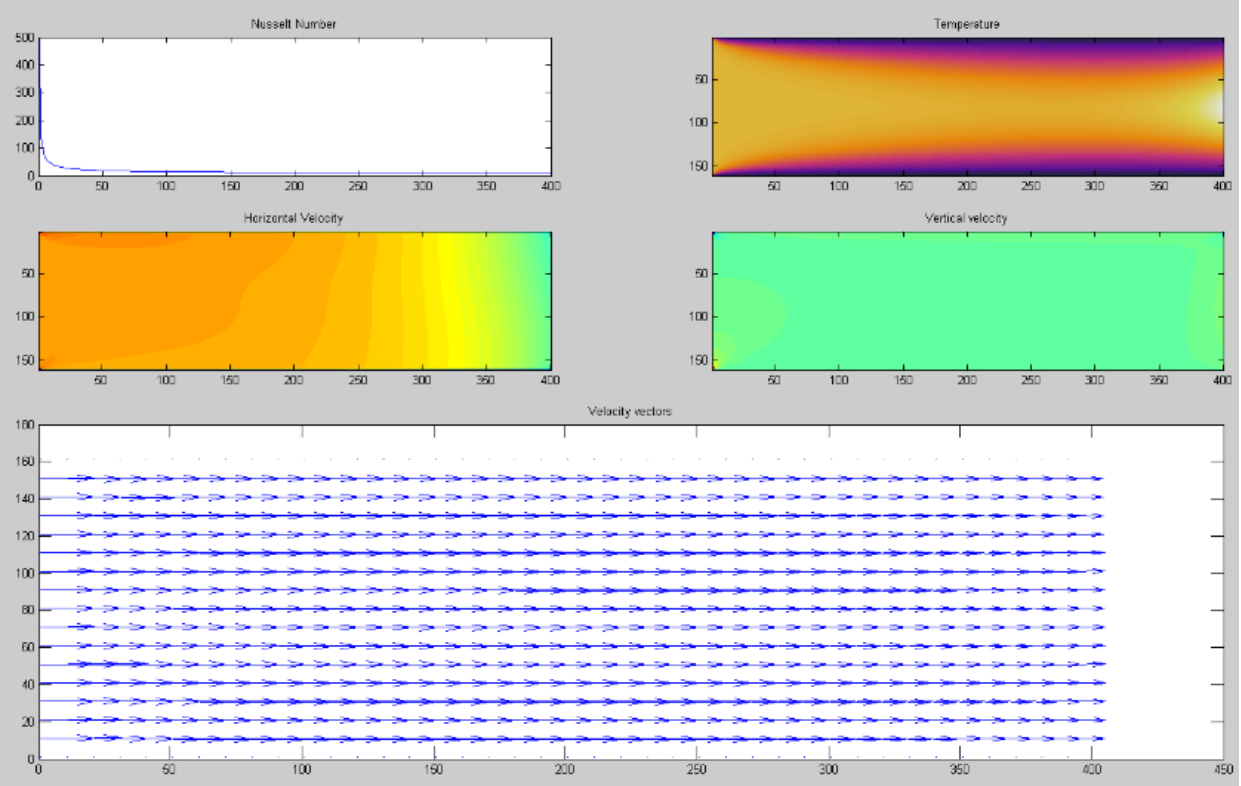


Figure 6-10 LBM model 2 plots at timestep 160000, eps=0.486, K=1e-8

A comparison of physical model with the LBM model was done.

For a rectangular box with varying boundary condition the thermal camera only had the capability to measure the surface temperature. The camera cannot measure the temperature below the surface. The temperature was influenced also, by the thermal conductivity of the box. The temperature was presented in both, the experiment and the simulation, but the results were different because the heat transfer was restricted by the boundary condition. In the LBM model, the top and bottom walls were considered isothermal. For this reason, in the simulation the walls had cold temperatures that they were presented in the plots as dark colour. In physical model, the walls were built with cardboard, which allows the heat transfer between the porous media and environment. The temperatures measured with thermal camera were actually the middle box

surface, not in the porous media. The duration of the experiment which was intended to produce the transfer of heat between air flow and in the entire porous media contained in the box, was two hours. That is for a long time in comparison with the LBM simulation. There was the difference between the temperatures used; in physical model was 16° C and for the LBM simulation was 10° C.

In the physical experiment, the air flow velocity was measured to be approximately 0.2 m/s. The value of velocity was assumed to be constant through the box and then the time required pass through was calculated as 1.6 s. This value of 0.2 m/s was used as input for velocity in the LBM model. The velocity in lattice system was calculated as 0.01 lu/ts.

The conversion of the time from the LBM model in physical time was obtained by using the velocity plots. Examination of the sequence (Fig. 6-7 to Fig. 6-10) shows that a pulse propagates across the lattice initially. From Fig. 6-7 where the velocities vectors are non-zero halfway across the lattice the corresponding distance traveled is:

$$d = \frac{0.32}{2} = 0.16 \text{ m} \tag{6.13}$$

In the simulation the corresponding physical time is:

$$t = \frac{0.16}{0.2} = 0.8 \text{ s} \tag{6.14}$$

The time for pulse to travel half way across the porous media is:

$$t_c = \frac{d}{c} = \frac{0.16}{340} = 0.47 \times 10^{-3} \approx 0.5 \text{ ms} \tag{6.15}$$

Where $c = 340$ m/s, which represents the speed of sound in air at 18° C (Roberson and Crowe, 1997).

The time per one time step is:

$$1_{ts} = \frac{0.5}{20300} = 2.5 \times 10^{-5}\,\mathrm{s} \tag{6.16}$$

For the Fig. 6-8 the physical time is:

$$t = \frac{65000}{20300} \times 0.5 = 1.61\,\mathrm{ms} \tag{6.17}$$

For the Fig. 6-9 the physical time is:

$$t = \frac{92000}{20300} \times 0.5 = 2.26\,\mathrm{ms} \tag{6.18}$$

For the Fig. 6-10 the physical time is:

$$t = \frac{160000}{20300} \times 0.5 = 3.94\,\mathrm{ms} \tag{6.19}$$

LBM appears to simulate a very short period of time, requiring a very large computational effort.

6.3.3 Model 3: with Porous Media, Nusselt Number, Temperature, Density, Particle Speed, and Velocity Vectors

From Fig. 6-11 to Fig. 6-15 presented results of the model's Nusselt number, temperature and velocities evolution in time, along the duct. This model was similar with model 2; however the porosity used was 0.440, and the permeability was 10^{-9} m^2 (one order magnitude smaller than in model 2). In adition the horizontal velocity plot was replaced by the density plot, and the vertical velocity was replaced by the particle speed plot. In the figures was introduced also, legend of

colours for the temperature, for the density, and for the particle speed. The velocities vectors were plotted as before.

The description of each set of results was:

- in the top left hand corner up the Nusselt number was plotted, which represented the ratio of convective to conductive heat transfer along the duct.
- in the top right hand corner up the temperature of the air through the porous media was plotted.
- in the middle left hand side a density was plotted.
- in the middle right hand side a particle speed was plotted.
- In the lower bottom the velocities vectors were plotted.

Each set of result applies for increasing numbers of timestepts of the lattice Boltzmann Model: 2000, 50000, 75000, 100000, and 125000, when the model was close to the steady state. The maximum number of time steps was 200000. The velocity was as input 0.2 m/s. The top and the bottom walls were isothermal along the duct, the temperature was considered as cold temperature and value was 0 (in the legend of colours was represented as dark purple). The hot temperature was considered with value 1, and the colour was represented as light yellow.

The Fig. 6-11 was taken at timestep 2000. The Nusselt number was vibrating close to one, thus there the heat transfer modes remain stationary through the evolution. In the temperature image was observed that was starting to pass the domain. The colours at the entrance of domain showed that the heat transfer was occurred, and then was the dark colour, which means there was not heat transfer, because was no flow in that time. The variation visible at the entrance for the density showed that the density started to be present in the domain. The particle speed plot

showed the value of the particle speed, the value was correlated with the legend of colour. For the time step 2000, the particle speed was in short part in the left hand side of the domain. The velocity vectors were presented in the bottom of figure. The direction of the vectors was horizontal inside the domain at the beginning of simulation.

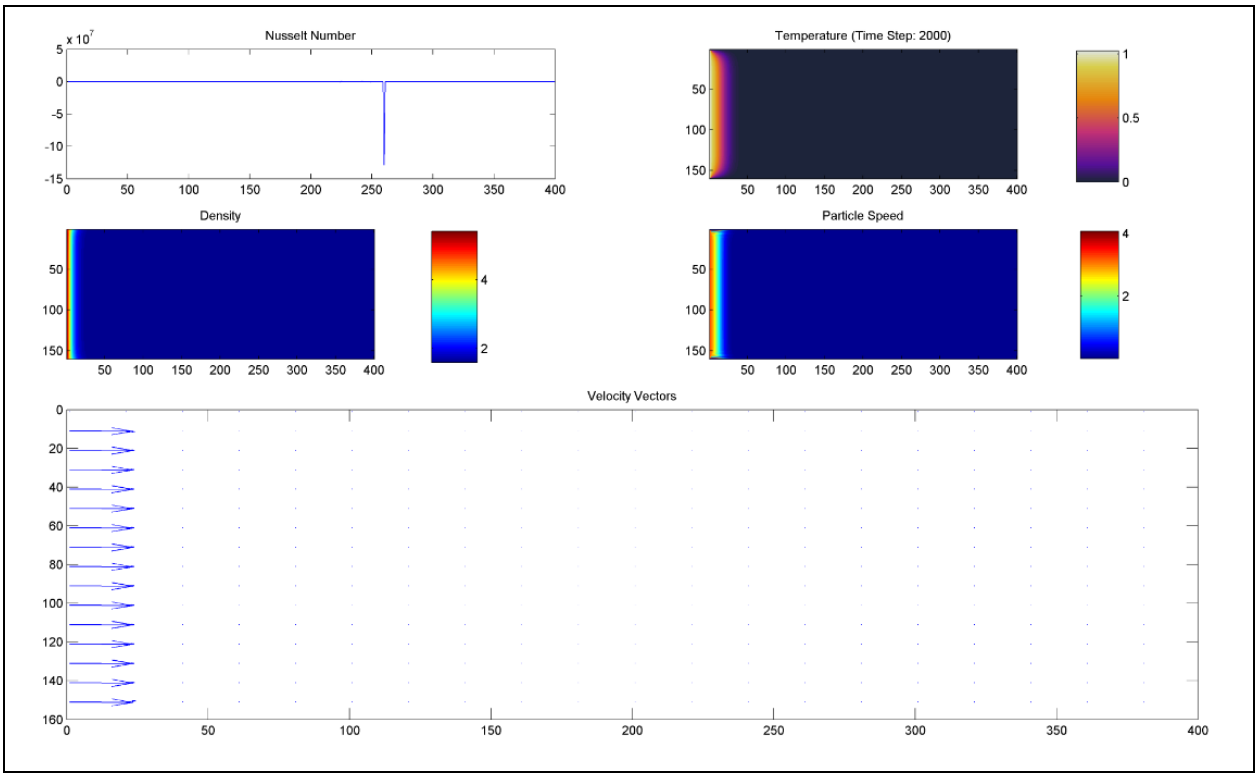


Figure 6-11 LBM model 3 plots at timestep 2000,eps=0.44, K=1e-9

The fig. 6-12 was taken at time step 5000. The Nusselt number still closes the value one after at beginning was dropping from high value (over 400). The temperature was visible progressing almost to reach half of the domain. The light yellow was the hottest temperature, and the colour changed when was colder. The top and the bottom walls still isothermal along the duct. The dark colour represented the cold temperature; the black showed was not heat transfer. The value of density increased at value 3 to the legend of colour. The particle speed was presented almost half of the domain. At the location of the steepest particle speed gradient, had lower value and the

spatial trend appears curved because it was influenced by the buoyancy. The blue colour was showed that, there were not particles speeds presented. The light blue colour indicated particle speed existed, and the yellow colour represented the maximum particle speed. The velocities vectors were presented in the half way of domain. The other forces as buoyancy influenced the senses of vectors. The lengths of velocities were becoming shorter until they were vanished.

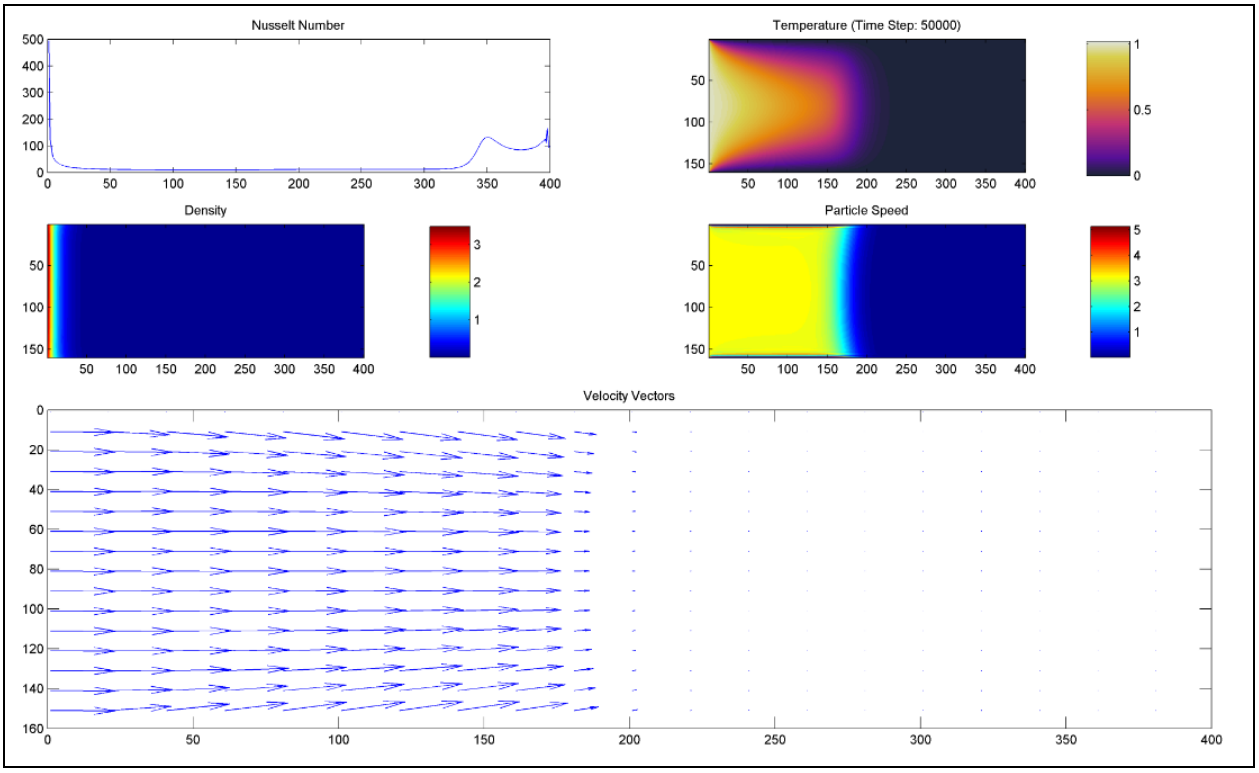


Figure 6-12 LBM model 3 plots at timestep 50000 eps=0.44, K=1e-9

Fig. 6-13 was taken at timestep 75000. In this figure were presented more progress in heat transfer, density, particle speed, and velocity vectors then before, very close the two thirds way of domain. The magnitude and direction of velocity are presented in the lower vector plot. The velocity senses were opposite because the buoyancy influence. The length of velocities is so short that they seem to disappear.

Fig. 6-14 was taken at timestep 100000. In this figure the temperature was close the exit of domain. The density was presented at entrance of the domain. The particle speed was over the two thirds of domain, and at walls was the value of zero, because at the walls velocity was zero. The velocity vectors were moved from the left hand side through right hand side close the exit of domain.

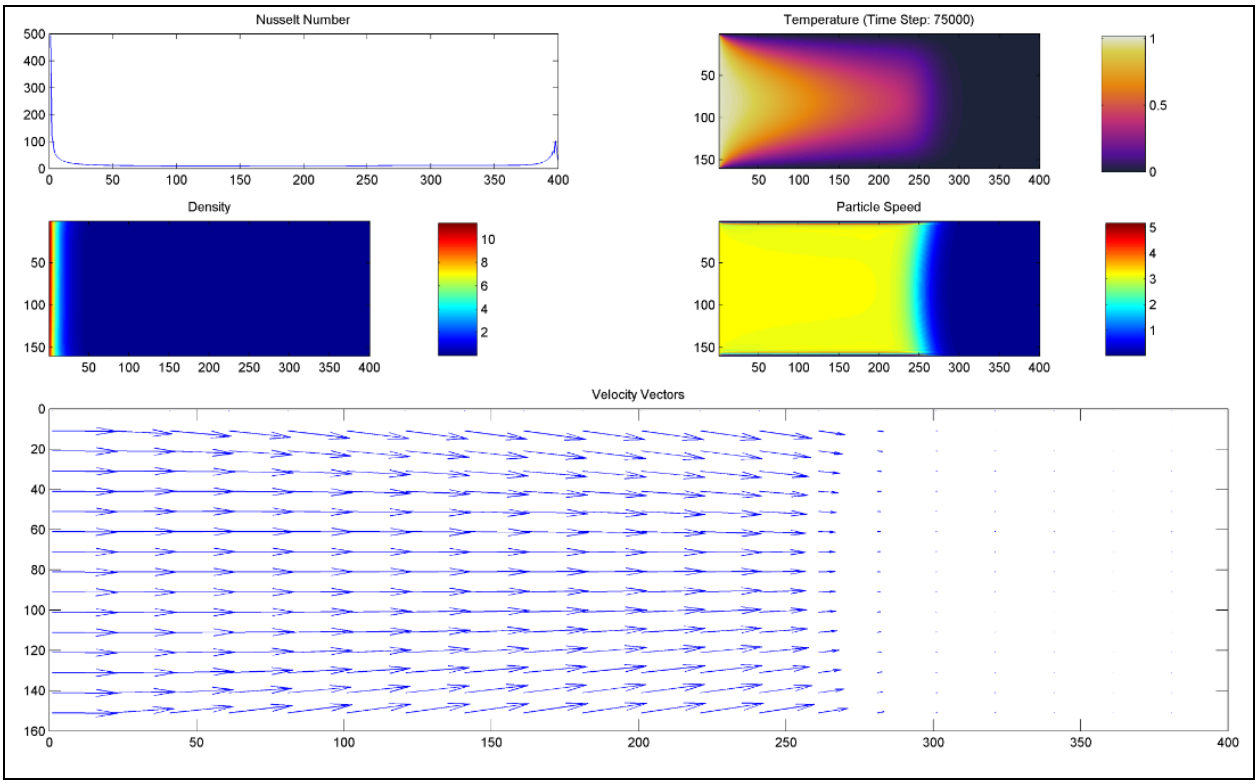


Figure 6-13 LBM model 3 plots at timestep 750000 eps=0.44, K=1e-9

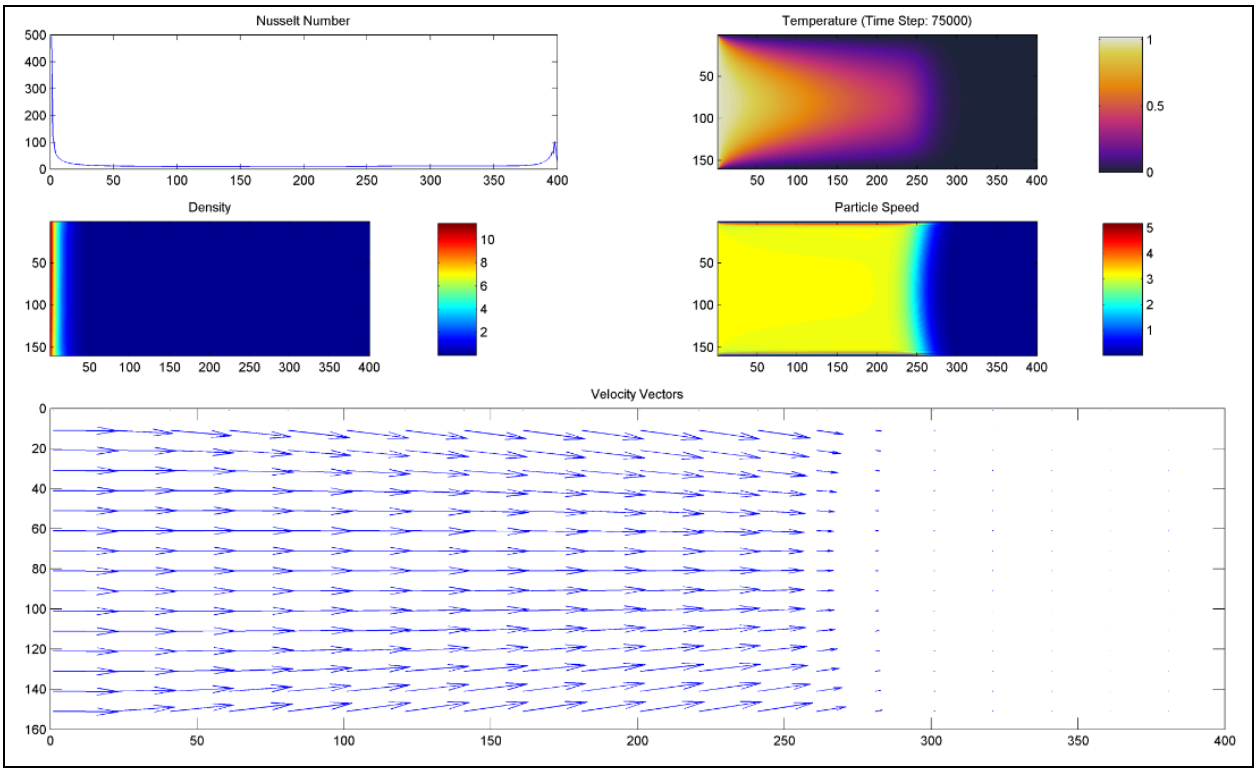


Figure 6-14 LBM model 3 plots at timestep 100000 eps=0.44, K=1e-9

Fig. 6-15 was taken at time step 125000. The Nusselt number was dropping at the beginning and then was stabilized close the value one, which means the heat transfer was stationary. The temperature was presented in the entire domain; the heat transfer was not uniformly along the domain. The color scale showed the density was increased at the beginning. The particle speed was almost the same along the domain, and top and bottom still with value zero. The velocity vectors image presented the velocity were through out of the domain. The direction was influenced by the buoyancy and there was not velocity reversal.

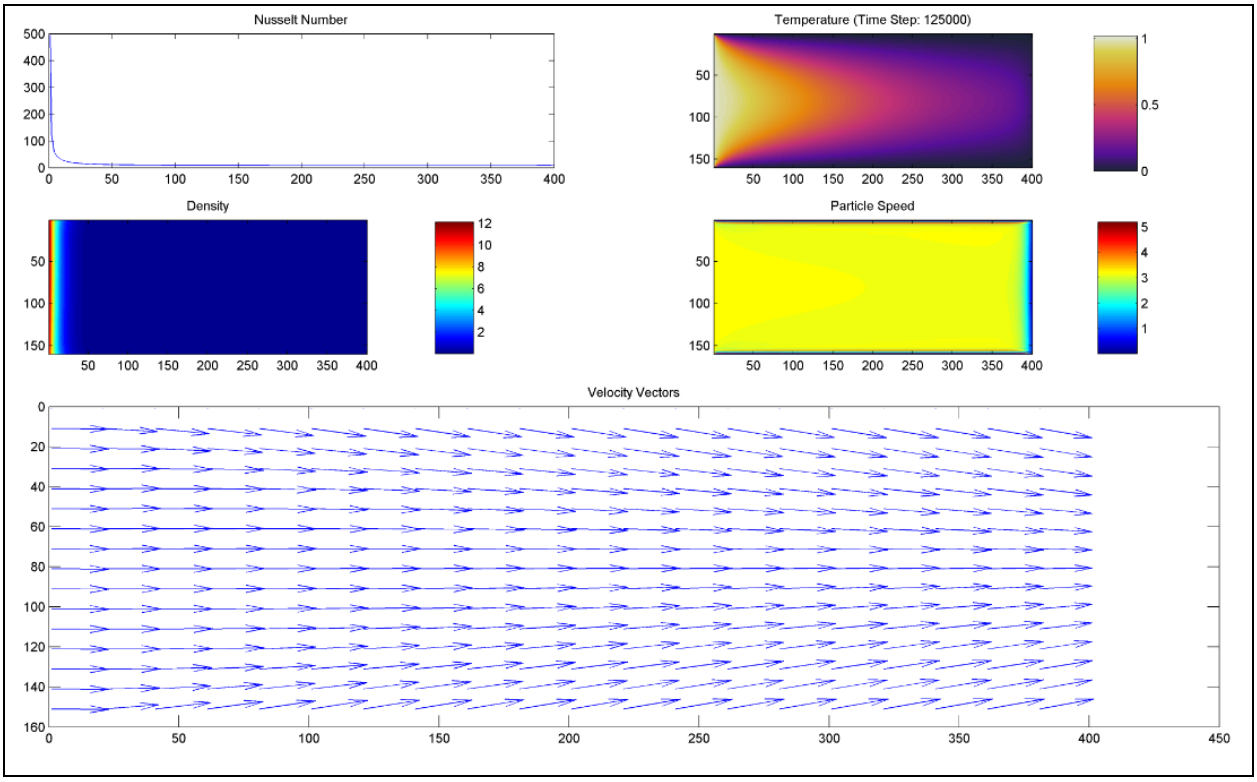


Figure 6-15 LBM model 3 plots at timestep 125000, eps=0.44, K=1e-9

6.4 Discussion of Simulation Experiment Results

The simulation of LBM model was run for different combinations of porosities (ε =0.521; 0.486; 0.440) and permeabilities (K=10^{-7} m^2; 10^{-8} m^2; 10^{-9} m^2). For the simulation good results have been obtained for a porosity of 0.440, and a permeability of 10^{-9} m^2.

A comparison of physical model with the LBM model was presented in Section 6.3.2. The physical travel time calculated for the simulation model had a much smaller value than the travel time in the physical experiment.

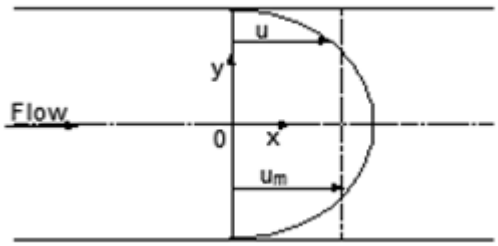
In the LBM model the Nusselt number plotted showed a very abrupt drop, almost immediately after the air flow entered in the porous media (at the left hand side). The Nusselt number represents the ratio of convective to conductive heat transfer across the boundary. The initial Nusselt number was approximately 400 and as simulation progress, and then was vibrating closes the value of 1 (where remain stationary). A higher value of the Nusselt number means that the heat transfer mechanism was more convective than conductive.

The equilibrium state was reached in the cases where the temperature and velocity vectors traveled across the entire domain. The isothermal upper and lower boundary conditions clearly remain all the time. The velocity and temperature profile was similar with Reynolds analogy profiles for temperature and velocity in pipe in a turbulent flow (Kakac *et al*., 2014). The measurements and interpretation of physical experiments results partially validates the LBM simulation code. Velocity and temperature is similar if the boundary conditions are similar. In Fig. 6-15a presented the velocity and temperature profiles in a pipe. This pipe had heated wall, and the temperature profile was changed, opposite than previous cases, where the wall was isothermal and cold temperature. In this figure, the Reynolds analogy was over-simplified, that means is for not only turbulent flow, and this analogy gives quite good agreement with experimental results for gases (Kakac *et al*., 2014).

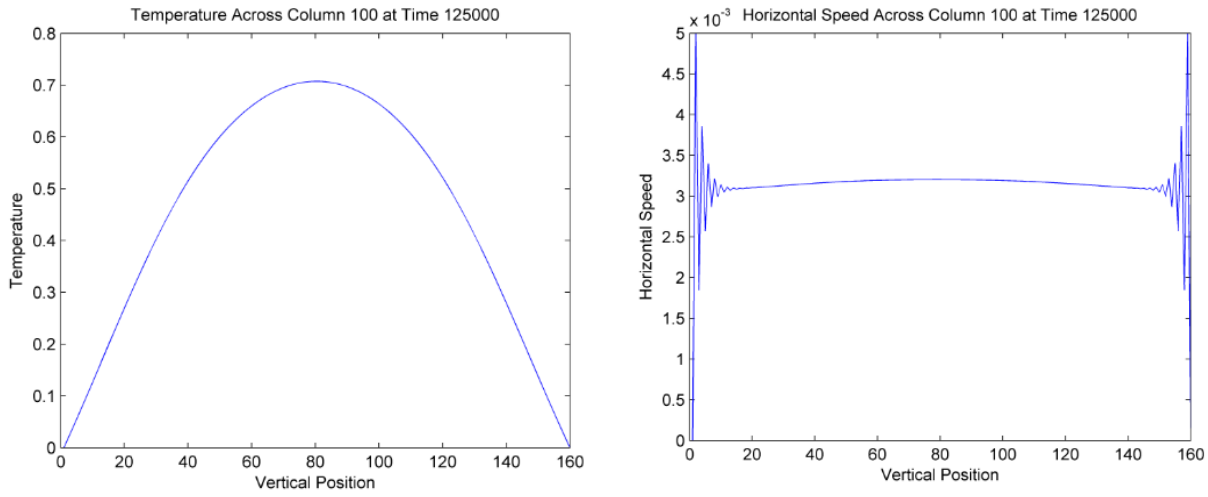
In a paper entitled “On the Extent and Acting of Heat Surface for Steam Boilers”, Osborne Reynolds (1874) suggested that the momentum and heat in a fluid are transferred in the same way. If the temperature of the fluid varies with a direction, the heat flow can be expressed in a similar way in the same direction.

Fig. 6-16b presented the velocity and temperature profile in the simulation, model 3 at timestep 125000. The maximum temperature was situated in the middle of pipe. The velocity profile was slightly different, because there are presented the buoyancy, which influenced the velocity. The value of velocity was starting with value zero at the walls, and then vibrating for short period. After the vibrations of speed were attenuated, in the middle of pipe, the value of velocity was bigger.

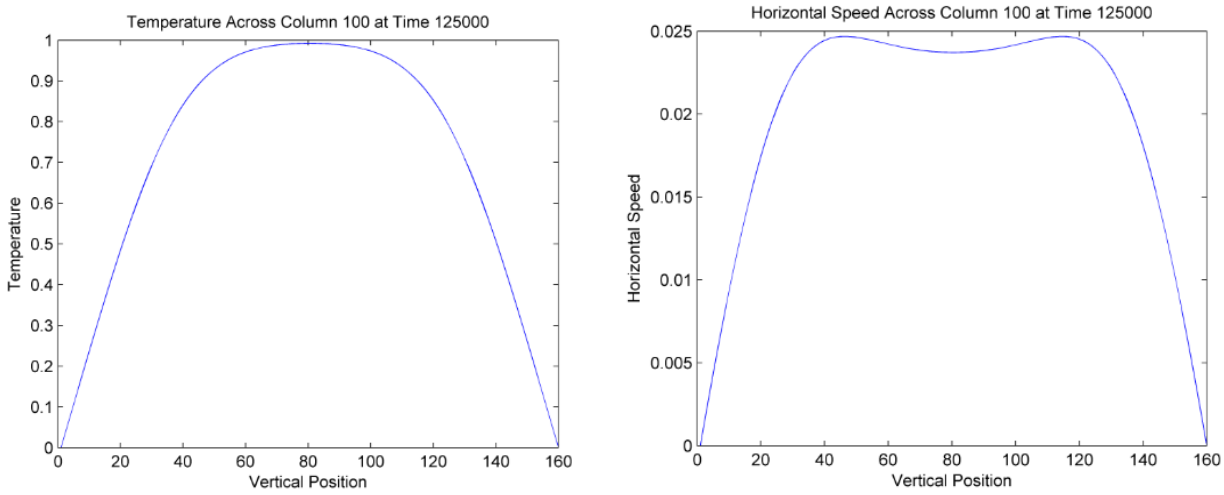
Fig. 6-16c presented the temperature and velocity profile in simulation (model 1) where there is not porosity. The profiles are similar with profiles showed in Fig. 6-16a and Fig. 6-17.



(a)



(b)



(c)

Figure 6-16 Velocity and temperature profiles in a pipe, (a) (Kakac et al., 2014); (b) Model 3 (timestep 125000), and (c) Model 1 (timestep 1250000

Fig. 6-17 presented the solution of the flow and temperature for a steady, fully developed, constant property, constant wall temperature laminar flow in a pipe (people.rit.edu/Laminar Flow Pipe).

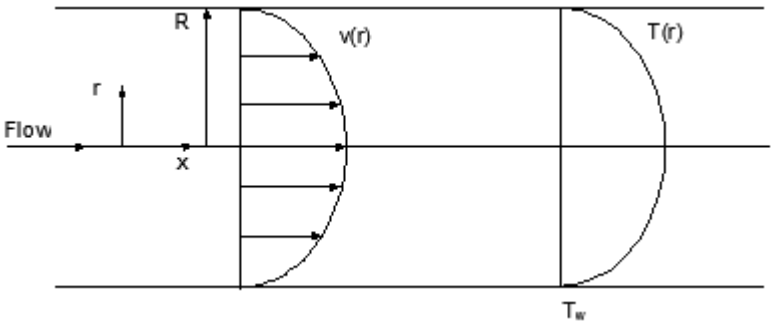


Figure 6-17 Velocity and temperature distribution in a pipe, (people.rit.edu/Laminar Flow Pipe)

7 Discussion, Summary and Conclusions

This chapter contains a discussion of the calculations, and how they relate to the experimental work. It will also show how the code can be used in other applications and what I have learned from this work.

7.1 Whether Calculations are Right or Wrong

In Chapter 6, three separate simulations using an LBM were presented (the last two of a porous medium). These were presented to be considered against the physical experiments reported in Chapter 5, in an attempt to verify the simulations code. The simulations were graded in that they converged on the conditions applying in the experiments.

The Reynolds number calculated was 1276, denoting laminar flow. In the LBM code the Nusselt number did not converge for permeability higher than 10^{-7} m^2, a value meaning very permeable. The temperature used in the code in lattice units was, for the hot temperature equal to 1, and for cold temperature equal to 0, these values mapped to a temperature range of 10K, whereas in the experiment a much higher temperature range was observed.

In the LBM the travel time had a much smaller value (milliseconds), in comparison with the travel time in the physical experiments (seconds); the calculations are Chapter 6. Even through the LBM appears to simulate a very short period of time, a huge computational effort is necessary. The travel time in the physical experiment is calculated with some error, because the velocity was taken as an approximation. However, there remains a big difference between time of the simulation and physical time of the experiment.

While the shapes of velocity and temperature profiles appeared consistent with the expected shapes for laminar flow, at least qualitatively, quantitative verification of the agreement of these profiles with analytical or further experimental activity remains a topic for further work.

Particular challenges with the LBM for a porous media presented in the simulations for, model 1, model 2 and model 3. Model 1 appeared to arrive as steady state after 50000 time steps. For the porous medium model, simulation proceeds for many hours, sometimes then terminating unexpectedly. In the code, it is possible that there is a mistake, or an error of omission or commission. Every effort was taken to eliminate such errors as the work proceeded. Although the work presented did not indicate a completely satisfactory conclusion, it is considered by all involved that it represents an excellent platform on which a final solution LBM can be built.

7.2 How the LBM can be Applied

In the literature, LBM was applied in many areas of fluid dynamics and heat transfer. Supercomputers may be needed to simulate more realistic problems. The computation effort just for a pipe is immense (not practical). It is hard to think of an application where another faster or more convenient method is not available. More applications are potentially in medicine and biology, where the results can justify more computational power.

7.2.1 3D Lattice Model

The MATLAB code is written for two dimensions, the D2Q9 Model, which has speed advantage over the other computational methods. LBM models the fluid at microscopic scale, unlike the CFD methods, which solve the conservation equations of macroscopic properties (mass,

momentum, and energy) numerically. The CFD method builds in viscosity and in the LBM viscous behaviour emerges. This code can be adapted from the 2D Lattice Model to 3D Lattice Model and through these solutions that are more realistic may be obtained by plotting in three dimensions. For 3D dimensions, the model will require more computational power and memory. Hecht and Harting (2012) presented that the advantage using a MATLAB code for D3Q19 is the possibility of specifying the exact position of the boundary, and visualization of the results (especially in dealing with complex boundaries, such as with porous media). If in a pipe, the computational effort for two dimensions is not practical, the computation effort in three dimensions will be too huge, and also, not practical.

7.2.2 Hexagonal Close Packed Lattice

The HCP lattice represents the structure, which contains atoms that were closely packed into the shape of a hexagon.

This model has advantage over other methods and yields more detail at reduced computational cost (Usman *et al*., 2009). The model presents a high degree of symmetry in regular arrangements and it make easier to classify them and to measure their densities, more efficient computation because of weighting factors ($w_a = 0.74$, $a = 1,...6$).

7.2.3 Different Boundary Conditions

Applying correct boundary conditions is a very important part of LBM model. There are several types of boundary condition that can be applied using LBM such as:

- *Periodic boundary conditions areas* the simplest boundary condition, which means the edges are closed; the inlet is outlet boundary along the wall.

- *Bounce-back boundary conditions* are when a particle reaches a wall node; the particle will come back to the fluid nodes along its incoming direction.
- *Dirichlet boundary conditions* specify the pressure/density at the boundaries. For these boundaries, the solution is given by the velocity boundaries.
- *Thermal boundary conditions (*for example: for adiabatic boundary conditions the heat flux should be equal zero). Alternatively, boundaries can be isothermal as in the case of the Simulations of Chapter 6.
- *Extrapolation boundary conditions* are therefore: for any given fluid flow, it was expected that there is one additional layer of sites, past the boundary, inside the wall.

Different authors used different methods for implementing the same boundary conditions. In Chapter 6 LBM (Model 3) stopped when the temperature front through passed the right hand side of the lattice. The velocity profile had developed through out the domain and appeared to be at steady state. Formulation of the boundary conditions is one of the most difficult aspects of LBM, and different authors have reported different methods as approaches to apply them in LBMs. It is suspected that the premature termination of LBM code operation referred to above is a result of a complex interaction in one of the boundary conditions which proved difficult to investigate.

7.2.4 Ice Nodes

Recently LBM have applied for the computational modeling complex problems of fluid flow with single and multiphase flow in porous medium. The fluid with multicomponent and phase change was included in the equations. One of the important studies was for solid-liquid phase change phenomena, also melting or solidification, wich existed in many industrial processes and

in nature. The phase transition for melting ice is needs to be considered the subject for more investigations. During the melting/freezing process in porous medium, there is still a large amount of water that is considered permeable medium. The fluid flow will become restricted if the ice nodes are presented; and at the boundary with ice the bounce back node will occur. A computation code can be developed for LBM with multiphase for melting process to obtain a good optimization in the mechanism of heat transfer (Song *et al*., 2012). Because the presence of ice has influence the heat transfer, that was involved in cooling air for the ventilation in underground mine. More study about this process is left as a project for future investigation.

7.2.5 Natural Heat Exchange Area

The Natural Heat Exchange Area (NHEA) is a caved area containing broken rocks connected to underground mine workings at Creighton Mine near Sudbury. This is the primary intake for the ventilation system of Creighton Mine and is recognized in the world for utilizing the heat storage capacity of fragmented rock to provide low-cost air-conditioning through the year (cooled air during the summer and heated air during the winter). Considering the broken rock mass as porous media at NHEA, the Lattice Boltzmann Method, can be used for modeling air flow and heat transfer. The simulation model allows to modify different parameters which characterize the broken rock mass and analyse the temperature evolution of the air flow. This was the principal objective of the thesis, but this requires too much computation effort for two dimensions, and is not practical.

7.3 Personal Development and Learning Outcomes

7.3.1 Technical Skill

Before starting this thesis my knowledge of the lattice Boltzmann Method was very limited. I thought I knew about the advantages of the method in modeling the heat transfer, based on its relative simplicity and good applicability of computer programming. I was very excited by the prospect of modeling a heat transfer of physical system and using the findings for thermal problems optimization. A particular case, which I had in my mind, was the Creighton's Mine Natural Heat Exchange Area.

In literature reviews related to my research topic, I covered a large number of papers, and books. Even though the LBM was studied and well documented, for the relatively new field of heat transfer in porous media the papers are very limited. The information from the literature reviews provided me with a good understanding of the Lattice Boltzmann method and steps required to build a numerical simulation model.

Another challenge for me was computer programming in MATLAB. I spent a great deal of time learning the basics of this language. Programming skills required to build this model were well beyond my prior experience; the complexity of the task of writing the code for the model required more in depth knowledge, assistance was needed to finish the code. Adam Turcotte, who is a computer programmer at MIRARCO, modified a code from Celik (2012), and finished LBM code that I started.

Through extensive research on LBM simulation technique and producing the MATLAB program to model the heat and mass flow through a porous media, I developed my computer and technical literacy.

7.3.2 Personal Skills

I have developed interpersonal and communication skills meeting people and networking at the 23rd World Mining Congress, Montreal, 2013. I also prepared and made technical presentations to fellow graduate students.

7.3.3 Key Skills

- High level of accuracy and attention to details
- Planning and organizational abilities that balance work, team support and responsibilities in a timely manner
- Resolve problem in a methodical manner independently and in a team work environment to find appropriate solutions
- Ability to listen closely to team discussions and recommendations, while keeping an open mind and noting suggestions to derive maximum benefit from brainstorming session
- Software knowledge: AutoCAD, Microsoft Office, MATLAB

8 References

Aaltosalmi, U., 2005, *Fluid Flow in Porous Media with the Lattice Boltzmann Method*, PhD Thesis, University of Jyvaskyla.

Almalowi, S. J., and Oztekin, A., 2012, "Flow Simulations Using Two Dimensional Thermal Lattice Boltzmann Method", *Journal of Applied Mathematics,* vol. 2012, Article ID 135173, no. 10.1155, pp. 1-12.

ANSYS Fluent Tutorial Guide, 2011

Avila, K., Moxei, D., Lozar, A., Avila, D., and Barkley, D., 2011, "The Onset of Turbulence in Pipe Flow", *Science* vol. 333, pp. 192-196.

Badino, M., 2006, *Probability and Statistic in Boltzmann's Early Papers on Kinetic Theory,* Dublin Core, Chicago.

Bartoloni, A., Battista, C., Cabasino, S., Paolucci, P.S., Pech, J., Sarno, R., Todesco, G.M., Torelli, M., Tross, W., Vincini, P., Benzi, R., Cabibbo, N., Massaioli, F., and Tripiccione R., 1993, "LBE Simulations of Rayleigh-Benard Convection on the APE100 Parallel Processor", *International Jurnal of Modern Physics,* vol. 4, pp. 993-1006.

Basak, T., Roy, S., Singh, A., and Balakrishnan, A.R., 2009, "Natural Convection Flows in Porous Trapezoidal Enclosures with Various Inclination Angles", *International Journal of Heat and Mass Transfer,* vol. 52, no. 19–20, pp. 4612-4623.

Batankar, S. V., 1980, *Numerical Heat Transfer and Fluid Flow,* McGraw-Hill, New York.

Bear, J., 1972, *Dynamics of Fluids in Porous Media,* CRC Press, Boca Raton, Florida.

Begun, R., and Basil, M. A., 2008, *Lattice Boltzmann Method and Its Applications to Fluid Flow Problems*, Euro Journals Publishing Inc.

Bhatnagar, P. L., Gross, E. P., and Krook, M., 1954, "A Model for Collision Process in Gases. Small Amplitude Processes in Charged and Neutral One-component Systems", *Physical Review,* vol. 94, no. 3, pp. 511-525.

Cadorin, M., Calabria, R., Bianchi, E., Chiariello, F., Pinelli, M., Vaccari, A., and P., 2012, "Micro Gas Turbine Fed by Natural Gas", *Journal of Engineering for Gas Turbine and Power,* vol. 134, no. 7, pp. 1-11.

Cai, J., and Huai, X., 2010, "Study on Fluid-solid Coupling Heat Transfer in Fractal Porous Medium by Lattice Boltzmann Method", *Journal of Applied Thermal Engineering,* vol. 30, pp. 715-723.

Cai, J., and Huai, X., 2009, "A Lattice Boltzmann Model for Fluid-solid Coupling Heat Transfer in Fractal Porous Media", *Chinese Physics Letter,* vol. 26, no. 6, pp. 064401-1 to 064401-4.

Canada. Introduction to Hydrology: Permeability versus Impermeability, Water, Water Everywere, 2010, Available at www.clasfaculty.ucdenver.edu (Accessed 16 December 2013).

Carman, P.C., 1937, "Fluid Flow through Granular Bed", *Trans. Inst. Chem. Eng. (London),* vol. 15, pp. 150-156.

Celik, S. B., 2012, *Analysis of Single Phase Fluid Flow and Heat Transfer in Slip Flow Regime by Paralel Implementation of Lattice Boltzmann Method on GPUS,* Master Thesis, Midle East Technical University, Ankara.

Chaikin, F. M., and Lubensky, T. C., 1995, *Principle of Condensed Matter Physics*, Cambridge University Press, Cambridge.

Chen, S., and Doolen, G. D., 1998, "Lattice Boltzmann Method for Fluid Flows", *Annual Rev. Fluid Mech.*, vol. 30, pp. 329-364.

Chen, S., Lee, S. M., and Sheu, J. D.,1998, "Numerical Analysis of Gas Flow in Microchannels", *International Journal of Computation and Methodology,* vol. 33, Issue 7, pp. 749-762.

Chapman, S., and Cowling, T. G., 1970, *The Mathematical Theory of Non-Uniform Gases*, 3rd edition, Cambridge University Library, New-York.

Chirila, B.D., 2010, *Introduction to Lattice Boltzmann Methods*, University of Wuerzburg.

COMSOL, 2014, *Introduction to COMSOL Multiphysics*

Concelliere, A., Chang, C., Fotti, E., Rothman, D.H., and Succi, S., 1990, "The Permeability of Random Medium: Comparison of Simulation with Theory", *Physics of Fluid A,* vol. 2, no. 12, pp. 2085-2088.

Coupez, T., Guillaume, F., and Hachem, E., 2010, *Stablized Finite Elements Methods for High Reynolds Free Surface Flows,* MINES Paris-Technology.

Darcy, H., 1856, *Les Fontaines Publiques da la Ville de Dijon*, Dalmont, Paris.

Darzi, R., A., A., Farhadi, M., Sedighi, K., Fattahi, E., and Nemati, H., 2011, "Mixed Convection Simulation of Inclined Lid Driven Cavity using Lattice Boltzmann Method", *IJST , Transactions of Mechanical Engineering,* vol. 35, pp. 73-83.

D'Orazio, A., and Succi, S., 2003, "Boundary Conditions for Thermal Lattice Boltzmann Simulations", *PMA Slot et al.: ICCS,* vol. LNCS 2657, pp. 977-986.

Dullien, F., A., L., 1979, *Fluid Transport and Pore Structure,* Academic Press, San Diego.

Ergun, S., 1952, "Fluid Flow through Packed Column", *Chemical Engineering Progress,* vol. 48, pp. 89-94.

Fatt, I., 1956, "The Network Model of Porous Media", *Trans. Am. Inst. Min.,* vol. 207, pp. 144-181.

Fava, L., Millar, D., Anderson, B., and Schafrik, S., 2012, "Modeling of Natural Heat Exchange Area at Creighton Mine", MIRARCO, Sudbury

Fei, K., Chen, T., and Hong, C., 2008, "Thermal lattice Boltzmann Simulation of Two-phase Flow at the Anode Microchannel of Micro Direct Methanol Fuel Cells", *1st ASME Micro/Nanoscale Heat Transfer International Conference, MNHT08, January 6, 2008 - January 9* American Society of Mechanical Engineers, Tainan, Taiwan, 2008, pp. 129.

FLOW-3D, 2009, User Manual.

Foti, E., and Succi, S., 1989, "Three Dimensional Flows in Complex Geometries with the Lattice Boltzmann Method", *Europhysics Letters,* vol. 10, no. 5, pp. 433-438.

Fowler, M., 2008, "Kinetic Theory of Gases: A Brief Review", *Physics of Fluid,* vol. 152, pp. 1-13.

Fox, R. and McDonald, A., 1985, *Introduction to Fluid Mechanics,* John Wiley and Sons, Singapore.

Frisch, U., Hasslacher, B., and Pomeau, Y., 1986, "Lattice Gas Automata for Navier-Stokes Equation", *Physical Review Letters,* vol. 56, pp. 1505-1508.

Gao, D., and Chen, Z., 2011, "Lattice Boltzmann Simulation of Natural Convection Dominated Melting in a Rectangular CavitybFilled with Porous Media", *International Journal of Thermal Sciences,* vol. 50, pp. 493-501.

Graebel, W. P., 2001, *Engineering Fluid Mechanics,* Taylor and Francisc, Florence.

Grucelsky, A., and Pozorski, J., 2012, "Lattice Boltzmann Simulation of Fluid Flow in Porous Media of Temperature-Affected Geometry", *Journal of Theoretical and Applied Mechanics,* vol. 50, pp. 193-214.

Guo, Z., Shi, B., and Zheng C., 2002, "A Coupled Lattice BKG Model for the Boussinesq Equations", *International Journal for Numerical Methods in Fluids,* vol. 39, pp. 325-342.

Guo, Z., Zheng C., and Shi, B., 2002, "An Extrapolation Method for Boundary Conditions in Lattice Boltzmann Method", *Physics of Fluid,* vol. 14, pp. 2007-2010.

Guo, Z., and Zhao, T. S., 2002, "Lattice Boltzmann Model for Incompressible Flows through Porous Media", *Physical Review*, vol. 66, pp. 036304-1036304-9.

Guo, Z., and Zhao, T. S., 2005, *A Lattice Boltzmann Model for Convection Heat transfer in Porous Media*, Taylor and Francis, Hong Cong University of Science and Technology, China.

Guo, Z., Chu-Guang, Z., Bao, C., and Zhao, Z. S., 2002, "Non -equilibrium Extrapolation Method for Velocity and Pressure Boundary Conditions in the Lattice Boltzmann Method", *Chinese Physics*, vol. 11, no. 4, pp. 366-374.

Haghshenas, A., Nasr, M. R., and Rahimian, M. H,. 2010, "Numerical Simulation of Convection in an Open-ended Square Cavity Filled with Porous Medium by Lattice Boltzmann Method", *International Communications in Heat and Mass Transfer*, vol. 37, pp. 1513-1519.

Han, Y., and Cundall, A. P., 2011, "Lattice Boltzmann Modeling of Pore -Scale Fluid Flow through Idealized Porous Media", *International Journal for Numerical Methods in Fluids*, vol. 67, pp. 1720-1734.

He, X., and Luo, L. S., 1997, "Theory of Lattice Boltzmann Method: From the Boltzmann Equation to the Lattice Boltzmann Equation", *Physical Review E*, vol. 56, no. 6, pp. 6811-6817.

He, X., and Shi, L., 1997, "Lattice Boltzmann Model for the Incompressible Navier-Stokes Equation", *Journal of Statistical Physics*, vol. 88, no. 3/4, pp. 927-944.

He, X. Y., Shi, B. and Zheng, C., 1998, "A Novel Thermal Model for the Lattice Boltzmann Method in Incompressible Limit", *Journal Computational of Physics*, vol. 146, pp. 282-300.

Hecht, M., and Harting, J., 2012, "Implementation of On-site Velocity Boundary Conditions forD3Q9 Lattice Boltzmann", *Journalof Statistic Dynamics, pp.1-2.*

Inamuro, T., Yoshino, M., and Ogino, F., 1997, "Accuracy of the Lattice Boltzmann Method for Small Knudsen Number with Finite Reynolds Number", *Physics of Fluid*, vol. 9, pp. 3535-3542.

Kakac, S., Yener, Y., and Pramuanjaroenkij, A., 2014, *Convective Heat Transfer*, 3rd Ed., Taylor and Francisc Group, Florida.

Keehm, Y., Mukerji, T., and Nur, A., 2004, "Permeability Prediction from Thin Sections: 3D Reconstruction and Lattice Boltzmann Flow Simulation", *Geophysical Research Letters*, vol. 31, pp. L04606-1 to L04606-4.

Keith, S., 1971, *Mechanics*, 3rd Ed., Addison-Wesley, Massachusetts.

Kinetic Theory of Gases: A Brief Review, 2008, Available at http://galileo.phys.virginia.edu/classes/252/kinetik_theory.html , (Accessed 15 August 2013).

Koponen, A., Kataja, M., and Timonen, J., 1997, "Permeability and Effective Porosity of Porous Media", *Physical Review E,* vol. 56, no. 3, pp. 3319-3325.

Koponen, A., Kataja, M., and Timonen, J., 1996, "Tortuous Flow in Porous Media", *Physical Review,* vol. E 54, no. 1, pp. 406-410.

Kutay, E. M., 2005, *Modeling Moisture Transport in Asphalt Pavements*, PhD Thesis, University of Maryland.

Latief, F. D. E., and Fauzi, U., 2007, "Performance Analysis of 2D and 3D Fluid Flow Modelling Using Lattice Boltzmann Method ", *Indonesian Journal of Physics,* vol. 18, no. 2, pp. 47-52.

Latt, J. 2007, *Hydrodynamic Limit of Lattice Boltzmann Equation*, PhD Thesis, University of Geneva.

Laminar Flow Pipe, Available at People.rit.edu/Laminar Flow Pipe.html, (Accessed 16 January 2014).

Lewis, R. W., Nithiarasu, P., Seetharamu, K., 2004, *Fundamentals of the Finite Element Method for Heat and Fluid Flow,* Wiley.

Li, L., Mei, R., and Klausner, J. F., 2013, "Boundary Conditions for Thermal Lattice Boltzmann Equation Method", *Journal of Computational Physics,* vol. 237, pp. 366-395.

Liao, Q., and Jen, T. C., 2011, "Application of Lattice Boltzmann Method in Fluid Flow and Heat Transfer" in *Computational Fluid Dynamics Technologies and Applications*, ed. Igor V. Minin and Oleg V. Minin, Online edition, pp. 29-68.

Liboff, L. R., 1998, *Kinetic Theory: Classical Quantum and Relativistic Description*, Springer, New York.

Liu, C. H., Lin, K. H., Mai, H. C., and Lin, C. A., 2010, "Thermal Boundary Conditions for Thermal Lattice Boltzmann Simulations", *Journal of Computers and Mathematics with Applications,* vol. 59, pp. 2178-2193.

MacNamara, G. R., and Zanetti, G., 1988, "Use of the Boltzmann Equation to Simulate Lattice Gas Automata", *Physical Review Letters,* vol. 61, no. 20, pp. 2332-2335.

Maier, R., Kroll, D., Kustovsky, Y., Davis, H. T., and Bernard, R., 1997, "Simulation of Flow through Bead Packs Using the Lattice Boltzmann Method", *AIP Publishing LLC,* vol. 10, no. 1, pp. 1-16.

Mehrizi, A. A., Sedighi, K., Afrouzi, H.H., and Aghili, A. A., 2012, "Lattice Boltzmann Simulation of Forced Convection in Vented Cavity Filled by Porous Medium with Obstruction", *World Applied Sciences Journal,* vol. 16, pp. 31-36.

Miguel, A. F., and Serrenho, A., 2007, "On the Experimental Evaluation of Permeability in Porous Media Using a Gas Flow Method", *Journal of Physics D: Applied Physics,* vol. 40, pp. 6824-6828.

Milne-Thomson, L. M., 1973, *Theoretical Aerodynamics*, Dover Publications.

Mohamad, A. A., 2011, *Lattice Boltzmann Method - Fundamentals and Engineering Applications with Computer Codes*, Springer, London.

Muhamed, E. K., 2005, *Modeling Moisture Transport in Asphalt Pavements,* PhD Thesis, University of Maryland.

Munson, B. R., Young, D.F., Okiishi, T.H., and Huesbsch, W.W., 2008, *Fundamentals of Fluid Mechanics*, 6th edition, Wiley.

Narender, R. K., 2011, *Multiphase Simulation Model Using Lattice Boltzmann Method*, University of Toledo.

Niathiarasu, K. N. S., and Sundararajan, T., 1997, "Natural Convective Heat Transfer in a Fluid Saturated Variable Porosity Medium", *International Journal of Heat and Mass Transfer*, vol. 40, no. 16, pp. 3955-3967.

Nield, D. A., and Bejan, A., 1992, *Convection in Porous Media*, 2nd Edition, Springer, New-York.

Nor, A. C. S., 2009, "A 3D Lattice BKG Scheme for Simulation of Thermal Flow in Cubic Cavity", vol. 37, pp. 47-54.

Nor, A. C. S., and Tanahashi T., 2008, "Development of 2-D and 3-D Double-Population Thermal Lattice Models", *Matematika, Department of Matematics, UTM,* vol. 24, no. 1, pp. 53-66.

Pace, D., 2007, *Mean Free Path of the Air*, PhD Thesis, UCLA.

Pan, C., Prins, F. C., and Miller, T. C., 2004, *A High-Performance Lattice Boltzmann Implementation to Model Flow in Porous Media,* Elsevier Science.

Pan, C., Luo, C., Luo, L.S., and Miller, T. C., 2006, "An Evolution of Lattice Boltzmann Schemes for Porous Media Flow Simulation", *Computational Fluids*, vol. 35, pp. 898-909.

Peng, Y., Shu, C., and Chew, Y. T., 2003, "Simplified Thermal Lattice Boltzmann Model for Incompressible Thermal Flows", *APS Journal,* vol. 68, no. 2, pp. 026701-1 to 026701-8.

Peng, Y., Shu, C., and Chew, Y.T., 2003, "A 3D Incompressible Thermal Lattice Boltzmann Model and its Application to Simulate Natural Convection in a Cubic Cavity", *Journal of Computational Physics,* vol. 193, no. 1, pp. 260-274.

Qian, Y. H., D'Humieres, D., and Lallemand, P., 1992, "Lattice BKG Models for Navier-Stokes Equation", *Europhysics Letters*, vol. 17, no. 6, pp. 479-484.

Roberson, J. A., and Crowe, C. T., 1997, *Engineering Fluid Mechanic,* 6th Ed., John Wiley and Sons Inc., New-York.

Rothman, D. H., and Zaleski, S., 1997, *Lattice-gas Cellular Automata*, Cambridge University Press, Cambridge, New York, Melbourne.

Rothman, D.H., 1988, "Cellular-Automatom Fluids: A Model for Fluid Flow in Porous Media", *Geophysics,* vol. 53, no. 4, pp. 509-518.

Sahimi, M., 1995, *Flow and Transport in Porous Media*, VCH, Weinheim.

Samarasinghe, A. M., Huang, Y. H., and Drnevich, V. P., 1982, "Permeability and Consolidation of Normally Consolidated Soils", *Journal of the Geotechnical Engineering Division,* vol. 108, pp. 835-850.

Scheidegger, A. E., 1957, *The Physics of Flow Through Porous Media*, MacMillan, New York.

Seta, T., Takegoshi, E., and Okui, K., 2006, "Mathematics and Computers in Simulation", *Elsevier Science,* vol. 72, no. 2-6, pp. 195-200.

Shan, X., 1997, "Simulation of Raleigh-Benard Convection Using a Lattice -Boltzmann Method", *Physical Review E,* vol. 55, pp. 2780-2788.

Shokouhmand, H., Jam, F., and Salipour, M. R., 2009, "Simulation of Laminar Flow and Convective Heat Transfer in Conduits Filled with Porous Media Using Lattice Boltzmann Method", *International Communications in Heat and Mass Transfer,* vol. 36, pp. 378-384.

Shu, C., Niu, D., and Chew, Y., T., 2005, "A Lattice Boltzmann Kinetic Model for Microflow and Heat Transfer", *Journal of Statistical Physics,* vol. 121, pp. 239-255.

Solidwork-*Computer Aided Design and Manufacturing*, 2014, New-Delhi: Prentice Hall of India.

Song, W., Li, B., and Fu, Z., 2012, "Application of Lattice Boltzmann Method in Heat and Moisture Transfer in Frozen Soil", *World Academy of Science, Engineering and Technology, vol. 71, pp. 1-4.*

Stockman, H. W., 1999, *A 3D Lattice Boltzmann Code for Modeling Flow and Multi-Component Dispersion*, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550.

Succi, S., 2001, *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*, Oxford University Press, Oxford.

Sukop, M., and Thorne, D., 2005, *Lattice Boltzmann Modeling an Introduction for Geoscientists and Engineers*, MS, Miami, Florida, USA.

Sutera, P. S., and Skalak, R., 1993, "The History of Poiseuille's Law", *Annual Review Fluid Mechanic*, vol. 25, pp. 1-19.

Tang, G. H., Tao, W.Q., and He, Y. L., 2005, "Thermal Boundary Condition for the Thermal Lattice Boltzmann Equation", *Physical Review E 72 016703*, pp. 016703-1 to 016703-6.

Texas. Oil and Gas Fundamentals, 2003, MPG Petroleum Inc., Available at: http://mpgpetroleum.com (Accessed 16 December 2013).

Thang, G.H., Tao, W.Q., and He, Y.L., 2005, "Thermal Boundary Condition for the Thermal Lattice Boltzmann Equation ", *Physical Review E 72*, vol. 72, pp. 016703-1 to 016703-6.

Turcotte, D. L., and Schubert, G., 2002, *Geodynamics*, Cambridge University Press, New York.

Turner, J. S., 1973, *Buoyancy Effect in Fluids*, Cambridge University Press, New York.

Usman, R. A., Enlezari, A., and Torsten, M., 2009, "The Lattice Boltzmann Method on Optimal Sampling Lattices", *IEEE Computer Graphycs*, vol. 15, no.10, pp. 1-12.

Vafai, K., 1984, "Convective Flow and Heat Transfer in Variable-Porosity Media", *J. Fluid Mech.*, vol. 147, pp. 233-259. Viggen, E.M., 2009, *The Lattice Boltzmann Method with Applications in Acoustics*, Department of Physics, NTNU.

Viggen, E.M., 2009, *The Lattice Boltzmann Method with Applications in Acoustics*, Department of Physics, NTNU.

Wenyu, S., Bingxi, L., Zhongbin, F., and Zhang, B., 2012, "Application of Lattice Boltzmann Methods in Heat and Moisture Transfer in Frozen Soil", *World Academy of Science Engineering Technology*, vol. 71, pp.1-4.

White, F.M., 1994, *Fluid Mechanics*, McGraw-Hill, New York.

White, F. M., 2006, *Viscous Fluid Flow*, 3rd edition, McGraw-Hill, New York.

Wang, J., Wang, M., and Li, Z., 2007, "A lattice Boltzmann Algorithm for Fluid-solid Conjugate Heat Transfer", *International Journal of Thermal Science*, vol. 46, pp. 228-234.

Wolfgang, P., 1973, *Thermodynamics and the Kinetic Theory of Gases*, 3rd edition, Dover Books on Physics.

Wolf-Gladrow, D.A., 2005, *Lattice Gas Cellular Automata and Lattice Boltzmann Models*, Springer.

Wolf-Gladrow, D.A., 2000, *Lattice Gas Cellular Automata and Lattice Boltzmann Models*, Springer, Bremerhaven, Germany.

Xu, Y., Liu, Y., and Huang, G., 2005, "Lattice Boltzmann Simulation of Momentum and Energy Transfer in a Porous Medium", *Modern Physics Letters B,* vol. 19, no. 28, 29, pp. 1531-1534.

Yang, T. Y., 1986, *Finite Element Structural Analysis*, Prentice-Hall, Englewood.

Yoshino, M., Matsuda, Y., and Shao, C., 2004, "Comparison of Accuracy and Efficiency between the Lattice Boltzmann Method", *International Journal of Computational Fluid Dynamics,* vol. 18, no. 4, pp. 333-345.

Zhang, D., Zhang, R., Chen, S., and Soll, W., 2000, "Pore Scale Study of Flow in Porous Media", *Geophysical Research Letters,* vol. 27, no. 8, pp. 1195-1198.

Zhao, C.Y., Dai, L. N., Tang, G.H., Qu, Z. G., and Li, Z. Y., 2010, "Numerical Study of Natural Convection in Porous Media Using Lattice Boltzmann Method", *International Journal of Heat and Fluid Flow,* vol. 31, pp. 925-934.

Zienkiewicz, O., and Taylor, R. L., 1967, *The Finite Element Method for Solid and Structural Mechanics,* McGraw Hill, New-York.

Zienkiewicz, O., Taylor, R. L., and Zhu, J. Z., 2005, *The Finite Element Method Set*, 6th Ed., Elsevier.

Zhaoli G., and Zhao, T. S., 2005, *A Lattice Boltzmann Model for Convection Heat transfer in Porous Media*, Taylor and Francis, Hong Cong University of Science and Technology, China.

Zhaoli, G., and Chuguang, Z., 2002, "An Extrapolation Method for Boundary Conditions in Lattice Boltzmann Method", *Physics of Fluid,* vol. 14, no. 6, pp. 2007-2010.

Zhaoli, G., Chu-Guang, Z., Bao, C., and Zhao, Z. S., 2002, "Non-equilibrium Extrapolation Method for Velocity and Pressure Boundary Conditions in the Lattice Boltzmann Method", *Chinese Physics,* vol. 11, no. 4, pp. 366-374.

Zou, Q., and He, X., 1997, "On Pressure and Velocity Boundary Conditions for the Lattice Boltzmann BKG Model", *Physics of Fluid,* vol. 9, no. 6, pp. 1591-1596.

Appendix 1

Developed LBM Codes

Model 1, without Porous Media, with Nusselt Number, Temperature, Density, Particle Speed, and Velocity Vectors

For this model, the code was modified and written the final version by Adam Turcotte, from a code written by Celik, 2012.

```
function time = LBM_CPU_RHO(Y)
    % Y is the Vertical Node Number
    % Slit Channel
    % Runs on CPU
    % Flow + Heat Transfer
    % Calculates Nusselt Number

    X = Y*4;
    %tstep = 100;
    tstep = 80000;
    Kn = 0;
    Kappa = 0;
    Lambda = Kn*(Y-1);
    C = Kappa*Lambda;
    Nu = zeros(X,1);
    Re = 10;
    Pe = 100;
    Ulattice = 0.02;
    Pr = Pe/Re;
    Viscosity=Ulattice*(Y-1)/Re;
    alpha = Viscosity/Pr;
    omega = 1/(3*Viscosity+0.5);
    oneMinusOmega = 1-omega;
    omegat = 1/(3*alpha+0.5);
    oneMinusOmegat = 1-omegat;
    rho = zeros(X,Y);
    f1 = rho;f2 = rho;f3 = rho;f4 = rho;f5 = rho;f6 = rho;f7 = rho;f8 =
rho;f9 = rho;
    u = rho; v=rho;
    T = zeros(X,Y);
    g1 = rho;g2 = rho;g3 = rho;g4 = rho;g5 = rho;g6 = rho;g7 = rho;g8 =
rho;g9 = rho;
    rho = rho+5;
    %rho = rho+1;
```

```
    %f1 = (1/9)*rho; f2 = (1/9)*rho; f3 = (1/9)*rho; f4 = (1/9)*rho;
    %f5 = (1/36)*rho; f6 = (1/36)*rho; f7 = (1/36)*rho; f8 = (1/36)*rho;
    %f9 = (4/9)*rho;

    tic
    for kk=1:tstep
        if mod(kk,10)==0
          %---------Nusselt-------------
            %[Nu, Nuold] =
NusseltcalculatorD2Q9(u,rho,Ulattice,U_in,T,X,Y,Height,Kn,Nu);
            [Nu, Nuold] = NusseltcalculatorD2Q9(u,rho,Ulattice,T,X,Y,Kn,Nu);
            NuoldSum = sum(abs(Nuold));
            NuSum = sum(abs(Nu));
            diff = abs(NuSum-NuoldSum);
            if diff<0.001
                display('Program converged and paused');
                savestring =
['Nusselt_','Kn',num2str(Kn),'Kappa',num2str(Kappa),'Pe',num2str(Pe),'Re',num
2str(Re),'N',num2str(Y),'r',num2str(X/Y),'.mat'];
                save(savestring)
                break
            end
            %disp(kk)
            %disp(Nu(max(X)-100))
            %Kn
            %Kappa
            %Y
        end
%% Momentum
    % Collision
        [f1,f2,f3,f4,f5,f6,f7,f8,f9] =
CollisionD2Q9(u,v,rho,f1,f2,f3,f4,f5,f6,f7,f8,f9,omega,oneMinusOmega);
    % Streaming
        [f1,f2,f3,f4,f5,f6,f7,f8] =
StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y);
    % Boundary Conditions
        [f1(1,:),f2(1,:),f4(1,:),f5(1,:),f6(1,:),f7(1,:),f8(1,:)] =
InletconstantVelocityD2Q9(f2(1,:),f3(1,:),f4(1,:),f6(1,:),f7(1,:),f9(1,:),Ula
ttice,Y);
        [f3(X,:),f6(X,:),f7(X,:)] = OutletConstantVelocity(
f1(X,:),f2(X,:),f4(X,:),f5(X,:),f8(X,:),f9(X,:),u(X-10,:));
        %[f3(X,:),f6(X,:),f7(X,:)] = OutletConstantVelocity(
f1(X,:),f2(X,:),f4(X,:),f5(X,:),f8(X,:),f9(X,:),u(X-1,:));
        %[f2(:,1),f5(:,1),f6(:,1)] =
SlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1),f9(:,1),u(:,2:3),K
n,Y);
        %[f4(:,Y),f7(:,Y),f8(:,Y)] =
SlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y),f9(:,Y),u(:,Y-2:Y-
1),Kn,Y);
        [f2(:,1),f5(:,1),f6(:,1)] =
NoSlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1));
        [f4(:,Y),f7(:,Y),f8(:,Y)] =
NoSlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y));
    % Macroscopic Values
        [u,v,rho] = MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y);
%% Heat Diffiusion
    % Collision
```

```
        [g1,g2,g3,g4,g5,g6,g7,g8,g9] =
CollisionTD2Q9(u,v,T,g1,g2,g3,g4,g5,g6,g7,g8,g9,omegat,oneMinusOmegat);
    % Streaming
        [g1,g2,g3,g4,g5,g6,g7,g8] =
StreamingD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,X,Y);
    % Boundary Conditions for Temperature
        g1(1,2:Y-1)=2/9-g3(1,2:Y-1);
        g5(1,2:Y-1)=1/18-g7(1,2:Y-1);
        g8(1,2:Y-1)=1/18-g6(1,2:Y-1);
    % Outlet zero-flux (Extrapolation)
        %g3(X,2:Y-1)=2*g3(X-1,2:Y-1)-g3(X-2,2:Y-1);
        %g6(X,2:Y-1)=2*g6(X-1,2:Y-1)-g6(X-2,2:Y-1);
        %g7(X,2:Y-1)=2*g7(X-1,2:Y-1)-g7(X-2,2:Y-1);
    % OUTLET HACK
        g3(X,2:Y-1) = -g1(X,2:Y-1);
        g6(X,2:Y-1) = -g8(X,2:Y-1);
        g7(X,2:Y-1) = -g5(X,2:Y-1);
    %Top Wall
        %[g4(:,Y),g7(:,Y),g8(:,Y)] = TJumpTop(g2(:,Y),g5(:,Y),g6(:,Y),T(:,[Y-
2:Y-1]),0,C);
        %[g2(:,1),g5(:,1),g6(:,1)] =
TJumpBottom(g4(:,1),g7(:,1),g8(:,1),T(:,2:3),0,C);  % 0 = Wall teperature
        [g4(:,Y),g7(:,Y),g8(:,Y)] = TNoSlipTop(g2(:,Y),g5(:,Y),g6(:,Y),0);
        [g2(:,1),g5(:,1),g6(:,1)] = TNoSlipBottom(g4(:,1),g7(:,1),g8(:,1),0);
    % Macroscopic
        T = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9);
    end
time = toc;
fprintf('CPU: time = %6.4f for %d X %d\n',time,X,Y);
end
%%%%%%%% End of Main Function %%%%%%%%

function [g4,g7,g8] = TJumpTop(g2,g5,g6,T,Tw,C)
    tw = (C*(4*T(:,2)-T(:,1))+2*Tw)/(2+3*C);
    g8=tw/18-g6;
    g7=tw/18-g5;
    g4=2*tw/9-g2;
end
function [g2,g5,g6] = TJumpBottom(g4,g7,g8,T,Tw,C)
    tw = (C*(4*T(:,1)-T(:,2))+2*Tw)/(2+3*C);
    g6=tw/18-g8;
    g5=tw/18-g7;
    g2=2*tw/9-g4;
end

%% ADDED BY ADAM %%
function [g4,g7,g8] = TNoSlipTop(g2,g5,g6,Tw)
    g8=Tw/18-g6;
    g7=Tw/18-g5;
    g4=2*Tw/9-g2;
end
function [g2,g5,g6] = TNoSlipBottom(g4,g7,g8,Tw)
    g6=Tw/18-g8;
    g5=Tw/18-g7;
    g2=2*Tw/9-g4;
end
```

```
function [f1,f2,f3,f4,f5,f6,f7,f8,f9] =
CollisionD2Q9(u,v,rho,f1,f2,f3,f4,f5,f6,f7,f8,f9,omega,oneMinusOmega)
    t10 = u.*u + v.*v;
    t10 = 1.5*t10;
    t1 =  u;
    t2 =  v;
    t3 = -u;
    t4 = -v;
    t5 =  u + v;
    t6 = -u + v;
    t7 = -u - v;
    t8 =  u - v;
    feq1 = rho/9 .*  (1 + 3*t1 + 4.5*t1.*t1 - t10);
    feq2 = rho/9 .*  (1 + 3*t2 + 4.5*t2.*t2 - t10);
    feq3 = rho/9 .*  (1 + 3*t3 + 4.5*t3.*t3 - t10);
    feq4 = rho/9 .*  (1 + 3*t4 + 4.5*t4.*t4 - t10);
    feq5 = rho/36 .* (1 + 3*t5 + 4.5*t5.*t5 - t10);
    feq6 = rho/36 .* (1 + 3*t6 + 4.5*t6.*t6 - t10);
    feq7 = rho/36 .* (1 + 3*t7 + 4.5*t7.*t7 - t10);
    feq8 = rho/36 .* (1 + 3*t8 + 4.5*t8.*t8 - t10);
    feq9 = 4*rho/9 .*(1 - t10);
    f1 = omega*feq1 + oneMinusOmega*f1;
    f2 = omega*feq2 + oneMinusOmega*f2;
    f3 = omega*feq3 + oneMinusOmega*f3;
    f4 = omega*feq4 + oneMinusOmega*f4;
    f5 = omega*feq5 + oneMinusOmega*f5;
    f6 = omega*feq6 + oneMinusOmega*f6;
    f7 = omega*feq7 + oneMinusOmega*f7;
    f8 = omega*feq8 + oneMinusOmega*f8;
    f9 = omega*feq9 + oneMinusOmega*f9;
end
function [g1,g2,g3,g4,g5,g6,g7,g8,g9] =
CollisionTD2Q9(u,v,th,g1,g2,g3,g4,g5,g6,g7,g8,g9,omegat,oneMinusOmegat)
    t1 =  u;
    t2 =  v;
    t3 = -u;
    t4 = -v;
    t5 =  u + v;
    t6 = -u + v;
    t7 = -u - v;
    t8 =  u - v;
    feq1 = th/9 .* (1 + 3*t1 );
    feq2 = th/9 .* (1 + 3*t2 );
    feq3 = th/9 .* (1 + 3*t3 );
    feq4 = th/9 .* (1 + 3*t4 );
    feq5 = th/36 .* (1 + 3*t5 );
    feq6 = th/36 .* (1 + 3*t6 );
    feq7 = th/36 .* (1 + 3*t7 );
    feq8 = th/36 .* (1 + 3*t8 );
    feq9 = 4*th/9;
    g1 = omegat*feq1 + oneMinusOmegat*g1;
    g2 = omegat*feq2 + oneMinusOmegat*g2;
    g3 = omegat*feq3 + oneMinusOmegat*g3;
    g4 = omegat*feq4 + oneMinusOmegat*g4;
    g5 = omegat*feq5 + oneMinusOmegat*g5;
    g6 = omegat*feq6 + oneMinusOmegat*g6;
    g7 = omegat*feq7 + oneMinusOmegat*g7;
```

```
    g8 = omegat*feq8 + oneMinusOmegat*g8;
    g9 = omegat*feq9 + oneMinusOmegat*g9;
end
function [f1,f2,f4,f5,f6,f7,f8] =
InletconstantVelocityD2Q9(f2,f3,f4,f6,f7,f9,Ulattice,Y)
    rhow = (f9 +f2 +f4 +2*(f3 +f6 +f7 ))/(1-Ulattice);
    f1  = f3  + 2*rhow*Ulattice/3;
    f5  = f7  + 0.5*(f4-f2) + rhow*Ulattice/6;
    f8  = f6  + 0.5*(f2-f4) + rhow*Ulattice/6;
end
function [u,v,rho] = MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y)
    rho = f1+f2+f3+f4+f5+f6+f7+f8+f9;
    usum =  f1-f3+f5-f6-f7+f8;
    vsum =  f2-f4+f5+f6-f7-f8;
    u = usum./rho;
    v = vsum./rho;
end
function th = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9)
    th = g1+g2+g3+g4+g5+g6+g7+g8+g9;
end
%function [Nu,Nuold] =
NusseltcalculatorD2Q9(u,rho,Ulattice,U_real,T,X,Y,Height,Kn,Nu)
function [Nu,Nuold] = NusseltcalculatorD2Q9(u,rho,Ulattice,T,X,Y,Kn,Nu)
% Ulattice=umean
Nuold = Nu;
Umean = zeros(X,1);
dy = 1;
ut = zeros(X,Y);
T_mean = zeros(X,1);
A = zeros(X,1);
H = Y-1;
        for i=1:X
            dummy=0;
            %for j=1:5:Y-1
            for j=1:5:Y-5
                dummy=dummy+(5*dy)*(19*rho(i,j)+75*rho(i,j+1)+...

50*rho(i,j+2)+50*rho(i,j+3)+75*rho(i,j+4)+19*rho(i,j+5))/288;
            end
            Rhomean(i)=dummy/H;
            dummy=0;
            %for j=1:5:Y-1
            for j=1:5:Y-5

dummy=dummy+(5*dy)*(19*rho(i,j)*u(i,j)+75*rho(i,j)*u(i,j+1)+...

50*rho(i,j)*u(i,j+2)+50*rho(i,j)*u(i,j+3)+75*rho(i,j)*u(i,j+4)+19*rho(i,j)*u(
i,j+5))/288;
            end

            Umean(i)=dummy/Rhomean(i)/H;

            dummy=0;
            ut(i,:) = rho(i,:).*u(i,:).*T(i,:);
            %for j=1:5:Y-1
            for j=1:5:Y-5
                dummy=dummy+5*dy*(19*ut(i,j)+75*ut(i,j+1)+50*ut(i,j+2)+...
```

```
                    50*ut(i,j+3)+75*ut(i,j+4)+19*ut(i,j+5))/288;
            end
            T_mean(i)=(dummy/Rhomean(i)/Umean(i)/H);
            A(i)= 49/20*T(i,1)-6*T(i,2)+15/2*T(i,3)-20/3*T(i,4)+...
                  15/4*T(i,5)-6/5*T(i,6)+1/6*T(i,7); % derivative
            Nu(i) = -A(i)/T_mean(i)*2*H;

        end
%         subplot(2,2,[1 3])
%         plot(Nu)
%         subplot(2,2,2)
%         imagesc(T')
%         subplot(2,2,4)
%         imagesc(u')
%         drawnow
        subplot(3,2,[1 3 5])
        plot(Nu)
        title('Nusselt Number')
        subplot(3,2,2)
        imagesc(T')
        colorbar;
        title('Temperature')
        subplot(3,2,4)
        imagesc(u')
        colorbar;
        title('Horizontal Velocity')
        subplot(3,2,6)
        imagesc(rho')
        colorbar;
        title('Density')
        drawnow
end
function [ f3,f6,f7] = OutletConstantVelocity( f1,f2,f4,f5,f8,f9,u_out )
    rho_o = (f9+f2+f4+2*(f1+f5+f8))./(1.0+u_out);
    f3 =f1 -0.667*rho_o.*u_out;
    f7 =f5 +0.5*(f2 -f4 )- rho_o.*u_out/6.0;
    f6 =f8 +0.5*(f4 -f2 )- rho_o.*u_out/6.0;
end
function [f2,f5,f6] = SlipBottomWallD2Q9(f1,f3,f4,f7,f8,f9,u,Kn,Y)
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,1)-u(:,2))/(2+3*Lamda);
    rhow = (f1+f3+f9+2*(f4+f7+f8));
    f2 = f4;
    f5 = rhow.*(1+uslip)/2 - (f1+f8) - (f2+f4+f9)/2;
    f6 = rhow.*(1-uslip)/2 - (f3+f7) - (f2+f4+f9)/2;
end
function [f4,f7,f8] = SlipTopWallD2Q9(f1,f2,f3,f5,f6,f9,u,Kn,Y)
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,2)-u(:,1))/(2+3*Lamda);
    rhow = (f1+f3+f9+2*(f2+f5+f6));
    f4 = f2;
    f7 = rhow.*(1-uslip)/2 - (f3+f6) - (f2+f4+f9)/2;
    f8 = rhow.*(1+uslip)/2 - (f1+f5) - (f2+f4+f9)/2;
end

%% ADDED BY ADAM %%
function [f2,f5,f6] = NoSlipBottomWallD2Q9(f1,f3,f4,f7,f8)
```

```
    f2=f4;
    f5=f7-0.5*(f1-f3);
    f6=f8+0.5*(f1-f3);
end
function [f4,f7,f8] = NoSlipTopWallD2Q9(f1,f2,f3,f5,f6)
    f4=f2;
    f7=f5+0.5*(f1-f3);
    f8=f6-0.5*(f1-f3);
end

function [f1a,f2a,f3a,f4a,f5a,f6a,f7a,f8a] =
StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y)
    f1a = f1([1,1:X-1],1:Y);
    f2a = f2(1:X,[1,1:Y-1]);
    f3a = f3([2:X,X],1:Y);
    f4a = f4(1:X,[2:Y,Y]);
    f5a = f5([1,1:X-1],[1,1:Y-1]);
    f6a = f6([2:X,X],[1,1:Y-1]);
    f7a = f7([2:X,X],[2:Y,Y]);
    f8a = f8([1,1:X-1],[2:Y,Y]);

end
```

Model 2, with Porous Media, Nusselt Number, Temperature, Horizontal and Vertical Velocity, and Velocity Vectors

This model was modified and introduced new parameters than Model 1 (porous media, horizontal and vertical velocity, and velocity vectors) by Adam Turcotte.

```
clear all

h     = 1e-2;        % Convective heat transfer coefficient [W/m3-K]
L     = 1.2e-1;      % Characteristic length  [m]
k     = 2.57e-2;     % Thermal conductivity  [W/m-K]
lamda = 2.3787e-8;   % Mean free path  [m]
u     = 1e-2;        % Mean velocity of fluid  [m/s]
nu    = 1.57e-5;     % Kinematic viscosity  [m2/s]
alpha = 2.1606e-5;   % Thermal diffusivity  [m2/s]
cp    = 1.005;       % Specific heat  [J/kg-K]
niu   = 1.983e-5;    % Dynamic viscosity [kg/m-s]
g     = 9.8;         % Acceleration due to gravity  [m/s2]
beta  = 3.43e-3;     % Thermal expansion coefficient  [1/K]
T1    = 283;         % Hot themperature  [K]
T0    = 273;         % Cold temperature  [K]
K     = 1e-8;        % Permeability of porous media  [m2]

% The Nusselt number
% Ratio of convective to conductive heat transfer
Nu = h*L/k;

% The Knudsen number
% Ratio between mean free path and physical length
Kn = lamda/L;

% The Reynolds number
% Ratio of inertial to viscous forces
Re = u*L/nu;

% The Peclet number
% Ratio of advective to diffusive transport rate
Pe = u*L/alpha;

% The Prandtl number
% Ratio of viscous to thermal diffusion rate
Pr = nu/alpha;

% The Rayleigh number
% Measure of the instability of a layer of fluid due to differences of
% temperature and density at the top and bottom
Ra = g*beta*(T1-T0)*L^3/(nu*alpha);
```

```
% The Darcy number
% Relative effect of the permeability of the medium to diameter squared
Da = K/L^2;

DLN(1) = Nu;
DLN(2) = Kn;
DLN(3) = Re;
DLN(4) = Pe;
DLN(5) = Pr;
DLN(6) = Ra;
DLN(7) = Da;

%NN(1) = 41;
%NN(2) = 41;
NN(1) = 161;
NN(2) = 400;

bugger=LBM_THESIS_ALPHA1(NN,DLN);
```

```
function time = LBM_THESIS_ALPHA1(NN,DLN)
    % Y is the Vertical Node Number
    % Slit Channel
    % Flow + Heat Transfer
    % Calculates Nusselt Number

    % Colormap that estimates the range of a thermal camera
    global THERMAL_MAP;
    THERMAL_MAP = [           0.1176,0.1412,0.2275; 0.1451,0.1314,0.2735;
        0.1725,0.1216,0.3196; 0.2000,0.1118,0.3657; 0.2275,0.1020,0.4118;
        0.2549,0.0922,0.4578; 0.2824,0.0824,0.5039; 0.3098,0.0725,0.5500;
        0.3373,0.0627,0.5961; 0.3637,0.0711,0.5875; 0.3902,0.0794,0.5789;
        0.4167,0.0877,0.5703; 0.4431,0.0961,0.5618; 0.4696,0.1044,0.5532;
        0.4961,0.1127,0.5446; 0.5225,0.1211,0.5360; 0.5490,0.1294,0.5275;
        0.5755,0.1377,0.5189; 0.6020,0.1461,0.5103; 0.6284,0.1544,0.5017;
        0.6549,0.1627,0.4931; 0.6814,0.1711,0.4846; 0.7078,0.1794,0.4760;
        0.7343,0.1877,0.4674; 0.7608,0.1961,0.4588; 0.7696,0.2167,0.4336;
        0.7784,0.2373,0.4083; 0.7873,0.2578,0.3831; 0.7961,0.2784,0.3578;
        0.8049,0.2990,0.3326; 0.8137,0.3196,0.3074; 0.8225,0.3402,0.2821;
        0.8314,0.3608,0.2569; 0.8402,0.3814,0.2316; 0.8490,0.4020,0.2064;
        0.8578,0.4225,0.1811; 0.8667,0.4431,0.1559; 0.8755,0.4637,0.1306;
        0.8843,0.4843,0.1054; 0.8931,0.5049,0.0801; 0.9020,0.5255,0.0549;
        0.8985,0.5434,0.0703; 0.8951,0.5613,0.0858; 0.8917,0.5792,0.1012;
        0.8882,0.5971,0.1167; 0.8848,0.6150,0.1321; 0.8814,0.6328,0.1475;
        0.8779,0.6507,0.1630; 0.8745,0.6686,0.1784; 0.8711,0.6865,0.1939;
        0.8676,0.7044,0.2093; 0.8642,0.7223,0.2248; 0.8608,0.7402,0.2402;
        0.8574,0.7581,0.2556; 0.8539,0.7760,0.2711; 0.8505,0.7939,0.2865;
        0.8471,0.8118,0.3020; 0.8504,0.8230,0.3782; 0.8538,0.8342,0.4543;
        0.8571,0.8454,0.5305; 0.8605,0.8566,0.6067; 0.8639,0.8678,0.6829;
        0.8672,0.8790,0.7591; 0.8706,0.8902,0.8353];

    %% SIMULATION CONFIGURATION %%
    Y      = NN(1);  % Number of nodes in the vertical direction
    X      = NN(2);  % Number of nodes in the horizontal direction
    tstep  = 160000; % Maximum number of time steps
    tplot  = 10;     % Iterations between successive graphical outputs
    thresh = 0.001;  % Nusselt difference threshold
    %% DIMENSIONLESS NUMBERS %%
    %Kn = 2.38e-7;    % Knudsen number
    %Ra = 9.55e+5;    % Rayleigh number (gbeta*T_diff*L^3)/(Viscosity*alpha)
    %Nu = zeros(X,1); % Nusselt number
    %Re = 6.38e+3;    % Reynolds number (L*Ulattice/Viscosity)
    %Pe = 4.63e+3;    % Péclet number (Re*Pr)
    %Pr = Pe/Re;      % Prandtl number (L*Ulattice/Diffusivity)
    %Da = 1e-7;       % Darcy number

    Nu = zeros(X,1);  % Nusselt number
    Kn = DLN(2);      % Knudsen number
    Re = DLN(3);      % Reynolds number (L*Ulattice/Viscosity)
    Pe = DLN(4);      % Péclet number (Re*Pr)
    Pr = DLN(5);      % Prandtl number
    Ra = DLN(6);      % Rayleigh number
    Da = DLN(7);      % Darcy number
```

```
%% CHARACTERISTIC LENGTH AND VELOCITY %%
L        = Y-1;  % Characteristic length
Ulattice = 0.02; % Characteristic velocity

%% SLIP BOUNDARY VALUES %%
Kappa  = 0;            % Temperature jump parameter
Lambda = Kn*L;         % Mean free path
C      = Kappa*Lambda; % Temperature jump coefficient
%% FLUID PROPERTIES %%
Viscosity = Ulattice*L/Re; % Kinematic viscosity
alpha     = Viscosity/Pr;  % Thermal diffusivity
%% POROUS MEDIUM %%
eps  = 0.440;               % Porosity
Feps = 1.75/sqrt(150*eps^3); % Geometric function
K    = Da*L^2;
%% TEMPERATURE %%
T_hot    = 1.0;                % Hot temperature (normalized)
T_cold   = 0.0;                % Cold temperature (normalized)
T_inlet  = T_hot;              % Inlet temperature
T_top    = T_cold;             % Top wall temperature
T_bottom = T_cold;             % Bottom wall temperature
%T_top    = T_hot;
%T_bottom = T_hot;
T_init   = T_cold;             % Initial temperature
T_diff   = T_hot-T_cold;       % Temperature difference
T0       = 0.5*(T_hot+T_cold); % Reference temperature
%% BODY FORCE %%
gbeta    = Ra*Viscosity*alpha/(T_diff*L^3);
buoyancy = [0,gbeta]; % Buoyancy force
a        = [0,0];     % Acceleration due to external force fields
%% RELAXATION PARAMETERS %%
omega             = 1 / (3*Viscosity + 0.5);
omegat            = 1 / (3*alpha     + 0.5);
oneMinusOmega     = 1 - omega;
oneMinusHalfOmega = 1 - 0.5*omega;
oneMinusOmegat    = 1 - omegat;
%% OTHER CONSTANTS %%
c0 = 0.5*(1+0.5*eps*Viscosity/K);
c1 = 0.5*eps*Feps/sqrt(K);
d1 = -eps*Viscosity/K;
d2 = eps*Feps/sqrt(K);
%% INITIAL CONDITIONS %%
rho = ones(X,Y);         % Fluid density
Fx  = zeros(X,Y);        % Total body force (horizontal component)
Fy  = zeros(X,Y);        % Total body force (vertical component)
f1  = zeros(X,Y);        % Density distribution function
f2  = zeros(X,Y);        % Density distribution function
f3  = zeros(X,Y);        % Density distribution function
f4  = zeros(X,Y);        % Density distribution function
f5  = zeros(X,Y);        % Density distribution function
f6  = zeros(X,Y);        % Density distribution function
f7  = zeros(X,Y);        % Density distribution function
f8  = zeros(X,Y);        % Density distribution function
f9  = zeros(X,Y);        % Density distribution function
u   = zeros(X,Y);        % Fluid velocity (horizontal component)
v   = zeros(X,Y);        % Fluid velocity (vertical component)
T   = T_init*ones(X,Y); % Fluid temperature
```

```
    g1  = zeros(X,Y);       % Temperature distribution function
    g2  = zeros(X,Y);       % Temperature distribution function
    g3  = zeros(X,Y);       % Temperature distribution function
    g4  = zeros(X,Y);       % Temperature distribution function
    g5  = zeros(X,Y);       % Temperature distribution function
    g6  = zeros(X,Y);       % Temperature distribution function
    g7  = zeros(X,Y);       % Temperature distribution function
    g8  = zeros(X,Y);       % Temperature distribution function
    g9  = zeros(X,Y);       % Temperature distribution function

    %% COMPUTE NUSSELT NUMBER AND PLOT RESULTS %%
    tic
    for kk=1:tstep
        if mod(kk,tplot)==0
            [Nu, Nuold] = ...
                NusseltcalculatorD2Q9(u,v,rho,Ulattice,T,X,Y,Kn,Nu);
            NuoldSum = sum(abs(Nuold));
            NuSum = sum(abs(Nu));
            diff = abs(NuSum-NuoldSum);
            if diff<thresh
                display('Program converged and paused');
                savestring = ['Nusselt_','Kn',num2str(Kn),'Kappa',...
                    num2str(Kappa),'Pe',num2str(Pe),'Re',num2str(Re),...
                    'N',num2str(Y),'r',num2str(X/Y),'.mat'];
                save(savestring)
                break
            end
            %disp(kk)
            %disp(Nu(max(X)-100))
            %Kn
            %Kappa
            %Y
        end

%% MOMENTUM TRANSFER %%
    % COLLISION
        [f1,f2,f3,f4,f5,f6,f7,f8,f9] = ...
            CollisionD2Q9(u,v,rho,Fx,Fy,eps,f1,f2,f3,f4,f5,f6,f7,f8,f9,...
                          omega,oneMinusOmega,oneMinusHalfOmega);
    % STREAMING
        [f1,f2,f3,f4,f5,f6,f7,f8] = ...
            StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y);
    % BOUNDARY CONDITIONS
        [f1(1,:),f2(1,:),f4(1,:),f5(1,:),f6(1,:),f7(1,:),f8(1,:)] = ...
            InletconstantVelocityD2Q9(f2(1,:),f3(1,:),f4(1,:),f6(1,:),...
                                      f7(1,:),f9(1,:),Ulattice,Y);
        [f3(X,:),f6(X,:),f7(X,:)] = ...
            OutletConstantVelocity(f1(X,:),f2(X,:),f4(X,:),f5(X,:),...
                                   f8(X,:),f9(X,:),u(X-10,:));
        %[f2(:,1),f5(:,1),f6(:,1)] = ...
        %    SlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1),...
        %                       f9(:,1),u(:,2:3),Kn,Y);
        %[f4(:,Y),f7(:,Y),f8(:,Y)] = ...
        %    SlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y),...
        %                    f9(:,Y),u(:,Y-2:Y-1),Kn,Y);
        [f2(:,1),f5(:,1),f6(:,1)] = ...
```

```
            NoSlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1));
        [f4(:,Y),f7(:,Y),f8(:,Y)] = ...
            NoSlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y));
    % MACROSCOPIC PROPERTY CALCULATION
        [u,v,rho,Fx,Fy] = ...
            MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y,...
                            eps,buoyancy,a,T_diff,c0,c1,d1,d2,T,T0);
%% HEAT TRANSFER %%
    % COLLISION
        [g1,g2,g3,g4,g5,g6,g7,g8,g9] = ...
            CollisionTD2Q9(u,v,T,g1,g2,g3,g4,g5,g6,g7,g8,g9,...
                           omegat,oneMinusOmegat);
    % STREAMING
        [g1,g2,g3,g4,g5,g6,g7,g8] = ...
            StreamingD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,X,Y);
    % BOUNDARY CONDITIONS
        % INLET
        g1(1,2:Y-1) = T_inlet*2/9  - g3(1,2:Y-1);
        g5(1,2:Y-1) = T_inlet/18 - g7(1,2:Y-1);
        g8(1,2:Y-1) = T_inlet/18 - g6(1,2:Y-1);
        % OUTLET (ZERO-FLUX; EXTRAPOLATION)
        g3(X,2:Y-1) = 2*g3(X-1,2:Y-1) - g3(X-2,2:Y-1);
        g6(X,2:Y-1) = 2*g6(X-1,2:Y-1) - g6(X-2,2:Y-1);
        g7(X,2:Y-1) = 2*g7(X-1,2:Y-1) - g7(X-2,2:Y-1);
        % TOP WALL
        %[g4(:,Y),g7(:,Y),g8(:,Y)] = ...
        %    TJumpTop(g2(:,Y),g5(:,Y),g6(:,Y),T(:,[Y-2:Y-1]),T_top,C);
        [g4(:,Y),g7(:,Y),g8(:,Y)] = ...
            TNoSlipTop(g2(:,Y),g5(:,Y),g6(:,Y),T_top);
        % BOTTOM WALL
        %[g2(:,1),g5(:,1),g6(:,1)] = ...
        %    TJumpBottom(g4(:,1),g7(:,1),g8(:,1),T(:,2:3),T_bottom,C);
        [g2(:,1),g5(:,1),g6(:,1)] = ...
            TNoSlipBottom(g4(:,1),g7(:,1),g8(:,1),T_bottom);
    % MACROSCOPIC PROPERTY CALCULATION
        T = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9);
    end
time = toc;
fprintf('CPU: time = %6.4f for %d X %d\n',time,X,Y);
end

%% TEMPERATURE BOUNDARY CONDITIONS %%
function [g4,g7,g8] = TNoSlipTop(g2,g5,g6,Tw)
    g8 = Tw/18 - g6;
    g7 = Tw/18 - g5;
    g4 = 2*Tw/9 - g2;
end
function [g2,g5,g6] = TNoSlipBottom(g4,g7,g8,Tw)
    g6 = Tw/18 - g8;
    g5 = Tw/18 - g7;
    g2 = 2*Tw/9 - g4;
end
%% TEMPERATURE JUMP BOUNDARY CONDITIONS
function [g4,g7,g8] = TJumpTop(g2,g5,g6,T,Tw,C)
    tw = (C*(4*T(:,2)-T(:,1))+2*Tw)/(2+3*C);
    g8 = tw/18 - g6;
    g7 = tw/18 - g5;
```

```
    g4 = 2*tw/9 - g2;
end
function [g2,g5,g6] = TJumpBottom(g4,g7,g8,T,Tw,C)
    tw = (C*(4*T(:,1)-T(:,2))+2*Tw)/(2+3*C);
    g6 = tw/18 - g8;
    g5 = tw/18 - g7;
    g2 = 2*tw/9 - g4;
end
%% COLLISION FOR MOMENTUM TRANSFER %%
function [f1,f2,f3,f4,f5,f6,f7,f8,f9] = ...
    CollisionD2Q9(u,v,rho,Fx,Fy,eps,f1,f2,f3,f4,f5,f6,f7,f8,f9,...
                  omega,oneMinusOmega,oneMinusHalfOmega)
    t10    = u.*u + v.*v;
    t10    = -1.5/eps*t10;
    t1     =  u;
    t2     =  v;
    t3     = -u;
    t4     = -v;
    t5     =  u + v;
    t6     = -u + v;
    t7     = -u - v;
    t8     =  u - v;
    a10    = u.*Fx + v.*Fy;
    a10    = -3.0/eps*a10;
    a1     =  Fx;
    a2     =  Fy;
    a3     = -Fx;
    a4     = -Fy;
    a5     =  Fx + Fy;
    a6     = -Fx + Fy;
    a7     = -Fx - Fy;
    a8     =  Fx - Fy;
    a11    = u.*Fx;
    a12    = v.*Fy;
    a13    = a11;
    a14    = a12;
    a15    = u.*Fx + v.*Fx + u.*Fy + v.*Fy;
    a16    = u.*Fx - v.*Fx - u.*Fy + v.*Fy;
    a17    = a15;
    a18    = a16;
    feq1   = rho/9   .* (1 + 3*t1 + (4.5/eps)*t1.*t1 + t10);
    feq2   = rho/9   .* (1 + 3*t2 + (4.5/eps)*t2.*t2 + t10);
    feq3   = rho/9   .* (1 + 3*t3 + (4.5/eps)*t3.*t3 + t10);
    feq4   = rho/9   .* (1 + 3*t4 + (4.5/eps)*t4.*t4 + t10);
    feq5   = rho/36  .* (1 + 3*t5 + (4.5/eps)*t5.*t5 + t10);
    feq6   = rho/36  .* (1 + 3*t6 + (4.5/eps)*t6.*t6 + t10);
    feq7   = rho/36  .* (1 + 3*t7 + (4.5/eps)*t7.*t7 + t10);
    feq8   = rho/36  .* (1 + 3*t8 + (4.5/eps)*t8.*t8 + t10);
    feq9   = 4*rho/9 .* (1 + t10);
    force1 = oneMinusHalfOmega*rho/9   .* (3*a1 + (9/eps)*a11 + a10);
    force2 = oneMinusHalfOmega*rho/9   .* (3*a2 + (9/eps)*a12 + a10);
    force3 = oneMinusHalfOmega*rho/9   .* (3*a3 + (9/eps)*a13 + a10);
    force4 = oneMinusHalfOmega*rho/9   .* (3*a4 + (9/eps)*a14 + a10);
    force5 = oneMinusHalfOmega*rho/36  .* (3*a5 + (9/eps)*a15 + a10);
    force6 = oneMinusHalfOmega*rho/36  .* (3*a6 + (9/eps)*a16 + a10);
    force7 = oneMinusHalfOmega*rho/36  .* (3*a7 + (9/eps)*a17 + a10);
    force8 = oneMinusHalfOmega*rho/36  .* (3*a8 + (9/eps)*a18 + a10);
```

```
    force9 = 4*oneMinusHalfOmega*rho/9 .* a10;
    f1     = omega*feq1 + oneMinusOmega*f1 + force1;
    f2     = omega*feq2 + oneMinusOmega*f2 + force2;
    f3     = omega*feq3 + oneMinusOmega*f3 + force3;
    f4     = omega*feq4 + oneMinusOmega*f4 + force4;
    f5     = omega*feq5 + oneMinusOmega*f5 + force5;
    f6     = omega*feq6 + oneMinusOmega*f6 + force6;
    f7     = omega*feq7 + oneMinusOmega*f7 + force7;
    f8     = omega*feq8 + oneMinusOmega*f8 + force8;
    f9     = omega*feq9 + oneMinusOmega*f9 + force9;
end
%% COLLISION FOR HEAT TRANSFER %%
function [g1,g2,g3,g4,g5,g6,g7,g8,g9] = ...`
    CollisionTD2Q9(u,v,th,g1,g2,g3,g4,g5,g6,g7,g8,g9,omegat,oneMinusOmegat)
    t1 =  u;
    t2 =  v;
    t3 = -u;
    t4 = -v;
    t5 =  u + v;
    t6 = -u + v;
    t7 = -u - v;
    t8 =  u - v;
    feq1 = th/9  .* (1 + 3*t1);
    feq2 = th/9  .* (1 + 3*t2);
    feq3 = th/9  .* (1 + 3*t3);
    feq4 = th/9  .* (1 + 3*t4);
    feq5 = th/36 .* (1 + 3*t5);
    feq6 = th/36 .* (1 + 3*t6);
    feq7 = th/36 .* (1 + 3*t7);
    feq8 = th/36 .* (1 + 3*t8);
    feq9 = 4*th/9;
    g1 = omegat*feq1 + oneMinusOmegat*g1;
    g2 = omegat*feq2 + oneMinusOmegat*g2;
    g3 = omegat*feq3 + oneMinusOmegat*g3;
    g4 = omegat*feq4 + oneMinusOmegat*g4;
    g5 = omegat*feq5 + oneMinusOmegat*g5;
    g6 = omegat*feq6 + oneMinusOmegat*g6;
    g7 = omegat*feq7 + oneMinusOmegat*g7;
    g8 = omegat*feq8 + oneMinusOmegat*g8;
    g9 = omegat*feq9 + oneMinusOmegat*g9;
end
%% INLET BOUNDARY CONDITIONS %%
function [f1,f2,f4,f5,f6,f7,f8] = ...
    InletconstantVelocityD2Q9(f2,f3,f4,f6,f7,f9,Ulattice,Y)
    rhow = (f9+f2+f4 + 2*(f3+f6+f7))/(1-Ulattice);
    f1  = f3 + 2*rhow*Ulattice/3;
    f5  = f7 + 0.5*(f4-f2) + rhow*Ulattice/6;
    f8  = f6 + 0.5*(f2-f4) + rhow*Ulattice/6;
end
%% MACROSCOPIC PROPERTY CALCULATION FOR MOMENTUM TRANSFER %%
function [u,v,rho,Fx,Fy] = ...
    MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y,...
                    eps,buoyancy,a,T_diff,c0,c1,d1,d2,T,T0)
    rho     = f1+f2+f3+f4+f5+f6+f7+f8+f9;
    usum    = f1-f3+f5-f6-f7+f8;
    vsum    = f2-f4+f5+f6-f7-f8;
    %Gx      = buoyancy(1)*T_diff+a(1);  % MIGHT HAVE TO COMPUTE A SPECIAL
```

```
    %Gy      = buoyancy(2)*T_diff+a(2);  % T_diff ON EACH ITERATION
    dT       = T-T0;
    Gx       = buoyancy(1)*dT+a(1);
    Gy       = buoyancy(2)*dT+a(2);
    aux_u    = usum./rho + 0.5*eps*Gx;
    aux_v    = vsum./rho + 0.5*eps*Gy;
    aux_den = c0+sqrt(c0*c0+c1*sqrt(aux_u.*aux_u+aux_v.*aux_v));
    u        = aux_u./aux_den;
    v        = aux_v./aux_den;
    uv_mod   = sqrt(u.*u+v.*v);
    Fx       = d1*u-d2*uv_mod.*u+eps*Gx;
    Fy       = d1*v-d2*uv_mod.*v+eps*Gy;
end
%% MACROSCOPIC PROPERTY CALCULATION FOR HEAT TRANSFER %%
function th = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9)
    th = g1+g2+g3+g4+g5+g6+g7+g8+g9;
end
%% COMPUTE NUSSELT NUMBER AND PLOT RESULTS %%
function [Nu,Nuold] = NusseltcalculatorD2Q9(u,v, rho,Ulattice,T,X,Y,Kn,Nu)
    global THERMAL_MAP;
    Nuold = Nu;
    Umean = zeros(X,1);
    dy = 1;
    ut = zeros(X,Y);
    T_mean = zeros(X,1);
    A = zeros(X,1);
    H = Y-1;

    for i=1:X
        dummy=0;
        %for j=1:5:Y-1
        for j=1:5:Y-5
            dummy=dummy+(5*dy)*(19*rho(i,j)+75*rho(i,j+1)+...
                50*rho(i,j+2)+50*rho(i,j+3)+75*rho(i,j+4)+...
                19*rho(i,j+5))/288;
        end
        Rhomean(i)=dummy/H;
        dummy=0;
        %for j=1:5:Y-1
        for j=1:5:Y-5
            dummy=dummy+(5*dy)*(19*rho(i,j)*u(i,j)+...
                75*rho(i,j)*u(i,j+1)+50*rho(i,j)*u(i,j+2)+...
                50*rho(i,j)*u(i,j+3)+75*rho(i,j)*u(i,j+4)+...
                19*rho(i,j)*u(i,j+5))/288;
        end

        Umean(i)=dummy/Rhomean(i)/H;

        dummy=0;
        ut(i,:) = rho(i,:).*u(i,:).*T(i,:);
        %for j=1:5:Y-1
        for j=1:5:Y-5
            dummy=dummy+5*dy*(19*ut(i,j)+75*ut(i,j+1)+50*ut(i,j+2)+...
                50*ut(i,j+3)+75*ut(i,j+4)+19*ut(i,j+5))/288;
        end
        T_mean(i)=(dummy/Rhomean(i)/Umean(i)/H);
```

```
        A(i)= 49/20*T(i,1)-6*T(i,2)+15/2*T(i,3)-20/3*T(i,4)+...
            15/4*T(i,5)-6/5*T(i,6)+1/6*T(i,7); % derivative
        Nu(i) = -A(i)/T_mean(i)*2*H;
    end

    % PLOT RESULTS
%   subplot(2,2,[1 3])
    subplot(4,2,1)
    plot(Nu)
    title('Nusselt Number')
    subplot(4,2,2)
    imagesc(T(:,Y:-1:1)')
    colormap(THERMAL_MAP);
    freezeColors
    title('Temperature')
    subplot(4,2,3)
    imagesc(u(:,Y:-1:1)')
    colormap('default');
    freezeColors
    title('Horizontal Velocity')
    subplot(4,2,4)
    imagesc(v(:,Y:-1:1)')
    colormap('default');
    freezeColors
    title('Vertical velocity')
    subplot(4,2,[5 6 7 8])
    NN=10;
    MM=10;
    uuu=u(1:NN:X,1:MM:Y);
    vvv=v(1:NN:X,1:MM:Y);
    [y,x] = meshgrid(1:MM:Y,1:NN:X);
    quiver(x,y,uuu,vvv)
    title('Velocity vectors')

    drawnow
end
%% OUTLET BOUNDARY CONDITIONS %%
function [f3,f6,f7] = OutletConstantVelocity(f1,f2,f4,f5,f8,f9,u_out)
    rho_o = (f9+f2+f4+2*(f1+f5+f8))./(1.0+u_out);
    f3    = f1 - 0.667*rho_o.*u_out;
    f7    = f5 + 0.5*(f2-f4) - rho_o.*u_out/6.0;
    f6    = f8 + 0.5*(f4-f2) - rho_o.*u_out/6.0;
end
%% NO-SLIP BOUNDARY CONDITIONS %%
function [f2,f5,f6] = NoSlipBottomWallD2Q9(f1,f3,f4,f7,f8)
    f2 = f4;
    f5 = f7 - 0.5*(f1-f3);
    f6 = f8 + 0.5*(f1-f3);
end
function [f4,f7,f8] = NoSlipTopWallD2Q9(f1,f2,f3,f5,f6)
    f4 = f2;
    f7 = f5 + 0.5*(f1-f3);
    f8 = f6 - 0.5*(f1-f3);
end
%% SLIP BOUNDARY CONDITIONS
function [f2,f5,f6] = SlipBottomWallD2Q9(f1,f3,f4,f7,f8,f9,u,Kn,Y)
```

```
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,1)-u(:,2))/(2+3*Lamda);
    rhow  = (f1+f3+f9+2*(f4+f7+f8));
    f2    = f4;
    f5    = rhow.*(1+uslip)/2 - (f1+f8) - (f2+f4+f9)/2;
    f6    = rhow.*(1-uslip)/2 - (f3+f7) - (f2+f4+f9)/2;
end
function [f4,f7,f8] = SlipTopWallD2Q9(f1,f2,f3,f5,f6,f9,u,Kn,Y)
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,2)-u(:,1))/(2+3*Lamda);
    rhow  = (f1+f3+f9+2*(f2+f5+f6));
    f4    = f2;
    f7    = rhow.*(1-uslip)/2 - (f3+f6) - (f2+f4+f9)/2;
    f8    = rhow.*(1+uslip)/2 - (f1+f5) - (f2+f4+f9)/2;
end
%% STREAMING %%
function [f1a,f2a,f3a,f4a,f5a,f6a,f7a,f8a] = ...
    StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y)
    f1a = f1([1,1:X-1],1:Y);
    f2a = f2(1:X,[1,1:Y-1]);
    f3a = f3([2:X,X],1:Y);
    f4a = f4(1:X,[2:Y,Y]);
    f5a = f5([1,1:X-1],[1,1:Y-1]);
    f6a = f6([2:X,X],[1,1:Y-1]);
    f7a = f7([2:X,X],[2:Y,Y]);
    f8a = f8([1,1:X-1],[2:Y,Y]);
end
```

Model 3, with Porous Media, Nusselt Number, Temperature, Density, Particle Speed, and Velocity Vectors

This model was modified and introduced new parameters than Model 1 (porous media, density, particle speed, and velocity vectors) by Adam Turcotte.

```
function [] = LBM_CREATE_SCENARIO(scenario_name)
    %% PHYSICAL PARAMETERS %%
    % NOTE: I think these need to be re-named to make things work
    h_phys     = 1e-2;       % Convective heat transfer coefficient [W/m3-K]
    L_phys     = 1.2e-1;     % Characteristic length  [m]
    k_phys     = 2.57e-2;    % Thermal conductivity  [W/m-K]
    lamda_phys = 2.3787e-8;  % Mean free path  [m]
    u_phys     = 1e-2;       % Mean velocity of fluid  [m/s]
    nu_phys    = 1.57e-5;    % Kinematic viscosity  [m2/s]
    alpha_phys = 2.1606e-5;  % Thermal diffusivity  [m2/s]
    cp_phys    = 1.005;      % Specific heat  [J/kg-K]
    niu_phys   = 1.983e-5;   % Dynamic viscosity [kg/m-s]
    g_phys     = 9.8;        % Acceleration due to gravity  [m/s2]
    beta_phys  = 3.43e-3;    % Thermal expansion coefficient  [1/K]
    T1_phys    = 283;        % Hot themperature  [K]
    T0_phys    = 273;        % Cold temperature  [K]
    K_phys     = 1e-9;       % Permeability of porous media  [m2]
    %% NUSSELT NUMBER (omit this) %%
    % Ratio of convective to conductive heat transfer
    %Nu = h_phys*L_phys/k_phys;
    %% KNUDSEN NUMBER %%
    % Ratio between mean free path and physical length
    Kn = lamda_phys/L_phys;
    %% REYNOLDS NUMBER %%
    % Ratio of inertial to viscous forces
    Re = u_phys*L_phys/nu_phys;
    %% PECLET NUMBER %%
    % Ratio of advective to diffusive transport rate
    Pe = u_phys*L_phys/alpha_phys;
    %% PRANDTL NUMBER %%
    % Ratio of viscous to thermal diffusion rate
    Pr = nu_phys/alpha_phys;
    %% RAYLEIGH NUMBER %%
    % Measure of the instability of a layer of fluid due to differences of
    % temperature and density at the top and bottom
    Ra = g_phys*beta_phys*(T1_phys-T0_phys)*L_phys^3/(nu_phys*alpha_phys);
    %% DARCY NUMBER %%
    % Relative effect of the permeability of the medium to diameter squared
    Da = K_phys/L_phys^2;
    %% NUMBER OF LATTICE POINTS %%
```

```
Y = 160;
X = 400;
%% CHARACTERISTIC LENGTH AND VELOCITY %%
L        = Y-1;  % Characteristic length
Ulattice = 0.02; % Characteristic velocity
%% SIMULATION TIMESTEP %%
timestep = 0;
%% SLIP BOUNDARY VALUES %%
Kappa  = 0;            % Temperature jump parameter
Lambda = Kn*L;         % Mean free path
C      = Kappa*Lambda; % Temperature jump coefficient
%% FLUID PROPERTIES %%
Viscosity = Ulattice*L/Re; % Kinematic viscosity
alpha     = Viscosity/Pr;  % Thermal diffusivity
%% POROUS MEDIUM %%
eps  = 0.440;                % Porosity
Feps = 1.75/sqrt(150*eps^3); % Geometric function
K    = Da*L^2;
%% TEMPERATURE %%
T_hot    = 1.0;               % Hot temperature (normalized)
T_cold   = 0.0;               % Cold temperature (normalized)
T_inlet  = T_hot;             % Inlet temperature
T_top    = T_cold;            % Top wall temperature
T_bottom = T_cold;            % Bottom wall temperature
%T_top    = T_hot;
%T_bottom = T_hot;
T_init   = T_cold;            % Initial temperature
T_diff   = T_hot-T_cold;      % Temperature difference
T0       = 0.5*(T_hot+T_cold); % Reference temperature
%% BODY FORCE %%
gbeta    = Ra*Viscosity*alpha/(T_diff*L^3);
buoyancy = [0,gbeta]; % Buoyancy force
a        = [0,0];     % Acceleration due to external force fields
%% RELAXATION PARAMETERS %%
omega             = 1 / (3*Viscosity + 0.5);
omegat            = 1 / (3*alpha     + 0.5);
oneMinusOmega     = 1 - omega;
oneMinusHalfOmega = 1 - 0.5*omega;
oneMinusOmegat    = 1 - omegat;
%% OTHER CONSTANTS %%
c0 = 0.5*(1+0.5*eps*Viscosity/K);
c1 = 0.5*eps*Feps/sqrt(K);
d1 = -eps*Viscosity/K;
d2 = eps*Feps/sqrt(K);
%% INITIAL CONDITIONS %%
rho = ones(X,Y);        % Fluid density
Fx  = zeros(X,Y);       % Total body force (horizontal component)
Fy  = zeros(X,Y);       % Total body force (vertical component)
f1  = zeros(X,Y);       % Density distribution function
f2  = zeros(X,Y);       % Density distribution function
f3  = zeros(X,Y);       % Density distribution function
f4  = zeros(X,Y);       % Density distribution function
f5  = zeros(X,Y);       % Density distribution function
f6  = zeros(X,Y);       % Density distribution function
f7  = zeros(X,Y);       % Density distribution function
f8  = zeros(X,Y);       % Density distribution function
f9  = zeros(X,Y);       % Density distribution function
```

```
    u   = zeros(X,Y);       % Fluid velocity (horizontal component)
    v   = zeros(X,Y);       % Fluid velocity (vertical component)
    T   = T_init*ones(X,Y); % Fluid temperature
    g1  = zeros(X,Y);       % Temperature distribution function
    g2  = zeros(X,Y);       % Temperature distribution function
    g3  = zeros(X,Y);       % Temperature distribution function
    g4  = zeros(X,Y);       % Temperature distribution function
    g5  = zeros(X,Y);       % Temperature distribution function
    g6  = zeros(X,Y);       % Temperature distribution function
    g7  = zeros(X,Y);       % Temperature distribution function
    g8  = zeros(X,Y);       % Temperature distribution function
    g9  = zeros(X,Y);       % Temperature distribution function
    Nu  = zeros(X,1);       % Nusselt number
    %% SAVE THE SCENARIO FILE %%
    vars = struct('Kn', Kn, 'Re', Re, 'Pe', Pe, 'Pr', Pr, 'Ra', Ra, ...
        'Da', Da, 'Y', Y, 'X', X, 'L', L, 'Ulattice', Ulattice, ...
        'timestep', timestep, 'Kappa', Kappa, 'Lambda', Lambda, ...
        'C', C, 'Viscosity', Viscosity, 'alpha', alpha, 'eps', eps, ...
        'Feps', Feps, 'K', K, 'T_hot', T_hot, 'T_cold', T_cold, ...
        'T_inlet', T_inlet, 'T_top', T_top, 'T_bottom', T_bottom, ...
        'T_init', T_init, 'T_diff', T_diff, 'T0', T0, 'gbeta', gbeta, ...
        'buoyancy', buoyancy, 'a', a, 'omega', omega, 'omegat', omegat, ...
        'oneMinusOmega', oneMinusOmega, ...
        'oneMinusHalfOmega', oneMinusHalfOmega, ...
        'oneMinusOmegat', oneMinusOmegat, 'c0', c0, 'c1', c1, 'd1', d1, ...
        'd2', d2, 'rho', rho, 'Fx', Fx, 'Fy', Fy, 'f1', f1, 'f2', f2, ...
        'f3', f3, 'f4', f4, 'f5', f5, 'f6', f6, 'f7', f7, 'f8', f8, ...
        'f9', f9, 'u', u, 'v', v, 'T', T, 'g1', g1, 'g2', g2, 'g3', g3, ...
        'g4', g4, 'g5', g5, 'g6', g6, 'g7', g7, 'g8', g8, 'g9', g9, ...
        'Nu', Nu);
    save([scenario_name '.mat'], 'vars');
end



function [] = LBM_PLOT_SCENARIO(scenario_name)
    %% LOAD THE SCENARIO FILE %%
    load([scenario_name '.mat'], 'vars');
    Y = vars.Y;
    X = vars.X;
    rho = vars.rho;
    u = vars.u;
    v = vars.v;
    T = vars.T;
    Nu = vars.Nu;
    timestep = vars.timestep;
    %% SET UP VARIABLES %%
    THERMAL_MAP = [           0.1176,0.1412,0.2275; 0.1451,0.1314,0.2735;
        0.1725,0.1216,0.3196; 0.2000,0.1118,0.3657; 0.2275,0.1020,0.4118;
        0.2549,0.0922,0.4578; 0.2824,0.0824,0.5039; 0.3098,0.0725,0.5500;
        0.3373,0.0627,0.5961; 0.3637,0.0711,0.5875; 0.3902,0.0794,0.5789;
        0.4167,0.0877,0.5703; 0.4431,0.0961,0.5618; 0.4696,0.1044,0.5532;
        0.4961,0.1127,0.5446; 0.5225,0.1211,0.5360; 0.5490,0.1294,0.5275;
        0.5755,0.1377,0.5189; 0.6020,0.1461,0.5103; 0.6284,0.1544,0.5017;
        0.6549,0.1627,0.4931; 0.6814,0.1711,0.4846; 0.7078,0.1794,0.4760;
        0.7343,0.1877,0.4674; 0.7608,0.1961,0.4588; 0.7696,0.2167,0.4336;
```

```
    0.7784,0.2373,0.4083; 0.7873,0.2578,0.3831; 0.7961,0.2784,0.3578;
    0.8049,0.2990,0.3326; 0.8137,0.3196,0.3074; 0.8225,0.3402,0.2821;
    0.8314,0.3608,0.2569; 0.8402,0.3814,0.2316; 0.8490,0.4020,0.2064;
    0.8578,0.4225,0.1811; 0.8667,0.4431,0.1559; 0.8755,0.4637,0.1306;
    0.8843,0.4843,0.1054; 0.8931,0.5049,0.0801; 0.9020,0.5255,0.0549;
    0.8985,0.5434,0.0703; 0.8951,0.5613,0.0858; 0.8917,0.5792,0.1012;
    0.8882,0.5971,0.1167; 0.8848,0.6150,0.1321; 0.8814,0.6328,0.1475;
    0.8779,0.6507,0.1630; 0.8745,0.6686,0.1784; 0.8711,0.6865,0.1939;
    0.8676,0.7044,0.2093; 0.8642,0.7223,0.2248; 0.8608,0.7402,0.2402;
    0.8574,0.7581,0.2556; 0.8539,0.7760,0.2711; 0.8505,0.7939,0.2865;
    0.8471,0.8118,0.3020; 0.8504,0.8230,0.3782; 0.8538,0.8342,0.4543;
    0.8571,0.8454,0.5305; 0.8605,0.8566,0.6067; 0.8639,0.8678,0.6829;
    0.8672,0.8790,0.7591; 0.8706,0.8902,0.8353];
Umean = zeros(X,1);
dy = 1;
ut = zeros(X,Y);
T_mean = zeros(X,1);
A = zeros(X,1);
H = Y-1;
%% COMPUTE NUSSELT NUMBER %%
for i = 1:X
    dummy = 0;
    for j = 1:5:Y-5
        dummy = dummy+(5*dy)*(19*rho(i,j)+75*rho(i,j+1)+...
            50*rho(i,j+2)+50*rho(i,j+3)+75*rho(i,j+4)+...
            19*rho(i,j+5))/288;
    end
    Rhomean(i) = dummy/H;
    dummy = 0;
    for j = 1:5:Y-5
        dummy = dummy+(5*dy)*(19*rho(i,j)*u(i,j)+...
            75*rho(i,j)*u(i,j+1)+50*rho(i,j)*u(i,j+2)+...
            50*rho(i,j)*u(i,j+3)+75*rho(i,j)*u(i,j+4)+...
            19*rho(i,j)*u(i,j+5))/288;
    end
    Umean(i) = dummy/Rhomean(i)/H;
    dummy = 0;
    ut(i,:) = rho(i,:).*u(i,:).*T(i,:);
    for j = 1:5:Y-5
        dummy = dummy+5*dy*(19*ut(i,j)+75*ut(i,j+1)+50*ut(i,j+2)+...
            50*ut(i,j+3)+75*ut(i,j+4)+19*ut(i,j+5))/288;
    end
    T_mean(i) = (dummy/Rhomean(i)/Umean(i)/H);
    A(i) = 49/20*T(i,1)-6*T(i,2)+15/2*T(i,3)-20/3*T(i,4)+...
        15/4*T(i,5)-6/5*T(i,6)+1/6*T(i,7); % derivative
    Nu(i) = -A(i)/T_mean(i)*2*H;
end
%% PLOT RESULTS %%
fig = figure('Visible','off');
subplot(4,2,1)
plot(Nu)
title('Nusselt Number')

subplot(4,2,2)
imagesc(T(:,Y:-1:1)')
colormap(THERMAL_MAP);
c1 = colorbar;
```

```
    freezeColors
    cbfreeze(c1);
    title(['Temperature (Time Step: ' num2str(timestep) ')'])

    subplot(4,2,3)
    imagesc(rho(:,Y:-1:1)')
    colormap('default');
    c2 = colorbar;
    freezeColors
    cbfreeze(c2);
    title('Density');

    subplot(4,2,4)
    abs_vel = sqrt(u.^2+v.^2);
    imagesc(abs_vel(:,Y:-1:1)')
    colormap('default');
    c3 = colorbar;
    freezeColors
    cbfreeze(c3);
    title('Particle Speed');

    subplot(4,2,[5 6 7 8])
    NN=20;
    MM=10;
    uuu=u(1:NN:X,Y:-MM:1);
    vvv=-1.0*v(1:NN:X,Y:-MM:1);
    [y,x] = meshgrid(1:MM:Y,1:NN:X);
    quiver(x,y,uuu,vvv)
    set(gca,'YDir','reverse');
    title('Velocity Vectors');

    %drawnow
    set(gcf,'PaperUnits','inches','PaperPosition',[0 0 16 9])
    print(fig, '-dpng', '-r300', [scenario_name '.png'])
end
```

```
function [] = LBM_RUN_SCENARIO
(input_name, output_name, max_timestep)
    %% SET UP THE SIMULATION STATE %%
    load([input_name '.mat'], 'vars');
    Kn = vars.Kn;
    Re = vars.Re;
    Pe = vars.Pe;
    Pr = vars.Pr;
    Ra = vars.Ra;
    Da = vars.Da;
    Y = vars.Y;
    X = vars.X;
    L = vars.L;
    Ulattice = vars.Ulattice;
```

```
Kappa = vars.Kappa;
Lambda = vars.Lambda;
C = vars.C;
Viscosity = vars.Viscosity;
alpha = vars.alpha;
eps = vars.eps;
Feps = vars.Feps;
K = vars.K;
T_hot = vars.T_hot;
T_cold = vars.T_cold;
T_inlet = vars.T_inlet;
T_top = vars.T_top;
T_bottom = vars.T_bottom;
T_init = vars.T_init;
T_diff = vars.T_diff;
T0 = vars.T0;
gbeta = vars.gbeta;
buoyancy = vars.buoyancy;
a = vars.a;
omega = vars.omega;
omegat = vars.omegat;
oneMinusOmega = vars.oneMinusOmega;
oneMinusHalfOmega = vars.oneMinusHalfOmega;
oneMinusOmegat = vars.oneMinusOmegat;
c0 = vars.c0;
c1 = vars.c1;
d1 = vars.d1;
d2 = vars.d2;
rho = vars.rho;
Fx = vars.Fx;
Fy = vars.Fy;
f1 = vars.f1;
f2 = vars.f2;
f3 = vars.f3;
f4 = vars.f4;
f5 = vars.f5;
f6 = vars.f6;
f7 = vars.f7;
f8 = vars.f8;
f9 = vars.f9;
u = vars.u;
v = vars.v;
T = vars.T;
g1 = vars.g1;
g2 = vars.g2;
g3 = vars.g3;
g4 = vars.g4;
g5 = vars.g5;
g6 = vars.g6;
g7 = vars.g7;
g8 = vars.g8;
g9 = vars.g9;
Nu = vars.Nu;
prev_timestep = vars.timestep;
clear vars;
%% RUN THE SIMULATION %%
for timestep = prev_timestep+1:max_timestep
```

```
%% MOMENTUM TRANSFER %%
% COLLISION
[f1,f2,f3,f4,f5,f6,f7,f8,f9] = ...
    CollisionD2Q9(u,v,rho,Fx,Fy,eps,f1,f2,f3,f4,f5,f6,f7,f8,f9,...
                  omega,oneMinusOmega,oneMinusHalfOmega);
% STREAMING
[f1,f2,f3,f4,f5,f6,f7,f8] = ...
    StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y);
% BOUNDARY CONDITIONS
[f1(1,:),f2(1,:),f4(1,:),f5(1,:),f6(1,:),f7(1,:),f8(1,:)] = ...
    InletconstantVelocityD2Q9(f2(1,:),f3(1,:),f4(1,:),f6(1,:),...
                              f7(1,:),f9(1,:),Ulattice,Y);
[f3(X,:),f6(X,:),f7(X,:)] = ...
    OutletConstantVelocity(f1(X,:),f2(X,:),f4(X,:),f5(X,:),...
                           f8(X,:),f9(X,:),u(X-10,:));
%[f2(:,1),f5(:,1),f6(:,1)] = ...
%    SlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1),...
%                       f9(:,1),u(:,2:3),Kn,Y);
%[f4(:,Y),f7(:,Y),f8(:,Y)] = ...
%    SlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y),...
%                    f9(:,Y),u(:,Y-2:Y-1),Kn,Y);
[f2(:,1),f5(:,1),f6(:,1)] = ...
    NoSlipBottomWallD2Q9(f1(:,1),f3(:,1),f4(:,1),f7(:,1),f8(:,1));
[f4(:,Y),f7(:,Y),f8(:,Y)] = ...
    NoSlipTopWallD2Q9(f1(:,Y),f2(:,Y),f3(:,Y),f5(:,Y),f6(:,Y));
% MACROSCOPIC PROPERTY CALCULATION
[u,v,rho,Fx,Fy] = ...
    MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y,...
                    eps,buoyancy,a,T_diff,c0,c1,d1,d2,T,T0);
%% HEAT TRANSFER %%
% COLLISION
[g1,g2,g3,g4,g5,g6,g7,g8,g9] = ...
    CollisionTD2Q9(u,v,T,g1,g2,g3,g4,g5,g6,g7,g8,g9,...
                   omegat,oneMinusOmegat);
% STREAMING
[g1,g2,g3,g4,g5,g6,g7,g8] = ...
    StreamingD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,X,Y);
% BOUNDARY CONDITIONS
% INLET
g1(1,2:Y-1) = T_inlet*2/9  - g3(1,2:Y-1);
g5(1,2:Y-1) = T_inlet/18 - g7(1,2:Y-1);
g8(1,2:Y-1) = T_inlet/18 - g6(1,2:Y-1);
% OUTLET (ZERO-FLUX; EXTRAPOLATION)
g3(X,2:Y-1) = 2*g3(X-1,2:Y-1) - g3(X-2,2:Y-1);
g6(X,2:Y-1) = 2*g6(X-1,2:Y-1) - g6(X-2,2:Y-1);
g7(X,2:Y-1) = 2*g7(X-1,2:Y-1) - g7(X-2,2:Y-1);
% TOP WALL
%[g4(:,Y),g7(:,Y),g8(:,Y)] = ...
%    TJumpTop(g2(:,Y),g5(:,Y),g6(:,Y),T(:,[Y-2:Y-1]),T_top,C);
[g4(:,Y),g7(:,Y),g8(:,Y)] = ...
    TNoSlipTop(g2(:,Y),g5(:,Y),g6(:,Y),T_top);
% BOTTOM WALL
%[g2(:,1),g5(:,1),g6(:,1)] = ...
%    TJumpBottom(g4(:,1),g7(:,1),g8(:,1),T(:,2:3),T_bottom,C);
[g2(:,1),g5(:,1),g6(:,1)] = ...
    TNoSlipBottom(g4(:,1),g7(:,1),g8(:,1),T_bottom);
% MACROSCOPIC PROPERTY CALCULATION
```

```
        T = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9);
    end
    %% SAVE THE NEW SIMULATION STATE %%
    vars = struct('Kn', Kn, 'Re', Re, 'Pe', Pe, 'Pr', Pr, 'Ra', Ra, ...
        'Da', Da, 'Y', Y, 'X', X, 'L', L, 'Ulattice', Ulattice, ...
        'timestep', timestep, 'Kappa', Kappa, 'Lambda', Lambda, ...
        'C', C, 'Viscosity', Viscosity, 'alpha', alpha, 'eps', eps, ...
        'Feps', Feps, 'K', K, 'T_hot', T_hot, 'T_cold', T_cold, ...
        'T_inlet', T_inlet, 'T_top', T_top, 'T_bottom', T_bottom, ...
        'T_init', T_init, 'T_diff', T_diff, 'T0', T0, 'gbeta', gbeta, ...
        'buoyancy', buoyancy, 'a', a, 'omega', omega, 'omegat', omegat, ...
        'oneMinusOmega', oneMinusOmega, ...
        'oneMinusHalfOmega', oneMinusHalfOmega, ...
        'oneMinusOmegat', oneMinusOmegat, 'c0', c0, 'c1', c1, 'd1', d1, ...
        'd2', d2, 'rho', rho, 'Fx', Fx, 'Fy', Fy, 'f1', f1, 'f2', f2, ...
        'f3', f3, 'f4', f4, 'f5', f5, 'f6', f6, 'f7', f7, 'f8', f8, ...
        'f9', f9, 'u', u, 'v', v, 'T', T, 'g1', g1, 'g2', g2, 'g3', g3, ...
        'g4', g4, 'g5', g5, 'g6', g6, 'g7', g7, 'g8', g8, 'g9', g9, ...
        'Nu', Nu);
    save([output_name '.mat'], 'vars');
end

%% TEMPERATURE BOUNDARY CONDITIONS %%
function [g4,g7,g8] = TNoSlipTop(g2,g5,g6,Tw)
    g8 = Tw/18 - g6;
    g7 = Tw/18 - g5;
    g4 = 2*Tw/9 - g2;
end
function [g2,g5,g6] = TNoSlipBottom(g4,g7,g8,Tw)
    g6 = Tw/18 - g8;
    g5 = Tw/18 - g7;
    g2 = 2*Tw/9 - g4;
end
%% TEMPERATURE JUMP BOUNDARY CONDITIONS
function [g4,g7,g8] = TJumpTop(g2,g5,g6,T,Tw,C)
    tw = (C*(4*T(:,2)-T(:,1))+2*Tw)/(2+3*C);
    g8 = tw/18 - g6;
    g7 = tw/18 - g5;
    g4 = 2*tw/9 - g2;
end
function [g2,g5,g6] = TJumpBottom(g4,g7,g8,T,Tw,C)
    tw = (C*(4*T(:,1)-T(:,2))+2*Tw)/(2+3*C);
    g6 = tw/18 - g8;
    g5 = tw/18 - g7;
    g2 = 2*tw/9 - g4;
end
%% COLLISION FOR MOMENTUM TRANSFER %%
function [f1,f2,f3,f4,f5,f6,f7,f8,f9] = ...
    CollisionD2Q9(u,v,rho,Fx,Fy,eps,f1,f2,f3,f4,f5,f6,f7,f8,f9,...
                  omega,oneMinusOmega,oneMinusHalfOmega)
    t10    = u.*u + v.*v;
    t10    = -1.5/eps*t10;
    t1     =  u;
    t2     =  v;
    t3     = -u;
    t4     = -v;
    t5     =  u + v;
```

```
    t6     = -u + v;
    t7     = -u - v;
    t8     =  u - v;
    a10    = u.*Fx + v.*Fy;
    a10    = -3.0/eps*a10;
    a1     =  Fx;
    a2     =  Fy;
    a3     = -Fx;
    a4     = -Fy;
    a5     =  Fx + Fy;
    a6     = -Fx + Fy;
    a7     = -Fx - Fy;
    a8     =  Fx - Fy;
    a11    = u.*Fx;
    a12    = v.*Fy;
    a13    = a11;
    a14    = a12;
    a15    = u.*Fx + v.*Fx + u.*Fy + v.*Fy;
    a16    = u.*Fx - v.*Fx - u.*Fy + v.*Fy;
    a17    = a15;
    a18    = a16;
    feq1   = rho/9   .* (1 + 3*t1 + (4.5/eps)*t1.*t1 + t10);
    feq2   = rho/9   .* (1 + 3*t2 + (4.5/eps)*t2.*t2 + t10);
    feq3   = rho/9   .* (1 + 3*t3 + (4.5/eps)*t3.*t3 + t10);
    feq4   = rho/9   .* (1 + 3*t4 + (4.5/eps)*t4.*t4 + t10);
    feq5   = rho/36  .* (1 + 3*t5 + (4.5/eps)*t5.*t5 + t10);
    feq6   = rho/36  .* (1 + 3*t6 + (4.5/eps)*t6.*t6 + t10);
    feq7   = rho/36  .* (1 + 3*t7 + (4.5/eps)*t7.*t7 + t10);
    feq8   = rho/36  .* (1 + 3*t8 + (4.5/eps)*t8.*t8 + t10);
    feq9   = 4*rho/9 .* (1 + t10);
    force1 = oneMinusHalfOmega*rho/9   .* (3*a1 + (9/eps)*a11 + a10);
    force2 = oneMinusHalfOmega*rho/9   .* (3*a2 + (9/eps)*a12 + a10);
    force3 = oneMinusHalfOmega*rho/9   .* (3*a3 + (9/eps)*a13 + a10);
    force4 = oneMinusHalfOmega*rho/9   .* (3*a4 + (9/eps)*a14 + a10);
    force5 = oneMinusHalfOmega*rho/36  .* (3*a5 + (9/eps)*a15 + a10);
    force6 = oneMinusHalfOmega*rho/36  .* (3*a6 + (9/eps)*a16 + a10);
    force7 = oneMinusHalfOmega*rho/36  .* (3*a7 + (9/eps)*a17 + a10);
    force8 = oneMinusHalfOmega*rho/36  .* (3*a8 + (9/eps)*a18 + a10);
    force9 = 4*oneMinusHalfOmega*rho/9 .* a10;
    f1     = omega*feq1 + oneMinusOmega*f1 + force1;
    f2     = omega*feq2 + oneMinusOmega*f2 + force2;
    f3     = omega*feq3 + oneMinusOmega*f3 + force3;
    f4     = omega*feq4 + oneMinusOmega*f4 + force4;
    f5     = omega*feq5 + oneMinusOmega*f5 + force5;
    f6     = omega*feq6 + oneMinusOmega*f6 + force6;
    f7     = omega*feq7 + oneMinusOmega*f7 + force7;
    f8     = omega*feq8 + oneMinusOmega*f8 + force8;
    f9     = omega*feq9 + oneMinusOmega*f9 + force9;
end
%% COLLISION FOR HEAT TRANSFER %%
function [g1,g2,g3,g4,g5,g6,g7,g8,g9] = ...`
    CollisionTD2Q9(u,v,th,g1,g2,g3,g4,g5,g6,g7,g8,g9,omegat,oneMinusOmegat)
    t1 =  u;
    t2 =  v;
    t3 = -u;
    t4 = -v;
    t5 =  u + v;
```

```
    t6 = -u + v;
    t7 = -u - v;
    t8 =  u - v;
    feq1 = th/9  .* (1 + 3*t1);
    feq2 = th/9  .* (1 + 3*t2);
    feq3 = th/9  .* (1 + 3*t3);
    feq4 = th/9  .* (1 + 3*t4);
    feq5 = th/36 .* (1 + 3*t5);
    feq6 = th/36 .* (1 + 3*t6);
    feq7 = th/36 .* (1 + 3*t7);
    feq8 = th/36 .* (1 + 3*t8);
    feq9 = 4*th/9;
    g1 = omegat*feq1 + oneMinusOmegat*g1;
    g2 = omegat*feq2 + oneMinusOmegat*g2;
    g3 = omegat*feq3 + oneMinusOmegat*g3;
    g4 = omegat*feq4 + oneMinusOmegat*g4;
    g5 = omegat*feq5 + oneMinusOmegat*g5;
    g6 = omegat*feq6 + oneMinusOmegat*g6;
    g7 = omegat*feq7 + oneMinusOmegat*g7;
    g8 = omegat*feq8 + oneMinusOmegat*g8;
    g9 = omegat*feq9 + oneMinusOmegat*g9;
end
%% INLET BOUNDARY CONDITIONS %%
function [f1,f2,f4,f5,f6,f7,f8] = ...
    InletconstantVelocityD2Q9(f2,f3,f4,f6,f7,f9,Ulattice,Y)
    rhow = (f9+f2+f4 + 2*(f3+f6+f7))/(1-Ulattice);
    f1  = f3 + 2*rhow*Ulattice/3;
    f5  = f7 + 0.5*(f4-f2) + rhow*Ulattice/6;
    f8  = f6 + 0.5*(f2-f4) + rhow*Ulattice/6;
end
%% MACROSCOPIC PROPERTY CALCULATION FOR MOMENTUM TRANSFER %%
function [u,v,rho,Fx,Fy] = ...
    MacroscopicD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,f9,X,Y,...
                    eps,buoyancy,a,T_diff,c0,c1,d1,d2,T,T0)
    rho     = f1+f2+f3+f4+f5+f6+f7+f8+f9;
    usum    = f1-f3+f5-f6-f7+f8;
    vsum    = f2-f4+f5+f6-f7-f8;
    %Gx      = buoyancy(1)*T_diff+a(1);  % MIGHT HAVE TO COMPUTE A SPECIAL
    %Gy      = buoyancy(2)*T_diff+a(2);  % T_diff ON EACH ITERATION
    dT      = T-T0;
    Gx      = buoyancy(1)*dT+a(1);
    Gy      = buoyancy(2)*dT+a(2);
    aux_u   = usum./rho + 0.5*eps*Gx;
    aux_v   = vsum./rho + 0.5*eps*Gy;
    aux_den = c0+sqrt(c0*c0+c1*sqrt(aux_u.*aux_u+aux_v.*aux_v));
    u       = aux_u./aux_den;
    v       = aux_v./aux_den;
    uv_mod  = sqrt(u.*u+v.*v);
    Fx      = d1*u-d2*uv_mod.*u+eps*Gx;
    Fy      = d1*v-d2*uv_mod.*v+eps*Gy;
end
%% MACROSCOPIC PROPERTY CALCULATION FOR HEAT TRANSFER %%
function th = MacroscopicTD2Q9(g1,g2,g3,g4,g5,g6,g7,g8,g9)
    th = g1+g2+g3+g4+g5+g6+g7+g8+g9;
end
%% OUTLET BOUNDARY CONDITIONS %%
function [f3,f6,f7] = OutletConstantVelocity(f1,f2,f4,f5,f8,f9,u_out)
```

```
    rho_o = (f9+f2+f4+2*(f1+f5+f8))./(1.0+u_out);
    f3    = f1 - 0.667*rho_o.*u_out;
    f7    = f5 + 0.5*(f2-f4) - rho_o.*u_out/6.0;
    f6    = f8 + 0.5*(f4-f2) - rho_o.*u_out/6.0;
end
%% NO-SLIP BOUNDARY CONDITIONS %%
function [f2,f5,f6] = NoSlipBottomWallD2Q9(f1,f3,f4,f7,f8)
    f2 = f4;
    f5 = f7 - 0.5*(f1-f3);
    f6 = f8 + 0.5*(f1-f3);
end
function [f4,f7,f8] = NoSlipTopWallD2Q9(f1,f2,f3,f5,f6)
    f4 = f2;
    f7 = f5 + 0.5*(f1-f3);
    f8 = f6 - 0.5*(f1-f3);
end
%% SLIP BOUNDARY CONDITIONS
function [f2,f5,f6] = SlipBottomWallD2Q9(f1,f3,f4,f7,f8,f9,u,Kn,Y)
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,1)-u(:,2))/(2+3*Lamda);
    rhow  = (f1+f3+f9+2*(f4+f7+f8));
    f2    = f4;
    f5    = rhow.*(1+uslip)/2 - (f1+f8) - (f2+f4+f9)/2;
    f6    = rhow.*(1-uslip)/2 - (f3+f7) - (f2+f4+f9)/2;
end
function [f4,f7,f8] = SlipTopWallD2Q9(f1,f2,f3,f5,f6,f9,u,Kn,Y)
    Lamda = Kn*(Y-1);
    uslip = Lamda*(4*u(:,2)-u(:,1))/(2+3*Lamda);
    rhow  = (f1+f3+f9+2*(f2+f5+f6));
    f4    = f2;
    f7    = rhow.*(1-uslip)/2 - (f3+f6) - (f2+f4+f9)/2;
    f8    = rhow.*(1+uslip)/2 - (f1+f5) - (f2+f4+f9)/2;
end
%% STREAMING %%
function [f1a,f2a,f3a,f4a,f5a,f6a,f7a,f8a] = ...
    StreamingD2Q9(f1,f2,f3,f4,f5,f6,f7,f8,X,Y)
    f1a = f1([1,1:X-1],1:Y);
    f2a = f2(1:X,[1,1:Y-1]);
    f3a = f3([2:X,X],1:Y);
    f4a = f4(1:X,[2:Y,Y]);
    f5a = f5([1,1:X-1],[1,1:Y-1]);
    f6a = f6([2:X,X],[1,1:Y-1]);
    f7a = f7([2:X,X],[2:Y,Y]);
    f8a = f8([1,1:X-1],[2:Y,Y]);
end
```

Appendix 2

LBM Worked Example for D2Q9 Model.

In this section, a LBM worked example for the D2Q9 model is presented. This example demonstrates detailed interactions between values that feature in the LBM using 'long hand' numerical calculation for a 4x3 lattice of 'fluid sites' (Fig.2-1). The calculations are based on equations presented in Chapter 3 and Chapter 4 and for simplicity, reflect flow in an open duct, rather than flow in a porous medium, which are also detailed in Chapter 3. The results of each step of the LBM calculation cycle are shown in Tables A2-2 to A2-11.

Fig. A2-1 setup of 4 node by 3 node lattice model for illustrative flow in a uniform duct.

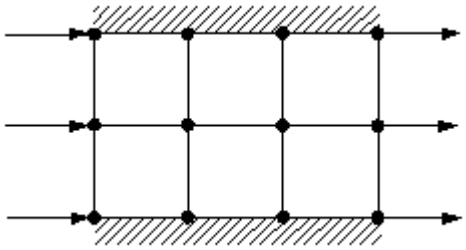


Fig. A2-1 Lattice model

The steps for calculations are:

1. Real World Values of Dimensionless Numbers for Problem

The dimensionless numbers used are in Table 6-1.

2. Mapping from Real World to Non-dimensional Lattice World

The value in lattice world are in the Table A2-1:

Table A2-1. Lattice value

Name	c_s^2	ρ	**u**	ν	τ	M	N
Value	1/3	10	0.1	4.29×10^{-4}	0.5001	4	3

3. Initialization

The value for $c_s^2 = 1/3$, thus $c^2 = 1$ (see Chapter 3). The equilibrium distribution functions of all nodes were calculated using the following equation (Eq. 3.58) where $c^2 = 1$:

$$f_a^{(eq)} = w_a \rho \left[1 + \frac{3\mathbf{e}_a \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{e}_a \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] \qquad a = 0,1,...8$$

The weight factors for D2Q9 model are: $w_0 = 4/9$ for the rest particle, $w_a = 1/9$ ($a = 1,2,3,4$) for particles streaming to the face-connected neighbours, and $w_a = 1/36$ ($a = 5,6,7,8$) for particles streaming to the edge-connected neighbours. The speed of sound is related to the lattice velocity and is $c_s = c/\sqrt{3}$. The relaxation time is given by the kinematic viscosity $\nu = c_s^2(\tau - 1/2)$.

$f_a^{eq} = f_a^{eq}(\mathbf{x}, t = 0)$

$\rho = \rho(\mathbf{x}, t = 0) = 10$

$\mathbf{u} = \mathbf{u}(\mathbf{x}, t = 0) = 0.1$

The non-equilibrium distribution function, $f_a(\mathbf{x},t=0)$ was initially set to be equal to equilibrium distribution function $f_a^{eq}(\mathbf{x},t=0)$.

The equilibrium distribution functions at each node were calculated:

$$f_0^{eq} = \rho \cdot \frac{4}{9}[1-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_1^{eq} = \rho \cdot \frac{1}{9}[1+3u_x+\frac{9}{2}u_x^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_2^{eq} = \rho \cdot \frac{1}{9}[1+3u_y+\frac{9}{2}u_y^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_3^{eq} = \rho \cdot \frac{1}{9}[1-3u_x+\frac{9}{2}u_x^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_4^{eq} = \rho \cdot \frac{1}{9}[1-3u_y+\frac{9}{2}u_y^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_5^{eq} = \rho \cdot \frac{1}{36}[1+3(u_x+u_y)+\frac{9}{2}(u_x+u_y)^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_6^{eq} = \rho \cdot \frac{1}{36}[1+3(-u_x+u_y)+\frac{9}{2}(-u_x+u_y)^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_7^{eq} = \rho \cdot \frac{1}{36}[1+3(-u_x-u_y)+\frac{9}{2}(u_x+u_y)^2-\frac{3}{2}(u_x^2+u_y^2)]$$

$$f_8^{eq} = \rho \cdot \frac{1}{36}[1+3(u_x-u_y)+\frac{9}{2}(u_x-u_y)^2-\frac{3}{2}(u_x^2+u_y^2)]$$

4. Collision

The Lattice Boltzmann equation with a single relaxation time can be written as follows (Eq. 3.42), where $\delta_t = 1$:

$$f_a(\mathbf{x}+\mathbf{e}_a,t+1)-f_a(\mathbf{x},t) = -\frac{1}{\tau}\left[f_a(\mathbf{x},t)-f_a^{eq}(\mathbf{x},t)\right]$$

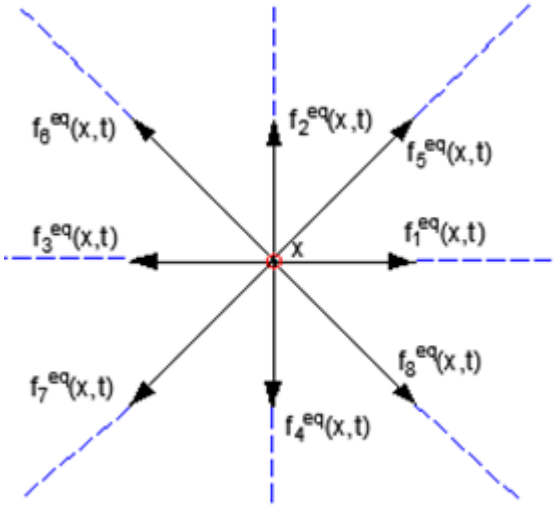
Where the second term is for collision and first term is for streaming.

5. Stream

Figure A2-2 shows the orientation of the equilibrium distribution function components at time t ($f_a^{eq}(\mathbf{x},t)$). These components at each node propagate to the neighbouring nodes and produce the non-equilibrium distribution function for the next time step of the neighbouring node $f_a(\mathbf{x}+\mathbf{e}_a,t+1)$. The propagation was done using the following equation:

$$f_a(\mathbf{x}+\mathbf{e}_a,t+1) = f_a(\mathbf{x},t)$$

Where $f_a(\mathbf{x},t)$ and $f_a(\mathbf{x}+\mathbf{e}_a,t+1)$ are the non-equilibrium distribution function in lattice direction a at node $\mathbf{x}$, at time t, at node $\mathbf{x}+\mathbf{e}_a$ at time $t+1$.



(a)

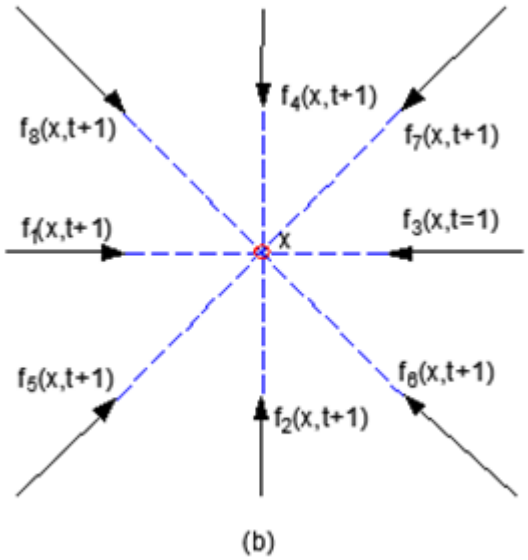


Fig. A2-2 (a) Orientation of components of equilibrium distribution function calculated in previous time step; (b) Presentation of non-equilibrium distribution function

6. Inlet Boundary

At the inlet, the known components are: $u_x, u_y, f_0, f_2, f_4, f_3, f_6, f_7$, and the unknown components are: ρ, f_1, f_8, f_5. The unknown components can be calculated from known components(Eqs. 4-24 to 4-27):

$$\rho_{in} = \frac{1}{1-u_{x(in)}}\left[f_0 + f_2 + f_4 + 2\left(f_3 + f_6 + f_7\right)\right]$$

$$f_1 = f_3 + \frac{2}{3}\rho u_x$$

$$f_8 = f_6 + \frac{1}{2}\left(f_2 - f_4\right) + \frac{1}{6}\rho u_x$$

$$f_5 = f_7 - \frac{1}{2}\left(f_2 - f_4\right) + \frac{1}{6}\rho u_x$$

7. Outlet Boundary

At the outlet, the known components are: $u_x, u_y, f_0, f_2, f_4, f_1, f_5, f_8$, and the unknown components are: ρ, f_3, f_6, f_7. The unknown components can be calculated from known components (Eqs. 4-46 to 4-49):

$$\rho_{out} = \frac{1}{1+u_{x(out)}}\left[f_0 + f_2 + f_4 + 2\left(f_1 + f_5 + f_8\right)\right]$$

$$f_3 = f_1 + \frac{2}{3}\rho u_x$$

$$f_6 = f_8 - \frac{1}{2}\left(f_2 - f_4\right) - \frac{1}{6}\rho u_x$$

$$f_7 = f_5 + \frac{1}{2}\left(f_2 - f_4\right) - \frac{1}{6}\rho u_x$$

8. No Slip Bottom Boundary

At the bottom, the known components are: $u_x, u_y, f_0, f_1, f_3, f_4, f_7, f_8$, and the unknown components are: ρ, f_2, f_6, f_5. The unknown components can be calculated from known components (Eqs. 4-32 to 4-34):

$$\rho_{bottom} = \frac{1}{1-u_{y(bottom)}}\left[f_0 + f_1 + f_3 + 2\left(f_4 + f_7 + f_8\right)\right]$$

$$f_2 = f_4$$

$$f_6 = f_8 + \frac{1}{2}\left(f_1 - f_3\right)$$

$$f_5 = f_7 - \frac{1}{2}\left(f_1 - f_3\right)$$

9. No Slip Top |Boundary

At the top, the known components are: $u_x, u_y, f_0, f_1, f_3, f_2, f_6, f_5$, and the unknown components are: ρ, f_4, f_7, f_8. The unknown components can be calculated from known components (Eqs. 4-39 to 4-41):

$$\rho_{top} = \frac{1}{1+u_{y(top)}}\left[f_0 + f_1 + f_3 + 2\left(f_2 + f_6 + f_5\right)\right]$$

$$f_4 = f_2$$

$$f_8 = f_6 - \frac{1}{2}\left(f_1 - f_3\right)$$

$$f_7 = f_5 + \frac{1}{2}\left(f_1 - f_3\right)$$

10. New Density, Velocities

After collision, the density of node $\mathbf{x}$, at time $t+1$ was calculated:

$$\rho(\mathbf{x}, t+1) = \sum_{a=0}^{8} f_a(\mathbf{x}, t+1)$$

The momentum of node $\mathbf{x}$, at the time $t+1$ was:

$$\mathbf{U}(\mathbf{x}, t+1) = \sum_{a=0}^{8}\left[f_a(\mathbf{x}, t+1)\mathbf{e}_a \right]$$

Macroscopic velocity was given by:

$$\mathbf{u}(\mathbf{x}, t+1) = \frac{\mathbf{U}(\mathbf{x}, t+1)}{\rho(\mathbf{x}, t+1)}$$

In this example the D2Q9 model, where the dimensions are M=4 node, and N=3 node. In the Tables A2-2 to A2-11 the value for densities, velocities, momentum, equilibrium distribution function, and distribution functions at different steps of calculations were presented.

Table A2-2. Initialization

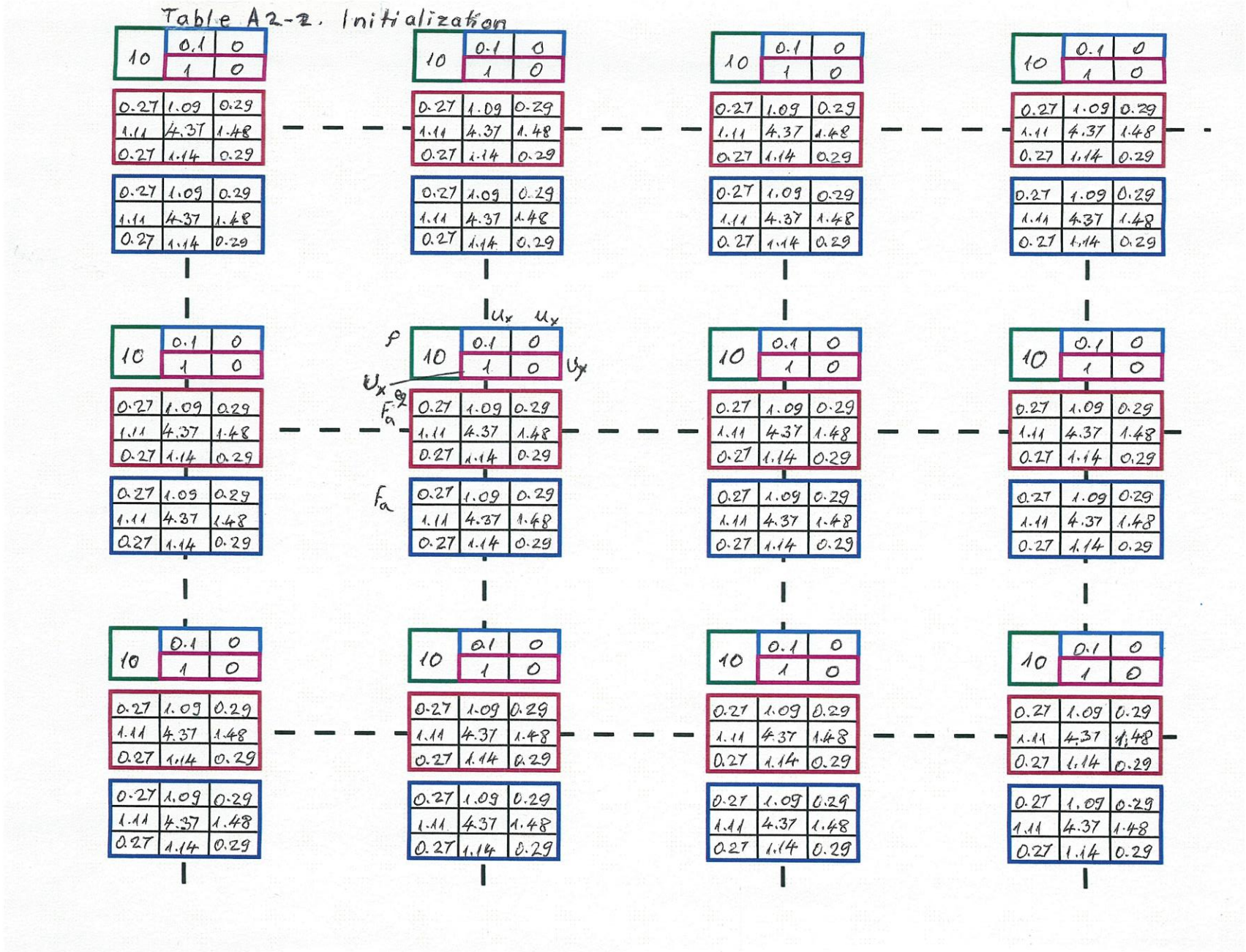
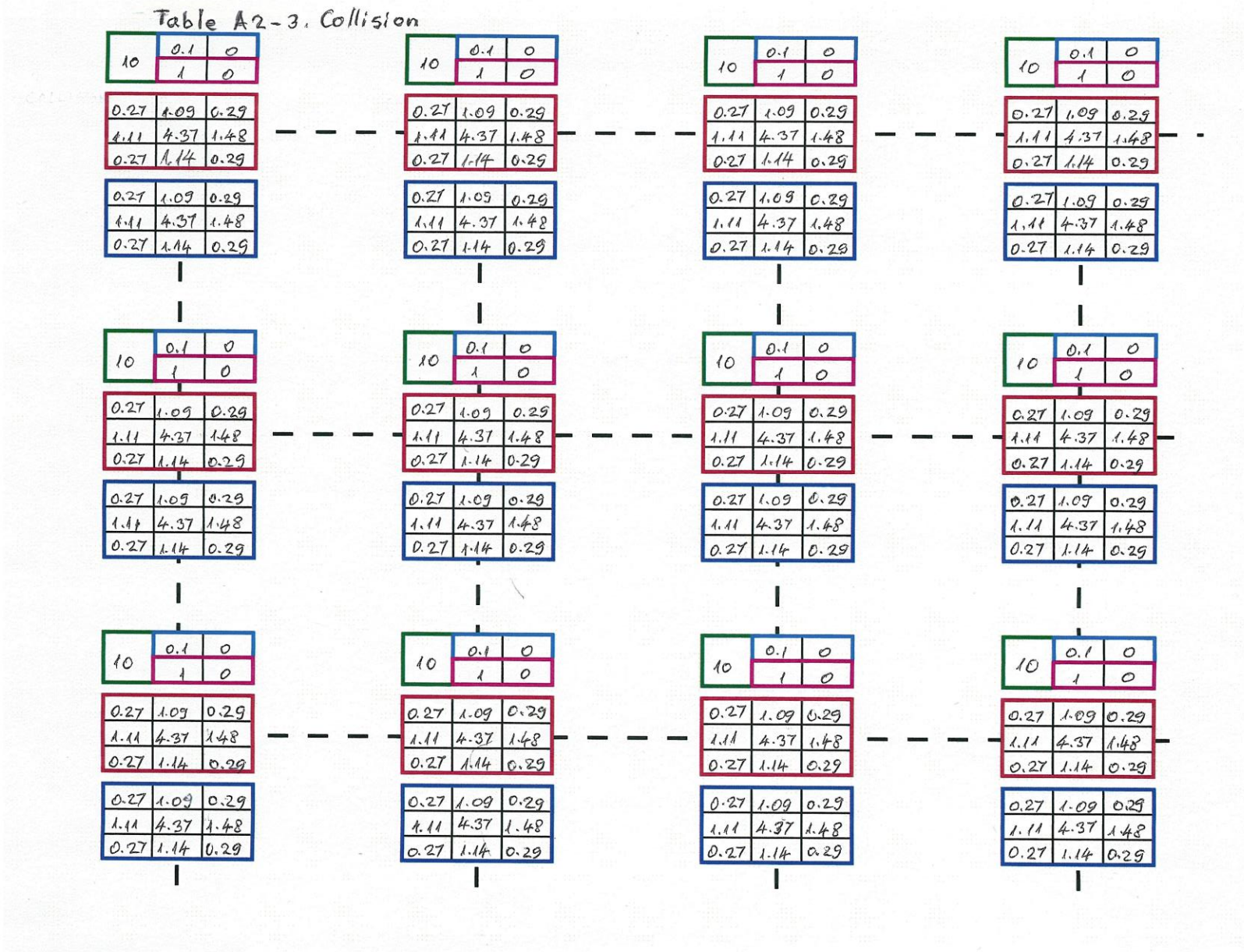


Table A2-3. Collision



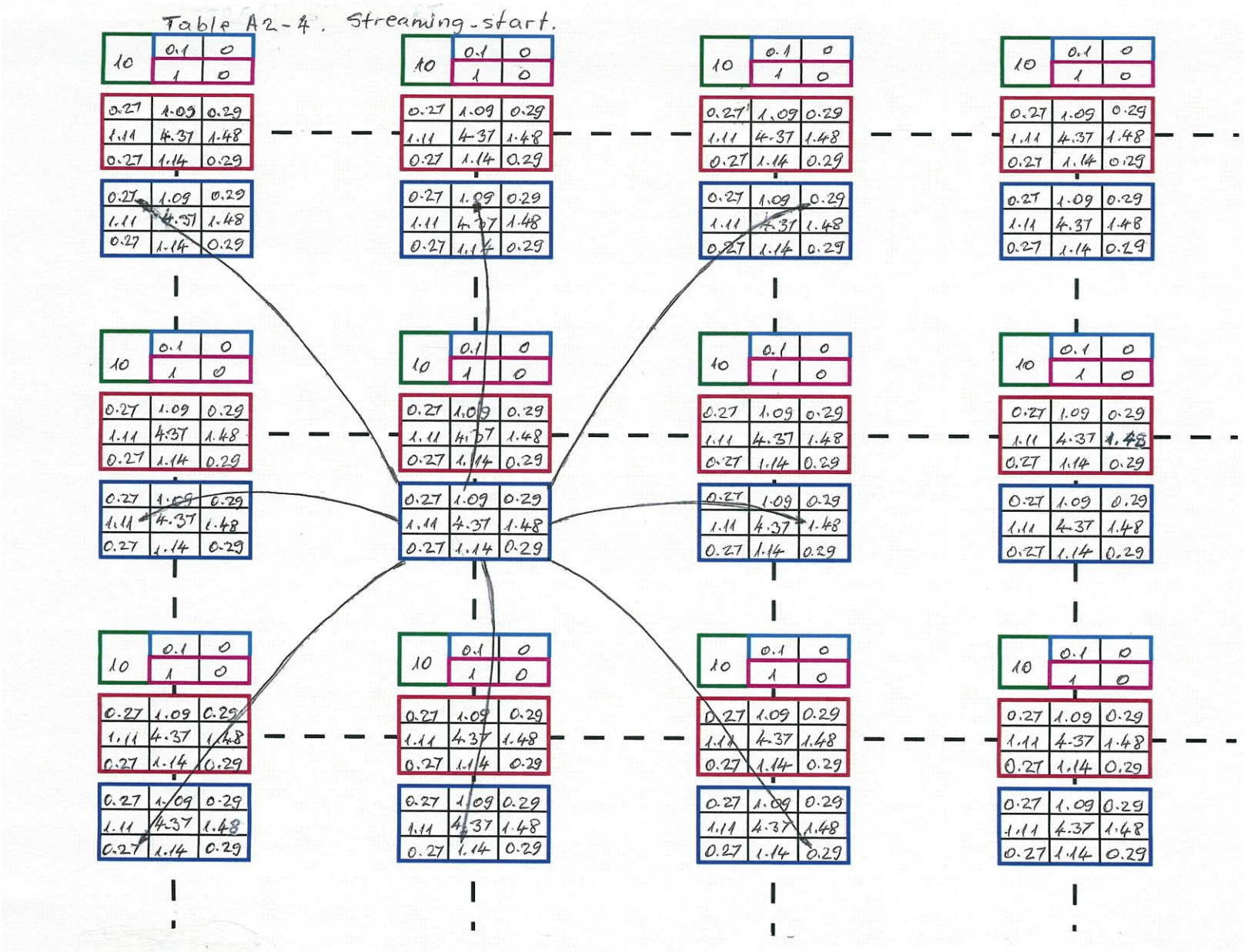


Table A2-4. Streaming-start.

Table A2- 5. Streaming - end

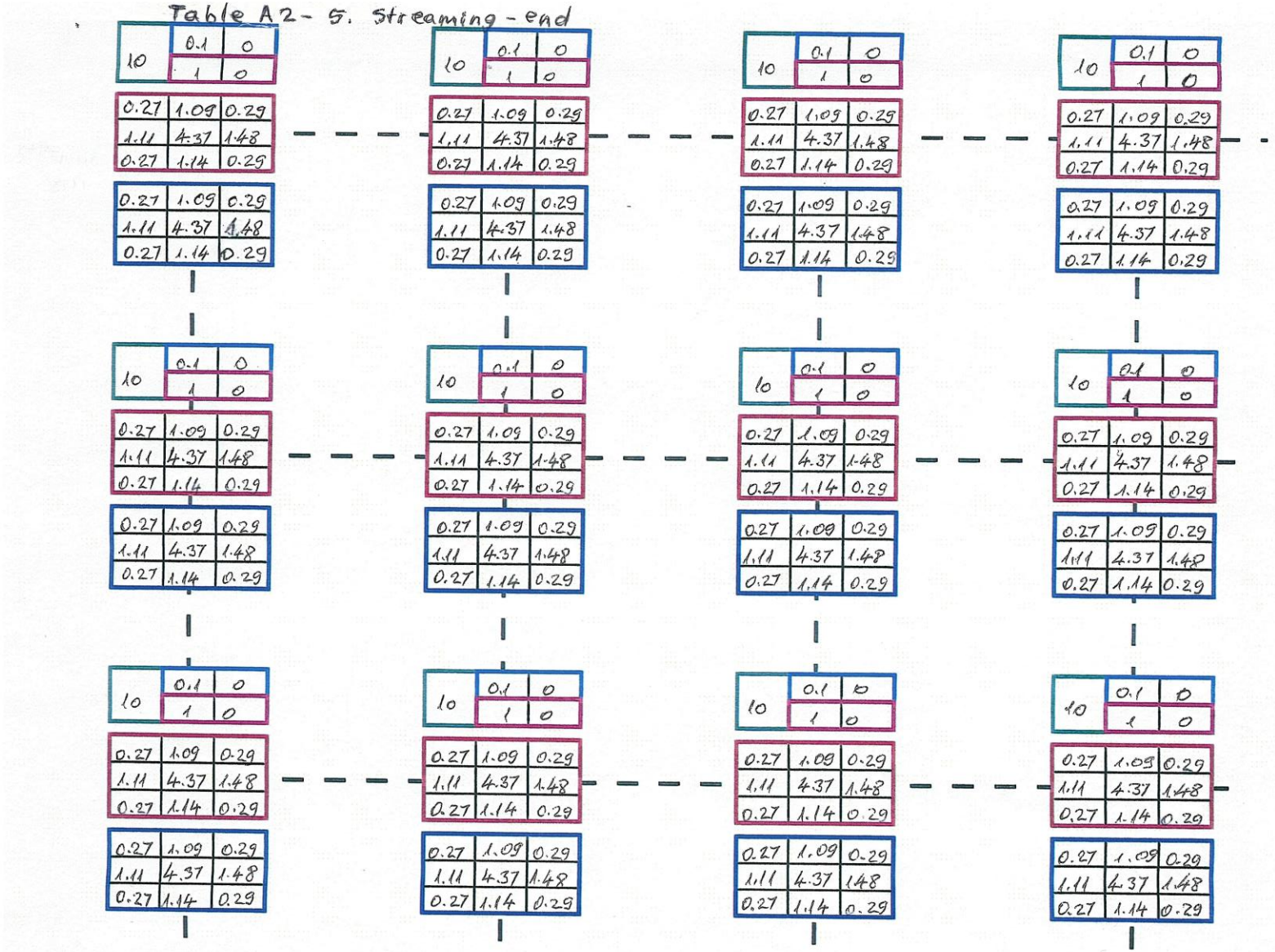


Table A2-6. Inlet.

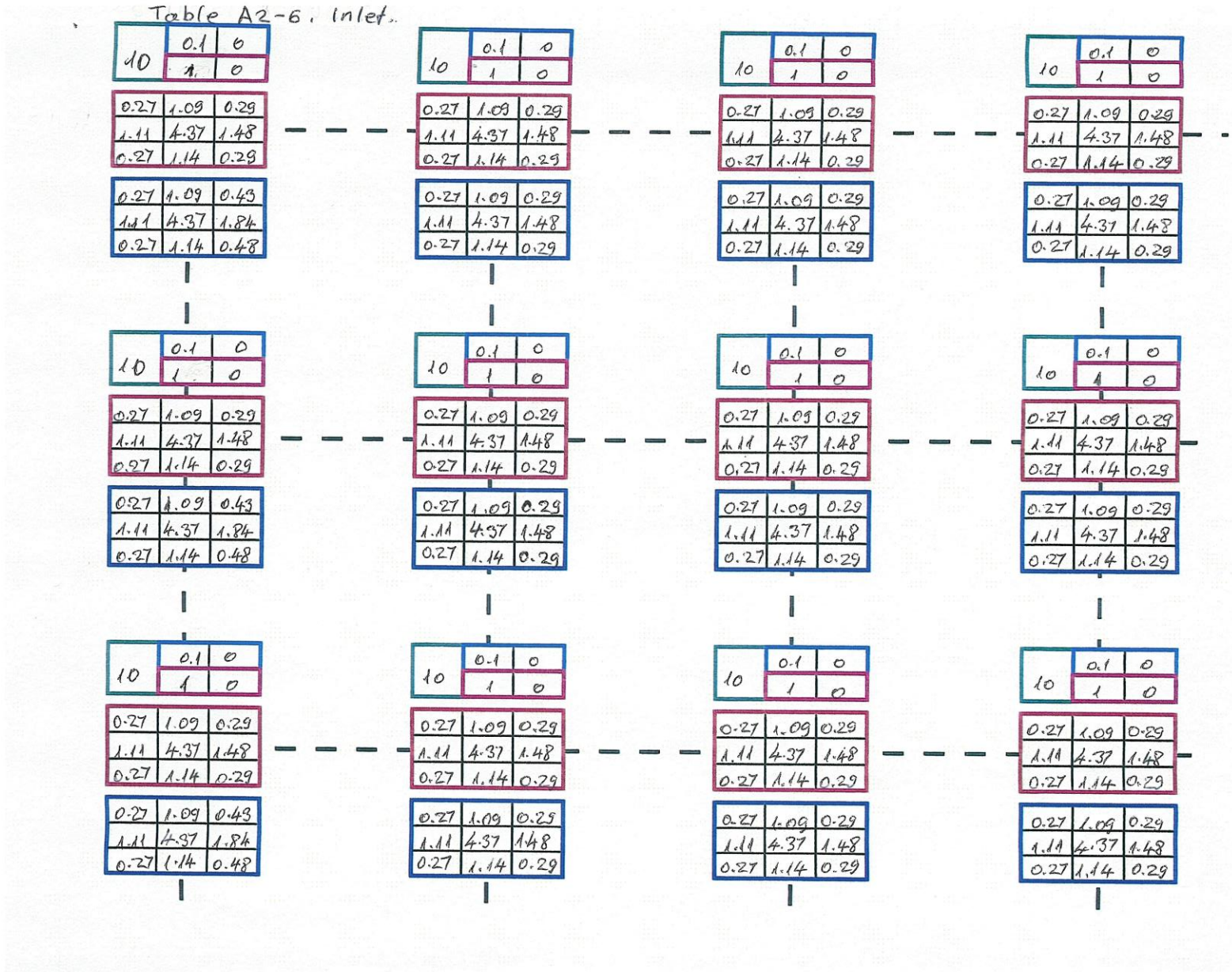


Table A2-7. Outlet boundary

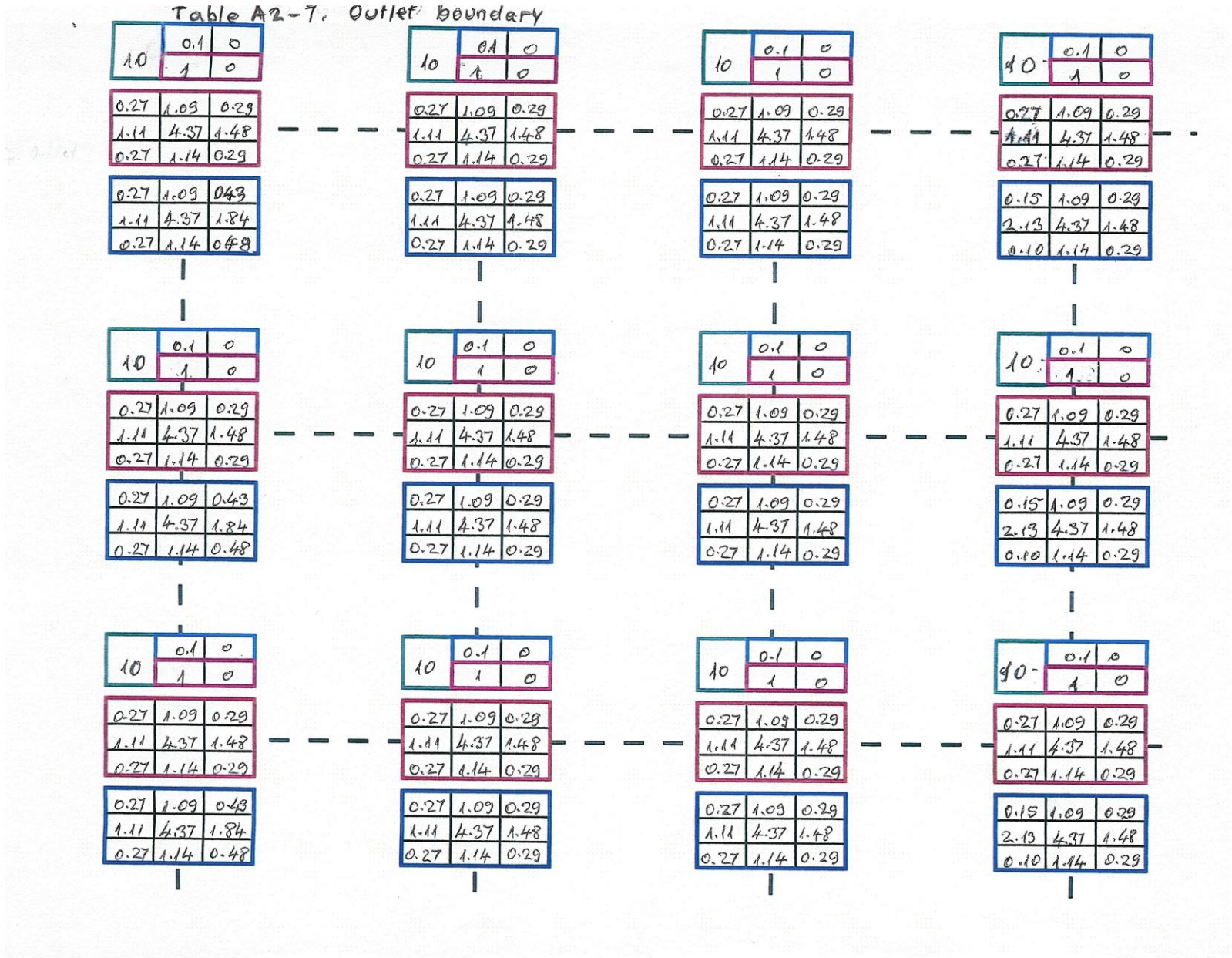


Table A2-8. Bottom boundary

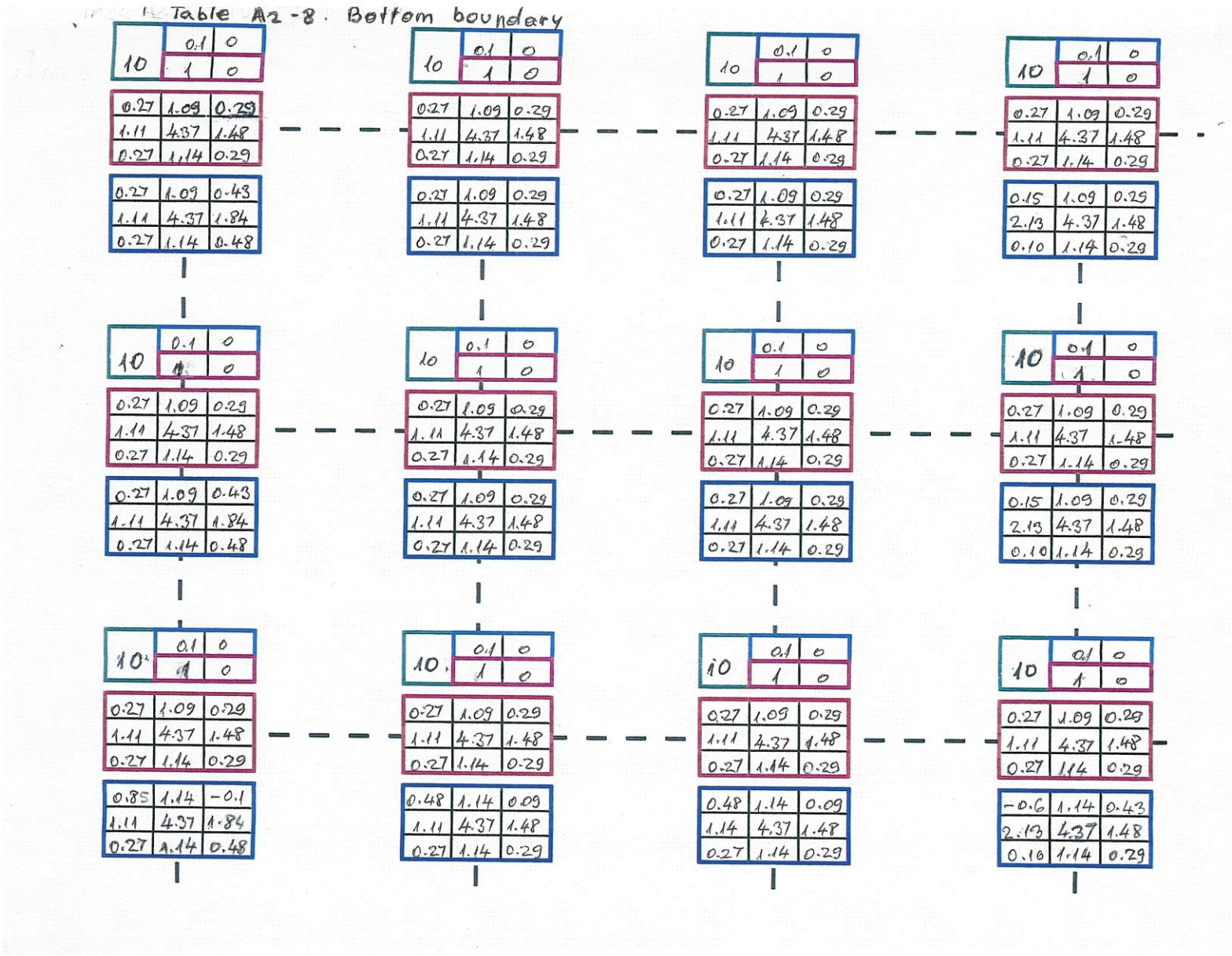


Table A2-9. Top boundary

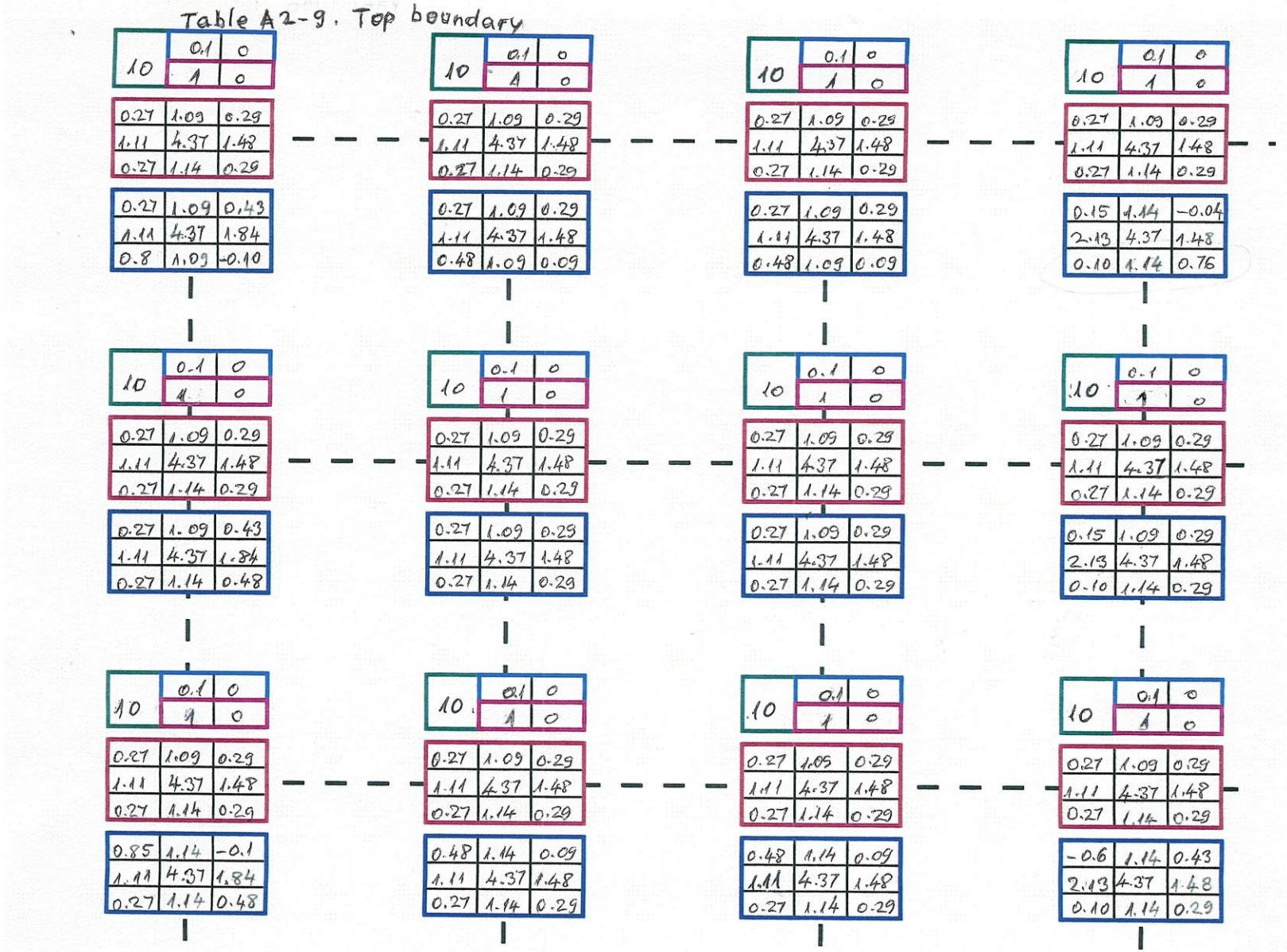


Table A2-10. New density, momentum, and velocity

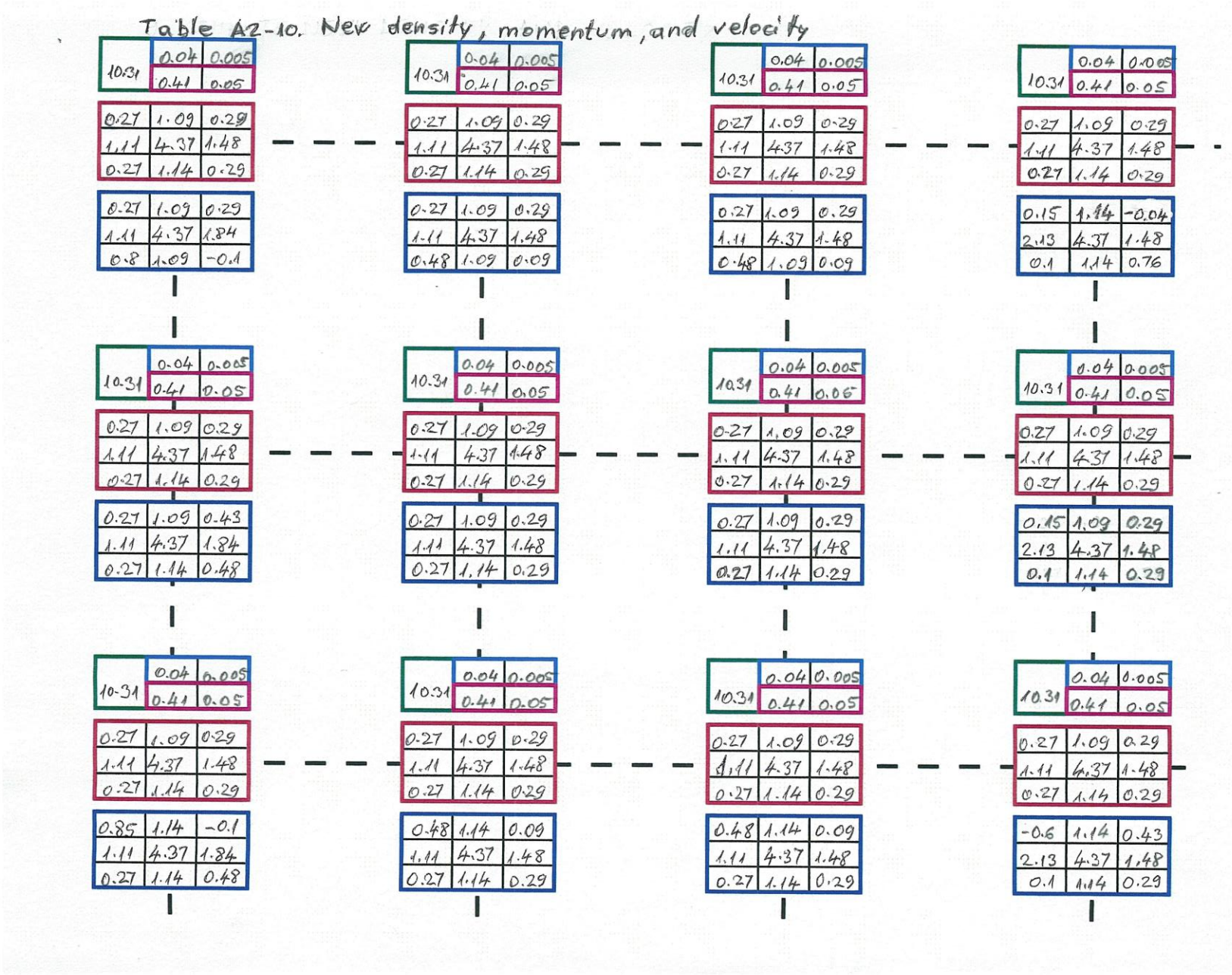


Table A2-11, Collision