NUMERICAL INVESTIGATIONS OF THE COUPLED DEM-LBM TECHNIQUE WITH APPLICATION TO LEAKAGE-SOIL INTERACTION DUE TO A LEAKING PIPE

JUN LI

BEng

A thesis submitted to The University of Birmingham

for the degree of

DOCTOR OF PHILOSOPHY

Department of Civil Engineering

The University of Birmingham

October 2012

ABSTRACT

Underground pipes have been widely used to transport water and sewage waste for drainage and water distributions. However, leakage problem of distribution pipes may result in localised underground cavities, which may expose infrastructures to the danger of collapse, or lead surface subsidence without any early warnings. Because of its special importance in the underground environment and safety, the interaction between leakage from a buried pipe and the surrounding soils has attracted attentions of infrastructure owners and geotechnical engineers. This, therefore, requires an understanding of the behaviour of a fluid-particle system.

This thesis aims to develop a numerical tool, *FPS-BHAM*, in exploring the large-scale fluid-particle system with local interaction behaviours being captured. A blocked partitioning domain decomposition strategy with the philosophies of parallel computing and combination with a large-scale modelling technique is proposed in this thesis.

The illustration of detailed implementation of DEM-LBM, with its verification in *FPS-BHAM* and its validation using a pipe leakage problem, are sequentially conducted. A good parallel behaviour is achieved by applying the blocked partitioning domain decomposition strategy, which is proposed in this thesis. The DEM-DFF technique is also successfully implemented in *FPS-BHAM* as well. Furthermore, a combination strategy between DEM-LBM and DEM-DFF is proposed in this thesis. A good

computational benefit is found to be achieved by adopting the proposed combination strategy. Finally, different behaviours between LBM and DFF during the dynamic propagation to the steady state are investigated by parametric studies.

KEY WORDS: Discrete Element Method, Lattice Boltzmann Method, Darcy Fluid Flow, Domain Decomposition, Combination Strategy, Parallel Computing

ACKNOWLEDGEMENTS

Firstly, I give praise to the creator of the world, our Lord Jesus Christ, for He gives me strength and wisdom to accomplish this thesis.

I wish to express my sincere gratitude to my superiors: Prof Andrew Chan and Dr John Bridgeman for their patient help and farsighted guidance throughout the course of my PhD study. A great appreciation goes to my wife, who is also my colleague, Ms Xilin Cui, for her steadfast love and support.

I wish to extend my gratitude to Dr Colin Thornton and Prof Charley Wu for their help with the *TRUBAL* code and their suggestions during the PhD course. I wish to thank Dr David Chapman and Dr Alexander Royal for their advices on the validation against experimental results. I would like to thank my former mentors: Prof Yuanlong Li and Dr Yingang Du as well, for their help and encouragement. A great thank goes to School of Civil Engineering for providing me a 3-year fully-funded PhD studentship. And I would like to appreciate the BlueBEAR Committee for their technique support by the HPC cluster for the numerical simulations in this thesis.

Finally, I thank my parents for their self-giving love and encouragement in all the years during my PhD course.

CONTENTS

LIST OF FIGURES, TABLES AND SYMBOLS

List of figures

List of tables

List of symbols

c	global damping coefficient
c'	global damping coefficient
C	LBM lattice speed
C_s	speed of sound
$\overline{d_p}$	average particle diameter in each cell
$\mathbf{e}_i$	LBM lattice velocities
$\mathbf{F}$	total force applied to particle
$\mathbf{F_c}$	total contact force
$\mathbf{F_b}$	body force
$\mathbf{F_f}$	fluid force applied to particle
$\mathbf{f}_b^p$	hydrostatic force
$\mathbf{f}_d$	hydrodynamic force
$\tilde{f}_i$	filtered density distribution functions
$\tilde{f}_i^{eq}$	filtered equilibrium density distribution functions
G	shear modulus

g	gravitational acceleration
g^*	modified gravitational acceleration
h	hydraulic head of the fluid.
$[k]$	permeability matrix.
k_n	contact stiffness in normal direction
k_t	contact stiffness in tangential direction
m	particle mass
Ma	Mach number
$N_{subcycle}$	number sub-cycles
n	porosity of soil bed
I	particle moment of inertia
p	fluid pressure
R_{ave}	average particle radius
S_c	Smagorinsky constant
T	total torque applied to particle
T_c	torques generated by contact force
T_f	torques generated from fluid force
u	fluid velocity
$u_{\max}$	maximum fluid velocity
V^P	volume of particle
v	particle translational velocity

$\mathbf{x}$	particle coordinates
Δh	LBM lattice spacing
$\Delta t_{c,DEM}$	critical time step used in DEM
Δt_{DEM}	DEM time step.
Δt_{LBM}	LBM time step.
$\Delta\boldsymbol{\alpha}$	relative displacement increments in the normal directions
$\Delta\boldsymbol{\delta}$	relative displacement increments in the tangential directions
λ	local solid/fluid ratio
μ	dynamic viscosity of fluid
ν	Poisson's ratio
$\boldsymbol{\theta}$	particle angular rotation
ρ	fluid density
ρ_s	particle density
τ_{total}	total relaxation time in LES model
τ	a dimensionless relaxation time
υ	kinematic viscosity of fluid
$\boldsymbol{\omega}$	particle rotational velocity
Ω_i^s	bounce back of non-equilibrium part of density distribution functions

CHAPTER 1: INTRODUCTION

1.1 Research background

Underground pipes have been widely used to transport water and sewage waste for drainage and water distributions. However, leakage problem of distribution pipes may result in underground cavities, which may expose infrastructures to the danger of collapse, or lead surface subsidence without any early warnings. Because of its special importance in the underground environment and safety, the interaction between leakage from a buried pipe and the surrounding soils has attracted attentions of infrastructure owners and geotechnical engineers.

In the vicinity of the leak (i.e. the leaking area), the flow paths of the leaking fluid in the soil could be greatly affected or even blocked due to the existence of soil. Away from the leaking area, only seepage flows that controlled by Darcy's law are assumed to exist when the soil is saturated. However, in the vicinity of the leaking area, the flow pattern of the leaking fluid can be significantly influenced by the high pressure and velocity as well as the presence of soil particles being transported by the fluid, and may demonstrate turbulent phenomenon. On the other hand, motion of the soil particles are mainly controlled by inter-particle contacts, gravity, pressure gradient force, and drag force by fluid flows. Hence, some of the soil particles may be mobilised and transported, (i.e. soil is locally fluidised.). Through such a process, a cavity may be produced and developed

gradually, which indicates a local large displacement occurring within the soil material.

The application of the analytical methods for solving such a problem encounters difficulties as the geo-materials are usually granular in nature, which exhibits inherently discontinuous, heterogeneous and generally anisotropic micromechanical behaviour. Moreover, the existence of multiphase interaction makes theoretical analysis even harder. Investigations using field and laboratory studies face many difficulties, such as disturbance of sample, difficulty in reproducing identical samples, and lack of transparency for visual measurements. Alternatively, numerical simulations may provide as an promising approach to explore the underlying mechanism of fluid-soil interaction.

1.2 Numerical techniques for modelling fluid-particle interactions

In recent years, numerical simulations of fluid-particle systems using a variety of techniques have been reported. Among those, the Discrete Element Method (DEM) has been regarded as an effective tool to trace the motion of solid particles (Cundall, 2001). Compared to the conventional Finite Element Method (FEM), DEM can be applied to a material subject to large displacements where compatibility condition is not valid (this is further explained in Section 2.2). Furthermore, both the macroscopic and microscopic analyses of a granular material can also be conducted through such method. Ever since DEM was proposed by Cundall (1971), different fluid methods have been coupled with it

to provide a solution to studies on the fluid flows and motion of particles with multiphase interaction. Among these methods, the coupling between DEM and Darcy fluid flow (DFF) (Hakuno and Tarumi, 1988) is a commonly-used technique for many geotechnical problems due to its simple form and low computational cost. Nevertheless, this technique is based on Darcy's law and only applicable to the flows with the Reynolds number less than ten (Shafipour and Soroush, 2008), e.g. laminar flows. Therefore, it is unsuitable to apply this technique to modelling the turbulent flows in the leaking area.

An alternative option is to couple the Lattice Boltzmann Method (LBM) with DEM (Cook et al, 2004). Unlike DFF, LBM with Large Eddy Simulation (LES) model can be applied in simulating turbulent flows. Furthermore, LBM directly describes fluid flows on the microscopic scale, so that the detailed flow behaviour can be traced. On the other hand compared with the conventional fluid method, such as Computational Fluid Dynamics (CFD), LBM is able to provide a relatively high computational efficiency in fine-grain modelling due to its explicit and local nature. Hence, the coupled DEM-LBM is regarded as a favourable tool in simulating the leaking area with a small-sized leak opening, which is presented in Section 2.7.

1.3 Aim of this research

Although DEM-LBM is quite efficient in small-scale modelling, the total computational

cost for simulating a real-scale problem could still be high due to fine grids adopted, which limits the applications of DEM-LBM technique. Therefore, it is necessary to enhance the performance of this technique and make it more applicable. In order to achieve this aim, two strategies are proposed in this thesis:

a) Parallel computing. Both DEM and LBM involve only local processes as well as the explicit schemes (explained in Sections 3.2 and 3.3, respectively), which makes DEM-LBM exhibit the highly natural parallelisation. In order to achieve optimum parallel performance, the computational load in each processor shall be as balanced as possible. The minimum serial calculation is required to be achieved so as to avoid idle waiting of processors. In addition, the data communication among processors shall be kept as small as possible.

b) Combining DEM-LBM with another simulation technique, which is more suitable for a large-scale simulation. For the leakage-soil interaction considered in this thesis, DEM-DFF can be employed to modelling the area far from the leakage, where the fluid flows are laminar. On the other hand, DEM-LBM is employed to investigate the refined leaking area with intensive changes in the fluid field and fluid-particle interactions. By combining these two techniques, an optimal performance in both computational efficiency and accuracy would be achieved.

Both the above philosophies lead to the domain decomposition scheme, in which the computational domain is decomposed into sub-domains, and various computational techniques are employed in different sub-domains according to local features. Thereafter, the calculations in each sub-domain are performed independently on a local processor working within a parallel computer.

Therefore, the aim of this thesis is to develop an efficient strategy for combination of DEM-DFF and DEM-LBM with parallel computing through domain decomposition. This is divided into the following objectives:

(i) to develop an efficient code named *FPS-BHAM* (abbreviation of *Fluid Particle System - University of Birmingham*) with successful implementation of DEM-LBM and DEM-DFF;

(ii) to propose a proper domain decomposition scheme in the program , which forms a basic platform of the combination of techniques with parallel computing;

(iii) to parallelise the code *FPS-BHAM*, so that parallel computing on a group of processors can be conducted; and

(iv) to propose a proper strategy for combination of DEM-DFF and DEM-LBM, so that the combined system works properly.

(v) to explore the pipe leakage problem using *FPS-BHAM*. The majority of the results is reported by Cui (2012). In this thesis, only a few of simple pipe leakage tests are presented to show the capability of this technique in modelling the leakage-soil

interaction.

1.4 Thesis layout

This thesis is divided into nine chapters. Chapter 2 gives an introduction of the fundamental theories involved in this thesis. Chapter 3 is devoted to a literature review of the studies on leakage-soil interaction; the different numerical techniques simulating fluid-particle systems; the parallel computing of DEM and LBM; and LBM combined with other large-scale modelling techniques. Chapter 4 provides details of the implementations of DEM-LBM and DEM-DFF in *FPS-BHAM*. In Chapter 5, the blocked partitioning domain decomposition in *FPS-BHAM* along with parallelisation scheme is presented. This is followed, in Chapter 6, by an illustration of the interface treatment between DEM-DFF and DEM-LBM sub-domains based on the platform of the blocked partitioning domain decomposition. In Chapter 7, results are provided to validate the strategies proposed in Chapters 4-6, including DEM-LBM implementation, the parallel performance of the proposed blocked partitioning domain decomposition, DEM-DFF simulations, and the proposed combination strategy between DEM-LBM and DEM-DFF. In Chapter 8, the influence of the different forms of governing equations in DFF and LBM on the performance of the combined system is under investigation. Finally, in Chapter 9, conclusions on the overall work presented in this thesis are provided and some further works are suggested.

CHAPTER 2: FUNDAMENTAL THEORIES

2.1 Introduction

The fundamental theories involved in this thesis is introduced in this chapter, including the discrete element method (DEM) (see Section 2.2), the lattice Boltzmann method (LBM) and its coupling with DEM (see Section 2.3), and the Darcy fluid flow (DFF) and its coupling with DEM (see Section 2.4).

2.2 Discrete element method (DEM)

DEM was initially proposed to deal with problems in rock mechanics (Cundall, 1971; and Cundall, 1974), and it has been increasingly used in the geotechnical field during recent decade (see e.g. O'Sullivan, 2011). In DEM, material is viewed as an assembly of discrete particles.

2.2.1 Soft-sphere DEM algorithm

Among literatures, two different algorithms are generally employed: hard-sphere and soft-sphere approaches (Duran, 2000). The soft-sphere DEM, which is employed in this thesis (see Section 3.2). Between any two particles in contact, a slight overlap is allowed. The contact is viewed as a dynamic process in which contact forces accumulate or dissipate over time. Contact forces can be subsequently obtained through the deformation

history at the contact. The motion of a single particle is governed by the Newton's second law in the form of the following equations:

$$m\frac{d\mathbf{v}}{dt} = \mathbf{F} \tag{2.1}$$

and

$$I\frac{d\boldsymbol{\omega}}{dt} = \mathbf{T}, \tag{2.2}$$

where $\mathbf{v}$, $\boldsymbol{\omega}$ are the translational and rotational velocity, and m and I are the particle mass and its moment of inertia. $\mathbf{F}$ and $\mathbf{T}$ are total external force and torque respectively applied to the particle.

The total force $\mathbf{F}$ and total torque $\mathbf{T}$ applied to the particle can be calculated by

$$\mathbf{F} = \mathbf{F}_c + \mathbf{F}_b + \mathbf{F}_f \tag{2.3}$$

and

$$\mathbf{T} = \mathbf{T}_c + \mathbf{T}_f, \tag{2.4}$$

where $\mathbf{F}_c$ is the summation of the contact forces over all contacts on the particle, i.e. the total contact force. $\mathbf{F}_b$ denotes the body force. $\mathbf{F}_f$ represents the fluid force applied on the particle, of which its calculations are demonstrated in Sections 2.3.2 and 2.4.2. $\mathbf{T}_c$ and $\mathbf{T}_f$ indicate the torques generated by the contact force and fluid force, respectively.

The calculation of the contact force is conducted in the normal and tangential directions as below:

$$\mathbf{F}_n^{t+\Delta t} = \mathbf{F}_n^t + k_n\Delta\boldsymbol{\alpha} \tag{2.5}$$

and

$$\mathbf{F}_t^{t+\Delta t} = \mathbf{F}_t^t + k_t \Delta \boldsymbol{\delta}, \tag{2.6}$$

where $\Delta\boldsymbol{\alpha}$ and $\Delta\boldsymbol{\delta}$ are the relative normal and tangential displacement increments within a DEM time step. k_n and k_t are the normal and tangential contact stiffnesses, which are computed by algorithms based on contact constitutive model (see Section 2.2.2).

For a sufficiently small DEM time step, the velocities and accelerations are assumed to be constant during each time step. With a central difference scheme, the location and rotation are determined as (Cundull and Strack, 1978)

$$\mathbf{x}^{t+\Delta t} = \mathbf{x}^t + \Delta t_{DEM} \cdot [\mathbf{v}^t \cdot (1 - \frac{c}{m}\frac{\Delta t_{DEM}}{2}) + \mathbf{F} \cdot \frac{\Delta t_{DEM}}{m}] / (1 + \frac{c}{m}\frac{\Delta t_{DEM}}{2}) \tag{2.7}$$

and

$$\boldsymbol{\theta}^{t+\Delta t} = \boldsymbol{\theta}^t + \Delta t_{DEM} \cdot [\boldsymbol{\omega}^t \cdot (1 - \frac{c'}{I}\frac{\Delta t_{DEM}}{2}) + \mathbf{T} \cdot \frac{\Delta t_{DEM}}{I}] / (1 + \frac{c'}{I}\frac{\Delta t_{DEM}}{2}), \tag{2.8}$$

where **x,** $\boldsymbol{\theta}$ are the particle coordinates and its angular rotation. *c*, *c'* are the global damping coefficients, and Δt_{DEM} denotes the DEM time step.

2.2.2 Contact constitutive model

The contact constitutive models involved in this thesis are the Hertz theory (Johnson, 1985), the theory of Mindlin and Deresiewicz (1953), and the Johnson-Kendall-Roberts (JKR) model (Johnson et al., 1971), which are introduced in this sub-section. For the

ease of explanation, some basic variables and their expressions are firstly defined.

As for two circular particles *A* and *B* in contact, their radii are denoted as R_A and R_B, masses as m_A and m_B, Young's modulus as E_A and E_B, shear modulus as G_A and G_B, and Poisson's ratios as ν_A and ν_B. The effective contact radius R^* is obtained from (see e.g. Yang, 2009)

$$1/R^* = 1/R_A + 1/R_B . \tag{2.9}$$

The effective mass m^* is obtained from

$$1/m^* = 1/m_A + 1/m_B . \tag{2.10}$$

The effective contact Young's modulus E^* is obtained from

$$1/E^* = (1-\nu_A^2)/E_A + (1-\nu_B^2)/E_B . \tag{2.11}$$

The effective contact shear modulus G^* is obtained from

$$1/G^* = (2-\nu_A)/G_A + (2-\nu_B)/G_B . \tag{2.12}$$

The relative approach between the two particles in contact (Figure 2.1) is expressed as

$$\boldsymbol{\alpha} = (R_A + R_B - D)\cdot \mathbf{n} \tag{2.13}$$

and

$$\alpha = R_A + R_B - D , \tag{2.14}$$

where D is the centre-to-centre distance and $\mathbf{n}$ is the unit vector pointing from centre of *A* to centre of *B*.

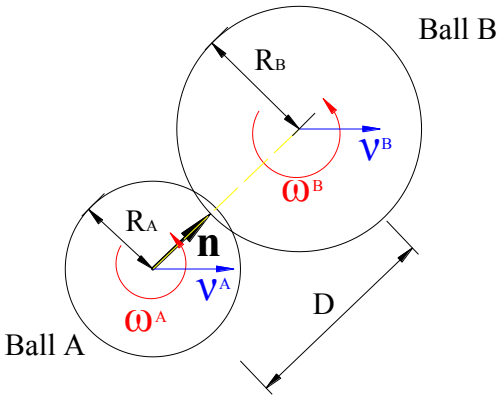


Figure 2.1: Two particles in contact

The relative normal and tangential displacement increments are obtained from

$$\Delta\alpha = \left(\mathbf{v}^B - \mathbf{v}^A\right)\cdot\mathbf{n}\cdot\Delta t_{DEM} \tag{2.15}$$

and

$$\Delta\boldsymbol{\delta} = \left(\mathbf{v}^B - \mathbf{v}^A\right)\Delta t_{DEM} - \Delta\alpha\mathbf{n} - \left(\boldsymbol{\omega}^A\times\mathbf{n}\right)R_A\Delta t_{DEM} - \left(\boldsymbol{\omega}^B\times\mathbf{n}\right)R_B\Delta t_{DEM}\,. \tag{2.16}$$

Hence, the tangential displacement is calculated by

$$\boldsymbol{\delta}^{t+\Delta t} = \boldsymbol{\delta}^t + \Delta\boldsymbol{\delta}\,. \tag{2.17}$$

During each DEM time step, the tangential force and displacement are re-orientated due to the rotation of the contact plane. The variables are adjusted by

$$\mathbf{F}_t^{re\text{-}oriented} = \boldsymbol{\Omega}\times\mathbf{F}_t^{old} \tag{2.18}$$

and

$$\boldsymbol{\delta}^{re-orientated} = \boldsymbol{\Omega}\times\boldsymbol{\delta}^{old}\,, \tag{2.19}$$

where $\boldsymbol{\Omega}$ is the rotation of the contact plane given by

$$\boldsymbol{\Omega} = \frac{\mathbf{n}}{D}\times\left(\left(\mathbf{v}^B - \mathbf{v}^A\right)\Delta t_{DEM} - \Delta\alpha\mathbf{n}\right)\,. \tag{2.20}$$

If the resultant tangential force satisfies the sliding criterion, sliding is assumed to occur and the tangential force is reset to the limiting value $\mu \cdot F_n$ (μ is the inter-particle friction coefficient).

The Hertz theory (see Johnson, 1985) is used to determine the normal contact stiffness,

$$k_n = 2E^* a_1, \tag{2.21}$$

where a_1 is the radius of the contact area calculated by

$$a_1 = \left(\alpha R^*\right)^{1/2}. \tag{2.22}$$

The tangential contact stiffness is determined by Thornton and Randall (1988) based on the theory of Mindlin and Deresiewicz (1953). By adopting an incremental approach, k_t can be calculated using the newly updated F_n and a_1. The expression for k_t is shown as follows (Thornton and Yin, 1991):

$$k_t = 8G^* a_1 \theta \pm \mu\left(1-\theta\right)\frac{\Delta F_n}{\Delta \delta}, \tag{2.23}$$

$$\theta = \left(1 - \frac{\left(F_t + \mu \Delta F_n\right)}{\mu F_n}\right)^{1/3} \qquad \Delta\delta > 0 \quad (loading) \tag{2.24}$$

$$\theta = \left(1 - \frac{\left(F_t^* - F_t + 2\mu \Delta F_n\right)}{2\mu F_n}\right)^{1/3} \qquad \Delta\delta < 0 \quad (unloading), \tag{2.25}$$

$$\theta = \left(1 - \frac{\left(F_t - F_t^{**} + 2\mu \Delta F_n\right)}{2\mu F_n}\right)^{1/3} \qquad \Delta\delta > 0 \quad (reloading), \tag{2.26}$$

and the negative sign in (2.23) is only taken during unloading. The forces F_t^* and F_t^{**} define the load reversal points and need to be continuously updated by

$$F_t^* = F_t^* + \mu\Delta F_n \tag{2.27}$$

and

$$F_t^{**} = F_t^{**} - \mu\Delta F_n \ , \tag{2.28}$$

If $|\Delta\delta| < \dfrac{\mu\Delta F_n}{8G^* a_1}$ with $\Delta F_n > 0$, let θ=1 until the condition $\sum 8G^* a_1 \Delta\delta > \mu \sum \Delta F_n$ is satisfied.

In order to account for inter-particle adhesion, the JKR model (Johnson et al., 1971) has been incorporated into the Hertz model to determine the normal contact stiffness (Thornton and Yin, 1991)

$$k_n = 2E^* a_2 \left(\frac{3 - 3\sqrt{\dfrac{a_3^3}{a_2^3}}}{3 - \sqrt{\dfrac{a_3^3}{a_2^3}}} \right), \tag{2.29}$$

where

$$a_2 = \left(\frac{3R^* F'}{4E^*} \right)^{1/3} , \tag{2.30}$$

$$a_3 = \left(\frac{3R^* F_c}{4E^*} \right)^{1/3} , \tag{2.31}$$

and F' is regarded to be the effective normal force

$$F' = F_n + 2F_c \pm \sqrt{4F_n F_c + 4F_c^2} . \tag{2.32}$$

If the particles are leaving each other, the negative sign is taken in (2.32). F_c is the so called 'pull-off' force indicating the minimum tensile force required to separate two adhesive particles in contact. It is given by

$$F_c = 3.0\pi\Gamma R^*, \tag{2.33}$$

where Γ is the surface energy of each solid particle in contact.

In the presence of tangential interactions, it is assumed that a peeling mechanism takes place initially when a tangential force is applied (Thornton, 1991). During this peeling process, the radius of the contact area reduces to

$$a_4 = \left(\frac{3R^*}{4E^*} \left(F_n + 2F_c \pm \sqrt{4F_nF_c + 4F_c^2 - \frac{F_t^2 E^*}{4G^*}} \right) \right)^{1/3}. \tag{2.34}$$

If the particles are leaving each other, the negative sign is taken in (2.34). The corresponding tangential stiffness is given by

$$k_t = 8G^* a_4 . \tag{2.35}$$

The peeling process terminates if the tangential force reaches a critical value of

$$F_{tc} = 4\left(\frac{G^*}{E^*} \left(F_nF_c + F_c^2 \right) \right)^{1/2}. \tag{2.36}$$

After the tangential force reaches to its critical value, sliding immediately happens if F_{tc} is greater than the sliding force F_{slc},

$$F_{slc} = \mu F' \left(\frac{2F' + F_n}{3F'} \right)^{3/2} \quad (F_n < -0.3F_c), \tag{2.37}$$

$$F_{slc} = \mu \left(F_n + 2F_c \right) \quad (F_n \geq -0.3F_c), \tag{2.38}$$

and the tangential force is then reset to F_{slc}. Otherwise, it undergoes a smooth transition to sliding. During this transition process, the radius of the contact area is given by

$$a_5 = \left(\frac{3R^*}{4E^*} \left(F_n + 2F_c \right) \right)^{1/3}, \tag{2.39}$$

and the corresponding tangential stiffness is calculated by (2.23)-(2.26) with F_n replaced by $F_n + 2F_c$, and a_1 replaced by a_5.

2.2.3 Critical time step in DEM

The critical time step used in DEM $\Delta t_{c,DEM}$ is calculated as (Yang, 2009)

$$\Delta t_{c,DEM} = \frac{\pi R_{min}}{\eta} \sqrt{\frac{\rho_s}{G}}, \tag{2.40}$$

in which

$$\eta = 0.8766 + 0.1631\nu, \tag{2.41}$$

where R_{min}, ρ_s, ν, and G are the minimum particle radius, particle density, particle Poisson's ratio and shear modulus, respectively. In consideration of computational stability, the time step used in DEM needs to be smaller than the critical one.

2.3 Lattice Boltzmann method (LBM) and its coupling with DEM

LBM originated from the late 1980s. It has been regarded as an alternative to the conventional macroscopic fluid models, and used for a wide range of applications (see Section 3.6). As indicated in Chen (1998), the lattice Boltzmann equation recovers the incompressible NS equations to the second order in both space and time, subject to the condition of low compressibility error.

2.3.1 Algorithm (Chen and Doolen, 1998)

In LBM, the fluid domain is divided into a square lattice with unity spacing. Fluid is assumed as packets of micro-particles residing on each lattice node. The D2Q9 model is adopted in this thesis, in which the velocity field is discretised into nine prescribed vectors (see Figure 2.2). In a unity time step, fluid particles are allowed either to remain at their current locations (which is referred to the zero velocity $\mathbf{e}_0$), or to travel to their adjacent nodes with velocities $\mathbf{e}_i \quad (i = 1,...8)$.

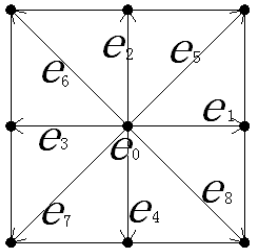


Figure 2.2: A typical lattice in D2Q9 model

As demonstrated in Figure 2.2, the prescribed velocities in a D2Q9 model are defined as

$$\mathbf{e}_0 = (0,0), \tag{2.42}$$

$$\mathbf{e}_i = C(\cos\{\pi(i-1)/2\}, \sin\{\pi(i-1)/2\}) \quad (i = 1,2,3,4), \tag{2.43}$$

$$\mathbf{e}_i = \sqrt{2}\cdot C(\cos\{\pi(2i-9)/4\}, \sin\{\pi(2i-9)/4\}) \quad (i = 5,6,7,8), \tag{2.44}$$

where C refers to the lattice speed, which is defined as the ratio of the unity lattice spacing Δh to LBM time step Δt_{LBM},

$$C = \Delta h / \Delta t_{LBM}. \tag{2.45}$$

C is also related to the fluid speed of sound C_s as

$$C = \sqrt{3} C_s . \tag{2.46}$$

Rather than describing a fluid flow by density, pressure, and flow rate, the primary variables in LBM are the density distribution functions f_i, along with the prescribed velocities $\mathbf{e}_i$. The governing equation for the notable lattice Boltzmann BGK model with Large Eddy Simulation (LES) turbulence implementation is shown as (Hou et al, 1996)

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t_{LBM}, t + \Delta t_{LBM}) = f_i(\mathbf{x}, t) - \frac{1}{\tau_{total}} [f_i(\mathbf{x}, t) - f^{eq}{}_i(\mathbf{x}, t)] \quad (i = 0, ..., 8) , \tag{2.47}$$

where f_i represents the probable quantity of micro-particles at a lattice node moving along the i^{th}-direction with velocity $\mathbf{e}_i$ at a particular time. f_i^{eq} are a set of distribution functions at which the systems are defined as the equilibrium, which is expressed as (Chen and Doolen, 1998)

$$f_i^{eq} = w_i \rho [1 + 3(\mathbf{e}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{e}_i \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u} \cdot \mathbf{u}], \tag{2.48}$$

$$w_0 = \frac{4}{9}, \qquad w_i = \frac{1}{9} \quad (i = 1, 2, 3, , \qquad w_i = \frac{1}{36} \quad (i = 5, 6, 7, ,$$

where $\mathbf{u}$ and ρ are the fluid velocity and density, respectively. τ_{total} is the total relaxation time, which is calculated as (Hou et al, 1996)

$$\tau_{total} = \frac{1}{2}[\tau + \sqrt{\tau^2 + \frac{18 \Delta h \cdot S_c^2}{\rho \cdot C^3} \bar{Q}}], \tag{2.49}$$

where S_c is the Smagorinsky constant, $\bar{Q}$ is related to the second-order moments of the non-equilibrium distribution functions, which is calculated as (2.50). And τ is a dimensionless relaxation time determined by (2.52).

$$\bar{Q} = \sqrt{2\sum_{i,j} \tilde{Q}_{ij} \tilde{Q}_{ij}} \tag{2.50}$$

$$\tilde{Q}_{ij} = \sum_{\alpha=0}^{8} e_{\alpha i} e_{\alpha j} (f_\alpha - f^{eq}{}_\alpha), \tag{2.51}$$

$$\tau = \frac{3 \cdot \upsilon}{\Delta h \cdot C} + \frac{1}{2}, \tag{2.52}$$

where υ is the kinematic viscosity of the fluid, and $e_{\alpha i}$ is the i^{th} component of the lattice velocity $\mathbf{e}_\alpha$.

Each LBM time step consists of two steps. Firstly, collision happens among the micro-particles travelling to a same node, which leads to the changes in the density distribution functions at that node. Thereafter, streaming process is conducted, in which the post-collision density distribution functions are streamed to their neighbouring nodes along the i^{th}-direction.

The macroscopic fluid variables such as density ρ, velocity **u** and pressure p can be given based on the philosophy of the mass and momentum conservations and the equation of state, respectively (Chen and Doolen, 1998):

$$\rho = \sum_i f_i, \tag{2.53}$$

$$\rho \mathbf{u} = \sum_i f_i \mathbf{e}_i, \tag{2.54}$$

$$\Delta p = C_s^2 \Delta \rho. \tag{2.55}$$

2.3.2 Boundary conditions

In LBM, the boundary conditions are applied to the computational domain through updating the density distribution functions at the boundary nodes. In this sub-section, the fluid-wall and fluid-particle interactions used in this thesis are illustrated, respectively.

No-slip stationary wall

No-slip stationary wall is a commonly-encountered boundary condition. The most popular way in applying it in LBM is the bounce-back rule. It is illustrated by He et al (1997) that the bounce-back rule achieves the second order accuracy when the wall is located half spacing away from the boundary nodes (Figure 2.3). It is assumed an incoming fluid particle will reflect back along its original direction,

$$f_{-i}(\mathbf{x}, t + \Delta t_{LBM}) = f_i(\mathbf{x}, t)\,, \tag{2.56}$$

where $-i$ stands for the opposite direction of i.

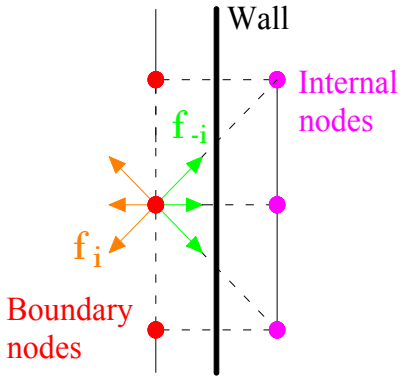


Figure 2.3: Bounce-back rule in LBM

No-slip moving boundary condition

In order to deal with the interactions between fluid flows and a moving solid boundary in LBM, Noble and Torczynski (1998) proposed Immersed Moving Boundary (IMB) condition. In IMB, the solid particles are firstly mapped onto the LBM framework. A nodal cell with the size identical to the lattice spacing is set up for each lattice node with its centre located at the node, see Figure 2.4. The local solid/fluid ratio λ is employed to represent the volume fraction of the nodal cell covered by a moving particle.

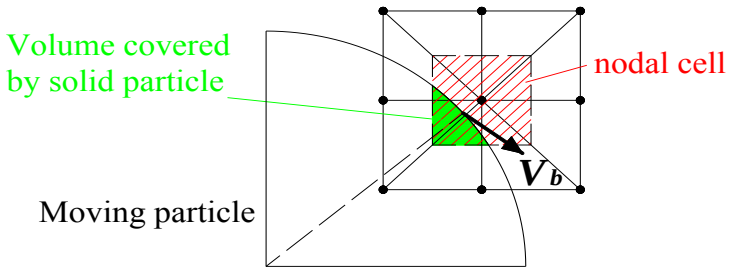


Figure 2.4: A nodal cell in IMB

The modified LBM governing equation with the implementation of IMB is shown as (Noble and Torczynski, 1998)

$$f_i(\mathbf{x}+\mathbf{e}_i\Delta t_{LBM}, t+\Delta t_{LBM}) = f_i(\mathbf{x},t) - \frac{1}{\tau_{total}}(1-\lambda)[f_i(\mathbf{x},t) - f^{eq}{}_i(\mathbf{x},t)] + \lambda\Omega_i^s, \quad (2.57)$$

$$\Omega_i^s = f_{-i}(\mathbf{x},t) - f_i(\mathbf{x},t) + f_i^{eq}(\rho, \mathbf{v}_b) - f^{eq}{}_{-i}(\rho, \mathbf{u}), \quad (2.58)$$

where Ω_i^s stands for the non-equilibrium component of the density distribution functions being bounced back. The bounce-back effect is considered on all the boundary

nodes along the directions pointing to the particles. According to the Newton's third law, the total effect of the bounce-back on the particle should be equal and opposite to the fluid force applied to the particle. Based on this philosophy, the total fluid force and torque acting on the moving particle can be obtained from the following equations (Feng et al, 2007):

$$\mathbf{F}_f = C\Delta h(\sum_m \lambda_m \sum_i \Omega_i^s \mathbf{e}_i), \tag{2.59}$$

$$\mathbf{T}_f = C\Delta h \sum_m (\mathbf{x}_m - \mathbf{x}_c) \times (\lambda_m \sum_i \Omega_i^s \mathbf{e}_i), \tag{2.60}$$

where $\mathbf{x}_m - \mathbf{x}_c$ denotes the vector from a lattice node to the particle centre. And the summation is conducted along the particle boundary. The calculated $\mathbf{F}_f$ and $\mathbf{T}_f$ are then used in DEM in (2.3) and (2.4). So far, a complete treatment for the fluid-particle interaction has been achieved.

2.4 Darcy fluid flow (DFF) and its coupling with DEM

DFF model is widely used in geotechnical applications, especially in simulating underground seepage flows, for its simplicity and low computational requirement (see Section 3.4).

2.4.1 Algorithm

The mass conservation for fluid flow is shown as

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho \mathbf{u})}{\partial \mathbf{x}} = 0, \tag{2.61}$$

where ρ is the density of fluid. $\mathbf{u}$ is the velocity vector. For seepage flow, the Mach number is usually low and the change of pressure is limited, the flow can be assumed incompressible. Therefore, the above equation can be simplified as

$$\frac{\partial \mathbf{u}}{\partial \mathbf{x}} = 0. \tag{2.62}$$

For a two-dimensional case, (2.62) can be written as

$$\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} = 0, \tag{2.63}$$

where u_x, u_y, $\frac{\partial u_x}{\partial x}$ and $\frac{\partial u_y}{\partial y}$ are fluid velocities, and velocity gradients in the x and y directions, respectively. The flow rate through a soil mass can be determined by the Darcy's law,

$$\begin{pmatrix} u_x \\ u_y \end{pmatrix} = \begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix} \begin{pmatrix} -\frac{\partial h}{\partial x} \\ -\frac{\partial h}{\partial y} \end{pmatrix}, \tag{2.64}$$

where h is the hydraulic head of the fluid. [k] is the permeability matrix. $\frac{\partial h}{\partial x}$ and $\frac{\partial h}{\partial y}$ are hydraulic gradient in the x and y directions, respectively. The relationship between the hydraulic head and the pore fluid pressure p is shown as

$$h = \frac{p}{\rho g} + y, \tag{2.65}$$

where y is the elevation of a point above the datum, and g is the gravitational acceleration. Combined (2.63)-(2.65), the governing equation used for Darcy fluid flow can be obtained as,

$$\frac{\partial}{\partial x}(k_{xx}(-\frac{\partial p}{\partial x})+k_{xy}(-\frac{\partial p}{\partial y}))+\frac{\partial}{\partial y}(k_{yx}(-\frac{\partial p}{\partial x})+k_{yy}(-\frac{\partial p}{\partial y}))=0\,. \tag{2.66}$$

2.4.2 DFF coupling with DEM

The two way coupling between DEM and DFF in undrained simulations (see Section 3.4) has facilitated the following philosophies: a) the particle motions are considered to induce changes in local permeability, and hence in seepage flows (by (2.66)); and b) the fluid exerts the hydrostatic force $\mathbf{f}_b^p$ and hydrodynamic force $\mathbf{f}_d$ on the solid particles.

The hydrostatic force $\mathbf{f}_b^p$ is generated due to the presence of pressure gradient. The force generated by pressure is taken into consideration, which is calculated as

$$\mathbf{f}_b^p = -(\nabla \mathbf{p})V^P\,, \tag{2.67}$$

where V^P is the volume of the particle. $\nabla \mathbf{p}$, the gradient of pressure, is calculated from central difference approach

$$\nabla p_x = \frac{p_{i+1,j}-p_{i-1,j}}{2\cdot\Delta x} \tag{2.68}$$

and

$$\nabla p_y = \frac{p_{i,j+1}-p_{i,j-1}}{2\cdot\Delta y}\,. \tag{2.69}$$

The hydrodynamic force includes the drag, lift and virtual mass forces. However, the lift and virtual mass force can be negligible compared to the drag force in the Darcy fluid

flow (Shafipour, 2008). The drag force can be expressed by a semi-empirical Ergun's Equation (Ergun, 1952),

$$\mathbf{f}_d = 150\mu \cdot (1-n)^2 \cdot |\mathbf{u}-\mathbf{v}| / (n\overline{d_p}^2) + 1.75 \cdot (1-n)\rho \cdot (\mathbf{u}-\mathbf{v})^2 / \overline{d}_p, \tag{2.70}$$

where n is the porosity of the soil bed; μ and ρ are the pore fluid dynamic viscosity and density, respectively; $\overline{d_p}$ is the average particle diameter in the each cell; $\mathbf{v}$ is particle velocity and $\mathbf{u}$ is the average fluid velocities in a cell, which is calculated as

$$u_i = -k \frac{\nabla p_i}{\rho \cdot g}. \tag{2.71}$$

Hence, the total force exerted on a single particle by fluid, denoted as $\mathbf{F}_f$, is expressed as (Shafipour, 2008)

$$\mathbf{F}_f = \mathbf{f}_b^p + \frac{\mathbf{f}_d}{1-n} V^p, \tag{2.72}$$

which is used in (2.3) for solving the new particle location in the DEM calculation.

CHAPTER 3: LITERATURE REVIEW

3.1 Leakage-soil interaction induced by a pipe leakage

Leakages are frequently found in underground transportation pipes (Rajani and McDonald, 1994; Makar, 2000; and Kunkel et al, 2008). A leak in a pipe may be attributed to various factors, such as seasonal changes, the differential settlements caused by heavy traffic loads, etc (Cui, 2012). Among literatures, the amount of water leaking from a broken pipe has drawn many attentions (Farley, 2001; Puust et al, 2010; Noack and Ulanicki, 2006). In addition, a leakage could also lead to local large displacement in the surrounding soil, and this would further induce many serious accidents relating to both safety and economic issues all around the world (Söderlund et al, 2007; Steindorff, 2008; and Lynch and Stimpson, 2011). However, due to the complicity of the physical behaviour of the phenomenon, to the authors' knowledge, there are no literatures published addressing the leakage-soil interaction due to a buried leaking pipe.

Laboratory tests were conducted on fine-grained soils subjected to a pipe leakage under different pressure settings (Rogers et al, 2008; Royal et al, 2008; and Supraksorn, 2009). From the experimental results, three failure mechanisms in the soil are identified: (i) permeation of the leaking fluid into soil; (ii) cavity being created within the soil; and (iii) the soil being ruptured with the leaking fluid migrated up to the top surface. However,

due to the lack of transparency of the 3D experiment, it is difficult to observe what happens inside the soil and also to acquire sufficient data for further analysis. In order to perform the observation inside the soil bed, a two-dimensional experimental study was carried out on a slot-shaped opening under the soil sample (Alsaydalani, 2010). With the aid of the Particle Image Velocimetery technique, the behaviour within the sample could be monitored and analysed. Some information, such as the vertical distribution of excessive pressure, the cavity size, and the particle velocity distribution could be obtained through the successful investigations. However, more information is expected in order to develop a better understanding of the leakage-soil interaction, such as the spatial distribution of excessive pressure, the inter-particle forces, the fluid forces applied to particles, etc. Such information can be easily obtained from analytical or numerical approach. In addition, the experiment faces the difficulties in reproducing identical samples, and the time-consuming nature of a laboratory work.

On the other hand, it has been identified that the behaviour of the leakage-soil interactions are influenced by quite a number of factors (Cui, 2012). The discontinuous and heterogeneous behaviour of the soil bed, the turbulent nature of the leaking water, and the rapid soil geometry change, make such a problem complicated in nature and difficult in theoretical analysis.

Compared with the theoretical and experimental approaches, a numerical tool is

expected in exploring the leakage-soil interactions, with the advantage of an easy acquisition of data as well as a capability of providing repeatable samples under various conditions. Therefore, in the following sections, the literature review is mainly conducted on the numerical tools for simulating geotechnical problems during the past several decades, which includes DEM-CFD, DEM-DFF, DEM-SPH, and DEM-LBM.

3.2 DEM and its application in soil mechanics

Before the 1970s, most researches in soil mechanics were conducted by the continuum approach, which is based on three assumptions: continuity, homogeneity and isotropy (Malvern, 1969). However, the real granular materials consist of particulate materials in contact and the surrounding voids, which determines the inherently discontinuous, heterogeneous and generally anisotropic micromechanical behaviour of soil materials. The limitation of such a conventional continuum model led to the development of DEM in the study of granular materials (Cundall, 1971). DEM is able to handle discontinuous material with a large displacement and mirco-structure rearrangement. It provides the complete information of any single particle over time. By statistical approach, the macroscopic properties including spatial distributions, stress tensors and strain tensors can also be obtained (Cundall and Strack, 1978; Gong, 2008). All these features boost the development of DEM, and make more and more researchers realising the importance of this discontinues approach in geotechnical applications. It has been

shown by Cundall and Hart (1992) that compared with other numerical tools such as Finite Element Method (FEM) and Boundary Element Method (BEM), DEM is more capable in modelling a discontinuous material. Cundall (2001) suggested that, as the improvement of computer hardware/software, the future trend for soil and rock modelling may be conducted by DEM rather than continuum computational methods.

As pointed out by Duran (2000), DEM models are normally classified into two categories as "soft-sphere" which was the original DEM model, and "hard-sphere" which was firstly introduced by Campbell and Brennen (1985). In the "hard-sphere" model, no overlap in particles is allowed during collisions. The contacts are assumed to happen instantly and the contact forces are calculated based on the energy equilibrium, which makes this model quite efficient in the system that is not densely packed (Deen et al, 2007). However, for most geotechnical problems, in which the quasi-static state of soil involved, the multiple collisions and the complex contact behaviours are difficult to handle in this model. In contrast, in the "soft-sphere" model, a small amount of overlap is allowed at the contact points. The contact forces are determined based on the overlap area in different contact constitutive theories. This model provides ease of treatment to multiple collisions and complex contact behaviour, leading to its prevalence in geotechnical application (O'Sullivan, 2011).

In the "soft-sphere" model, a virtual spring connects the two contacting particles at the

contact points in both the normal and tangential directions. In the original DEM model, a linear spring contact constitutive theory was used, indicating that the force and displacement relationship of the spring follows the Hooke's Law (Cundall and Strack, 1978, 1979). However, due to the change of contact area during collision, it is considered that the response of elastic spheres in contact is more accurately described by the Hertz contact theory (Johnson, 1985). In 1988, Cundall employed the Hertz normal contact theory and a simplified Mindlin tangential contact theory (Mindlin, 1949) in DEM. Later, the complete tangential contact theory of Mindlin and Deresiewcz (1953) combined with the Hertz normal contact theory was implemented in DEM by Thornton and Randall in 1988.

The classical Hertz contact theory neglects adhesion of particles, thereby an important improvement was incorporating the JKR adhesive theory (Johnson, 1985) into DEM, which is done by Thornton and Yin (1991). In geotechnical applications, such an improvement indicates the extension of DEM applications from simulating cohesionless soils to capturing the mechanical response of cohesive soils (Delenne et al, 2004; Cui et al, 2012). In the JKR theory, the adhesive force between two particles is regarded as a 'pull-off' force indicating the minimum tensile force required to break the contact. Its value is closely related to the particle surface energy. Theoretically speaking, this parameter is difficult to evaluate because it depends on many variables related to physical and chemical properties (Israelachvili, 1991). And in numerical simulations,

this parameter is often determined empirically (Zhu, 2007). However, one notable work is done by Cui (2012) to explore the relationship between the apparent cohesion of a soil and the particle surface energy in geotechnical applications.

In recent decades, irregular-shaped particles are becoming more and more popular in DEM simulations, e.g. Lin and Ng (1997) and Han et al (2007b). In their work, assemblies of ellipsoids and polygons were employed, respectively, to provide resistance to rolling motion and to capture the inherent geometry-dependent asperity behaviour. An alternative approach is to use 'clumps' of particles, in which particles are bonded together at the contact points, e.g. McDowell and Harireche (2002). In this thesis only circular particles are adopted for the sake of simplicity and computational economy. However, employing irregular-shaped particles in future research can be beneficial, which provides a way in investigating the particle interlock behaviour in the leakage-soil interaction.

As the interactions between soil and fluid flows are quite common, soil behaviour in geotechnical problem does not merely depend on the behaviour of soil phase, but also on the fluid movement in the ground. Among these cases, some can be solved with certain hypotheses, e.g. Ng and Dobry's work (1994). In their work, the undrained condition was simulated by assuming that, no volume changes occurred in shear test before sample failure (i.e. the constant volume assumption). However, in most cases

with a rapid motion of groundwater, which is also exhibited in the leakage-soil interaction, the explicit consideration of pore fluid is necessary. Therefore, in the recent decades, DEM has been incorporated with fluid modelling methods so that the interactions between the soil and fluid flows can be described. The following sections (Sections 2.3-2.7) give a literature review of the widely used techniques coupling DEM with different fluid methods.

3.3 Coupled DEM-CFD technique

DEM was coupled with Computational Fluid Dynamics (CFD) by Tsuji et al (1993) in simulations of a two-dimensional fluidised bed. Good agreement with the experimental results was achieved in terms of particle motion. The fluid-particle systems consist of large-sized fluid cells with small-sized particles submerged in them. For the fluid phase, the local averaged Navier–Stokes (NS) equations were solved, which was originated from Two Fluid Model (TFM) (Anderson and Jackson, 1967). These equations were integrated with the SIMPLE method (Patankar, 1980) in the staggered grid system. By solving the NS equations, a local averaged pore fluid pressure and velocity within each CFD cell were obtained. Thereafter, the determination of the local averaged fluid force applied to the particles is conducted with hydrostatic and hydrodynamic force equations. This fluid force was applied to particle phase for DEM computations, as well as to the local momentum term in NS equation for representing the influence of particles on the

fluid phase.

During the recent two decades, DEM-CFD has been widely accepted in both research and industry fields, and useful reviews can be found in Zhu et al (2007) and Yang (2009). Among the literatures, some notable publications are working on extending applications of this technique: Kawaguchi et al (1998) extended its applications to three-dimensional simulations, and Mikami et al (1998) employed this technique in the simulations of cohesive particles. For liquid-particle systems, particle-particle collisions are influenced by the drainage of the liquid between the colliding particles and the acceleration of the liquid surrounding the particles. In order to consider such an influence, Zhang et al (1999) modified DEM-CFD by incorporating two forces, i.e. the pressure force and virtual mass forces. This model was then successfully used by Li et al (2001) for the hydrodynamic description of a gas-liquid-particle flow. In order to improve the accuracy and to modify the algorithm of this technique, researches were carried out including: Xu and Yu (1997) which modified the gas-particle interaction implementation, and Feng and Yu (2004) which evaluated the formulations of the local averaged Navier–Stokes equations. Furthermore, Hoomans et al (1996) employed the "hard-sphere" model in DEM-CFD in simulating gas–solid two-phase flows in gas-fluidised beds.

The fluid force applied to an isolated particle has been well-established, and a summary

of the equations can be accessed in Zhu et al (2007). However, for a particulate system, even though the local porosity is employed in the fluid force equations, e.g. Di Felice (1994), the effect due to the presence of other particles on fluid force calculation cannot be fully considered (Zhu et al, 2007). This is attributed to a reduction in the size of fluid space, and hence a strong localisation effect of fluid velocity gradient. For the applications in which the fluid-particle interactions are crucial, Direct Numerical Simulation (DNS), which is a fine resolution of CFD, has been conducted, e.g. Hu (1996) and Pan et al (2002). In principle, for any fluid-particle system, DNS coupled with DEM is able to determine the behaviours of both particle and fluid with high accuracy. However, a large number of particles with fine resolution in the fluid mesh would cause a computational barrier in applying this technique to the real-scale engineering problems.

3.4 Coupled DEM-DFF technique

Compared with CFD, Darcy Fluid Flow (DFF) has a much more simple form with less parameters and variables, i.e. permeability and pressure function, allowing relatively lower computational cost, which makes this technique widely used in geotechnical applications. However, as the technique is limited to flows with low Reynolds number (Charbeneau, 2006), it is only suitable for describing the seepage behaviour far away from the leak.

This technique was firstly proposed by Hakuno and Tarumi (1988) to model the dynamic behaviour of a fully-saturated soil in undrained condition. In DFF, the pore fluid was regarded as an elastic medium without shear resistance. Hence, when the particles were moving, excess pore pressures were generated due to the pore volume changes. The pressures dissipated according to the Darcy's law. In their work, it assumed that an instant equilibrium state was achieved in the end of every calculation cycle. Once the equilibrium pressure gradients were obtained from DFF solver, the values were used in calculating the fluid force in DEM.

However, it is recognised that envisaging pore fluid flow as a dynamic process rather than an instant equilibrium achieves more realistic solutions (Bonilla, 2004). In his work, the effect of pore pressure generation and dissipation were combined together with the aid of Hagen-Poiseuille theory. The pore pressure was obtained by integrating over a time increment.

In the above DEM-DFF implementations, the pore pressures were determined at the pore level, and all individual pores were required to be identified at every calculation cycle. It is computationally expensive in the cases where particles move with large velocities so that pores are continuously created and lost over time. Alternatively, Nakase et al. (1999) simplified the implementation by considering pore pressures at the cell level. In their two-dimensional simulations, the computational domain was divided

into square cells, the dimensions of which were chosen so that there are about 15 circular particles whose centres were located inside the cell. Volume change of pores generating excess pore pressure was thus calculated from the weighted average displacement of particles in the four neighbouring cells. Assumption was made that the pressure calculated at the centre of each cell presents the value throughout the whole cell. A finite difference scheme was used to solve the equations describing pore pressure dissipations. Once pressure dissipated for a time increment, the updated pressure gradients were used to calculate the fluid force applied to particles.

A similar coupled technique was also utilised by Shafipour and Soroush (2008). While in this work, volume changes of pores in each cell were calculated from the strain tensor in cell. Although it may provide better accuracy in comparison with Nakase et al. (1999), it involves more complicated calculations as partial differential equations need to be solved. Furthermore, in addition to the hydrostatic forces exerted on a particle by pore pressure gradient, the drag force is also taken into account.

During the same period, the technique has also been applied to sand production phenomenon in the oil wells (O'Connor et al 1997, Preece et al 1999, Jensen and Preece 2000). Due to the dominate influence of the pressure boundary applied to this problem, the excess pore pressure generated by the pore volume change can be neglected, and therefore the DFF implementation is largely simplified.

For the leakage-soil interaction in this thesis, as hydraulic radius is adopted to provide flow channels (see Section 3.4.2), the generation of the excess pore pressure is mainly contributed by the flow boundary conditions, rather than the deformation of the pore size. For this reason, a simple DFF model, which is based on the Laplace equation, is employed.

3.5 Coupled DEM-SPH technique

Smoothed Particle Hydrodynamics (SPH) was first proposed by Lucy (1977) and Gingold and Monaghan (1977) as one of the macroscopic fluid applications of the Mesh-free Particle Methods (MPMs), in which the fluid system is described by a finite number of discrete particles without the Eulerian grid. This method is based on the Lagrangian philosophy, using a single particle representing a part of the fluid with a kernel function (Li and Liu, 2007). Similar to DEM, this technique can be easily employed in the problems with discontinuity or large displacements, such as free-surface or moving boundary problems in fluid simulations. This feature makes SPH coupled with DEM a special tool in simulating solid-liquid mixture in some geotechnical problems (Li et al., 2007; and Morris and Johnson, 2009).

However, as pointed out by Alejandro (2009), SPH is computationally more expensive and provides lower accuracy compared with other grid-based methods. Furthermore,

due to the Lagrangian description of the fluid particles, the viscosity and turbulence models would be difficult to apply. Although some progress has been achieved by adding Large Eddy Simulations (LES) into SPH (Rogers and Dalrymple, 2004), the implementation is still very complicated. Therefore, in most SPH applications, only artificial viscosity is used and laminar flows are modelled (Alejandro, 2009). For simulating the flows of the leaking fluid in this thesis, SPH is regarded not suitable because the flow near the leak is expected to be turbulent.

3.6 Lattice Boltzmann Method

The Lattice Boltzmann Method (LBM), which was originated from the Lattice Gas (LG) automata (Succi, 2001), was firstly used for studying the shock-wave structures (Inamuro and Sturtevant, 1990). LBM regards fluid as packets of micro-particles, of which the distribution is controlled by the discretised Boltzmann equation. Rather than solving the NS equations as the conventional CFD techniques do, the LBM equation is solved explicitly and only local calculations and data streaming are involved during computations. Although the LBM equation was derived based on the first order finite difference, it was found to be able to recover the second order NS equations (Chen and Doolen, 1998).

LBM has been proved to be very efficient in fluid modelling. A comparison was

conducted between LBM and CFD on homogeneous isotropic turbulence simulation with Reynolds number up to 25,500 (Satofuka and Nishioka, 1999). With the same grid size and approximately same accuracy being achieved, LBM required less than half CPU time of CFD. Furthermore, as suggested by Kwon and Hosoglu (2008), solid boundary conditions can be easily applied to LBM, which makes a detailed description of fluid-particle interaction easily achieved. These features attracted interests of many researchers in the development and promotion of the method.

Among the literatures, contributions have been made to extend the application of LBM, including: incorporating a turbulence model into LBM (e.g. Eggel, 1996); modelling of multiphase flow (e.g. Premnath et al, 2005); and simulating the capillary interaction between colloidal particles in multi-component interface (e.g. Onishi et al, 2008). Much progress has also been made in development of various LBM boundary conditions to fit different situations. The bounce-back scheme as the originally fixed wall boundary condition was employed in LBM to apply no-slip wall conditions (Lavall´ee et al, 1991). Later, He at el (1997) proposed the halfway bounce-back scheme, which doubled the accuracy of the original one. The moving boundary condition is achieved by Immersed Moving Boundary (IMB) scheme, which was proposed by Noble and Torczynski in 1998. For specified velocity and pressure boundary conditions, the Zou-He boundary scheme was proposed based on the philosophy of the bounce-back of non-equilibrium distribution (Zou and He, 1997). And on the computational issues, researchers have

devoted to improve the numerical stability or efficiency by proposing the multi-time relaxation LBM (Lallemand and Luo, 2000); the implicit LBM scheme (Krafczyk, 2001); and the locally refined LBM meshes (Filippova and HÄanel, 1998).

3.7 Coupled DEM-LBM technique

Since firstly introduced by Cook et al (2000) in a two-dimensional particle-laden fluid simulation, DEM-LBM has stimulated more and more interest for researchers intending to simulate a fluid-particle system with a detailed description rather than a macroscopic one at the cell level. The validation and evaluation of DEM-LBM has been given in the recent years, indicating this technique is very promising to provide good performance of simulations. Owen et al (2011) demonstrated its capability in quantitative descriptions of fluid-particle interactions by comparing the results from a sequence of numerical tests with the empirical or experimental data. The tests studied the drag coefficients in a 2D flow through a circular cylinder and a square cylinder, as well as a 3D flow through a particle assembly, respectively. The investigations on the velocity field and hydrodynamic torque in a 2D cylindrical Couette flow were also conducted. Furthermore, Mansouri et al (2009) and Feng et al (2010) carried out three-dimensional validations separately against experimental data, and good agreements were obtained by both.

In the two-dimensional simulations conducted by Cook et al (2000, 2004), the fluid flows were limited to low Reynolds numbers. Therefore, Feng et al (2007) and Han et al (2007a) incorporated a simple LES model into the coupled technique and provided a way to simulate fluid-particle systems with a high Reynolds number. In their work, a sub-cycle time scheme was also proposed regarding stability issues. In the same year, irregular shaped particles were employed in the DEM-LBM simulations (Han et al, 2007b), which implies the interlock between particles could be taken into account. However, for the two-dimensional simulations, a physically unrealistic situation appears in the form of no connected paths for fluid flow through a densely-packed material. Boutt et al (2007) adopted a special treatment by setting up a hydraulic radius for a solid particle when considering the fluid-particle interactions. Although this treatment may cause subtle difference in the hydrodynamic force calculation, the simulations with this treatment could still give satisfied results for a two-dimensional qualitative analysis. In 2010, Mansouri et al employed DEM-LBM to calculate the permeability of a cemented polydisperse granular material. The simulations of sand production were also performed with this technique by Ohtsuki and Matsuoka (2008) and Ghassemi and Pak (2011). Furthermore, Leonardi et al (2011) incorporated a power law and a Bingham model into the DEM-LBM framework, which extends its applications to the non-Newtonian fluid. Moreover, by adopting the JKR model in the DEM contact mechanics in Cui et al (2012), the mechanical behaviour of apparent cohesion in soil can be modelled with DEM-LBM implemented.

Due to the ease of solid boundary treatment and high computational efficiency in modelling the physical behaviours at a small scale, DEM-LBM has been regarded as a suitable tool to simulate fluid-particle systems with fine meshes. However, for a large-scale problem with the local behaviour required to be captured in detail, such as a soil bed subjected a local leakage, DEM-LBM would encounter with difficulties as a large amount of computational time along with memory space is expected. Hence, it is beneficial to parallelise the DEM-LBM computing as well as to combine it with other techniques which are more suitable for simulating large-scale problems. For this reason, the following two sections present a literature review of the previous works on DEM and LBM parallelisation, and combination of LBM with large-scale modelling techniques.

3.8 Parallelisation of DEM and LBM

The initial idea of DEM parallelisation can be found in Ghaboussi et al (1993), in which the contact detection was carried out simultaneously by a number of processors. In 1995, Schroder carried out regular domain decomposition in a 2D serial code with the Message Passing Interface (MPI) as the communication library, and each processor held a copy of global arrays. However, no report on the parallel performance has been reported. Washington and Meegoda (2003) implemented a paralleled DEM algorithm in a cluster with 512 processing nodes. Although the contact calculations were parallelised,

the particle data were still stored in all processors. Forces and moments were processed globally by the inter-processor communication, which formed the major bottleneck of this algorithm. Not only it caused a great amount of communication, but also the number of particles modelled was limited. Thus, only a speedup of 8.73 times was reported for 512 processor nodes in this work. Later, a great improvement has been achieved by Darmana et al (2006). After domain decomposition, the calculations on each processor were conducted only for a unique sub-domain and synchronizations were only required for the particles at the sub-domain boundaries. In addition, a mirror domain approach was applied in order to keep the load balance in a system with non-uniform spatial distribution. Using 32 processors, the parallel efficiency was around 0.6 for a 270,000-particle system. In 2011, Kafui et al adopted a commercial parallelisation platform 'parawise graph partitioner' to evenly distribute particles between processors, and a 27-times speedup was achieved on 32 processors. On a shared-memory cluster, Shigeto and Sakai (2011) scaled the simulations by multi-thread parallel with OpenMP, a speedup of 5.4 times was achieved on 8 processors.

For LBM parallelisation, the pioneer work was carried out by Satofuka and Nishioka (1999) with strip decomposition, in which the fluid domain was divided into vertical strips or horizontal strips, and the data in each strip was stored and processed by local processor. In their work, a speedup of 14 times was achieved using 16 processors. In order to take the good advantage of the data locality in the LBM algorithm, Schepke et

al (2009) demonstrated that the blocked partitioning could achieve better performance in parallelisation. For the simulation of a fluid flow through a porous medium partly occupied over the domain, modifications were proposed by some researchers, such as Pan et al (2006), Wang et al (2005), in order to provide a balanced load among processors.

As pointed out by Feng et al (2007), due to the explicit and local processes involved in both DEM and LBM, DEM-LBM has the nature of parallelisation. However, there have been very few literatures reporting the parallel work on the overall DEM-LBM computations, in spite of the well-established parallelisation strategies developed for DEM and LBM separately. Therefore, the parallel computing of DEM-LBM is proposed in this thesis (see Chapter 4), which is beneficial to extend its applications to large-scale problems with a local detailed description.

3.9 LBM combined with large-scale modelling techniques

An alternative way to efficiently model the multi-scale behaviour is to combine LBM with other fluid methods which are more suitable for large-scale simulations, such as DFF and CFD. The basic idea is to apply different models at different fluid regions, with data exchange at the interfaces between neighbouring regions by a prescribed exchanging strategy. As such an idea promotes a better computational performance,

researchers have devoted to its development, which is presented as follows.

Generally, it is straight forward to convert density distribution functions in LBM to the macroscopic fluid variables used in DFF and CFD. However, the reverse process is regarded difficult and of key importance in the combination of LBM and other fluid methods. An early attempt can be found on combination between LBM and CFD by Kwon and Hosoglu (2008) in simulating a wave propagation problem. This combination made the simulation of complex geometry involved in fluid-solid interaction conveniently described. In their work, only D2Q5 model was used in LBM aiming to reduce the number of unknown density distribution functions during data conversion, and the unknown distribution functions were simply obtained from the mass conservation equation. Another work is conducted for the combination of LBM and CFD (Chen et al, 2011). In this work, the density distribution function is reconstructed by a more generalised relationship between macroscopic and microscopic variables (Xu et al, 2012). As the above work was published in the writing-up stage, the theories and implementations have not been included in this thesis. However, further consideration of this work would be beneficial in the future. Furthermore, with both the non-equilibrium bounce-back theory and the extrapolation theory, the LBM D2Q9 model was successfully combined with DFF (Li et al, 2010). The details are presented as part of innovations in this thesis (see Section 5.3).

In addition, all the above combinations were performed on the fluid phase only, whereas the presence of a large number of solid particles would trigger necessary treatments in the combination strategy, which is illustrated in Section 5.4.

CHAPTER 4: IMPLEMENTATIONS OF DEM-LBM AND DEM-DFF in *FPS-BHAM*

4.1 Introduction

In FPS-BHAM, only a two-dimensional model has been built up for simulating a fluid-particle system so far. However, the DEM calculation is based on three-dimensional analysis, with the particle motions are restrained within a single plane. When dealing with the fluid-particle interactions, the spherical particles are treated as cylinders with a reduced size (see Section 4.3.2).

The implementations regarding DEM-LBM and DEM-DFF in FPS-BHAM are presented in this chapter, including the implementations of DEM (see Section 4.2), DEM-LBM (see Section 4.3), and DEM-DFF (see Section 4.4), and the modification of gravitational acceleration (see Section 4.5).

4.2 DEM implementation

The DEM implementation in FPS-BHAM can be divided into three parts: the contact detection scheme, the contact force calculations and the contact array storage strategy. The contact force calculations from a well-established code *TRUBAL*, which is based on the contact theories stated in Thornton (1991), are directly transferred into *FPS-BHAM* for this thesis. The contact detection scheme adopted in this thesis is presented in

Section 5.5. In the following section, the contact array is introduced for storing the contact information in the DEM calculations.

A contact array is allocated to store the contact data, which is used in the history-based contact constitutive theory. Once a contact is detected, the following work is conducted. If the contact is a new one, the contact force is calculated directly, and the relevant data is stored in the contact array for further usage. However, if the contact is already stored in this array, which indicates an old contact, the data in the array is required to be extracted for use. After the contact force being calculated, the new data is adopted to replace the old one in the contact array. Otherwise, if a contact is dismissed, the memory shall be set free for reuse. As the data required to be stored in the contact array is of a large amount and the contacts are dynamic, the operations such as contact searching, allocating, and destruction would be time consuming. However, there are few literatures reporting the storage strategy for contact array, with the consideration of parallelisation. Hence, in this section, the array storage strategy used in this thesis is presented.

The structure for storing an individual contact in the array in the memory is shown in Figure 4.1, which is similar to the one used in Cundall and Strack (1978). For a ball-ball contact, the *First Component* stores the smaller index of the particle in contact, while the *Second Component* stores the larger one. For a wall-ball contact, the *First Component* stores the index of the wall and the *Second Component* stores that of the

particle. *Contact Information* stores all data related to the contact force calculations. *Next Contact* stores the address of the next contact in a contact chain, which builds up a link connecting another contact in the array.

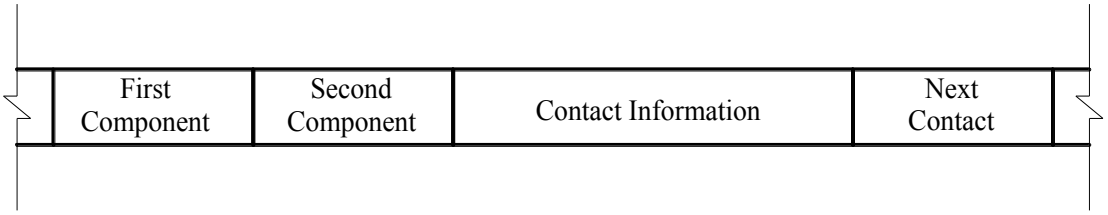


Figure 4.1: Structure of a contact array

Once the contact array is allocated, many contact chains are set up in this array. All contacts in a single chain share the same *First Component*. The address of the first contact in the chain is denoted as *Entrance*, while the *Next Contact* of the end of the chain is denoted as -1. By accessing *Entrance* of each chain, all contacts in the chain can be traced. In addition, an *Empty Contact Chain* is set up to connect all free spaces in the contact array (see Figure 4.2).

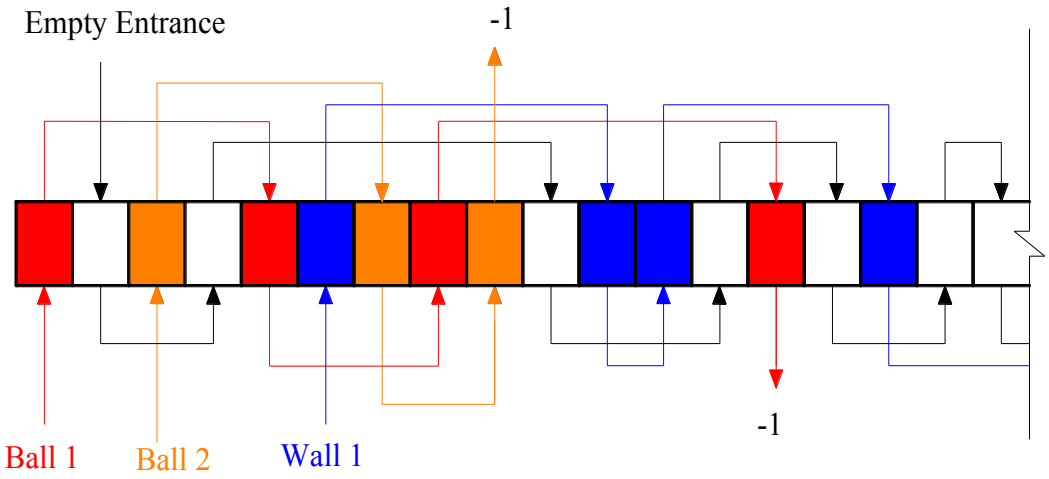


Figure 4.2: Contact chains in contact array

The contact searching in the array can be easily performed by tracing the chain of which the bounded *First Component* is that of the contact. In order to reduce duplicated searching through a single chain, a new chain is set up correspondingly. If the particle is found in the original chain, this contact is removed from it and recorded in the new chain. In this way, the number of the contacts in the original chain is reduced, and hence the contact searching becomes faster. If the particle is not found in the chain, a new space is allocated from the *Empty Contact Chain* to store this contact, and the contact is also recorded in the corresponding new chain. At the end of each calculation cycle, all the remaining contacts in the original chain shall be dismissed as they actually no longer exist. Therefore such a chain is destructed by linking it to the *Empty Contact Chain*. All the existing contacts recorded in the new chain are then used for the next calculation cycle.

4.3 DEM-LBM Implementation

4.3.1 Time steps and sub-cycling

Since the DEM and LBM adopt different time steps, special attention is required to maintain an overall stability of the DEM-LBM calculations. The time step used in DEM is chosen to be a value smaller than the critical one, which is determined by (2.40).

While in LBM, the speed of sound which is directly related to the lattice speed through (2.46) is usually quite high (around 1482m/s in 20°C water) compared with fluid velocity. However, employing such a high value in the LBM simulation may lead to an unacceptable computational cost in a large-scale modelling. It is recommended by Han et al (2007a), if the Mach number (*Ma*) calculated using (4.1) is much smaller than 1, the compressibility error is controlled within a reasonable level.

$$Ma = \frac{u_{\max}}{C}, \tag{4.1}$$

where $u_{\max}$ is the maximum fluid velocity in a simulation. Once the lattice speed is determined, the lattice spacing and time step are uniquely related according to (2.46). In general, a simulation with a larger spacing and time step results in a saving in computational cost, but compromises in accuracy. Therefore, in this thesis, the lattice spacing Δh and time step Δt_{LBM} are chosen with comprehensive considerations of both accuracy and computational cost.

As the time step calculated in LBM has been found to be larger than that in DEM, Feng et al (2007) proposed the Sub-cycling time scheme to deal with such a difference. The time step used in DEM, Δt_{DEM}, is adopted as follow:

$$\Delta t_{DEM} = \frac{\Delta t_{LBM}}{N_{subcycle}}, \tag{4.2}$$

where

$$N_{subcycle} = \text{int}[\frac{\Delta t_{LBM}}{\Delta t_{c,DEM}}] + 1. \tag{4.3}$$

In other words, within a single LBM calculation cycle, $N_{subcycle}$ sub-cycles of DEM is computed. In addition, the fluid force applied to particles remains constant during each LBM cycle.

4.3.2 Hydraulic radius

In a two-dimensional simulation, fluid channels through a densely packed soil are blocked, which is not true in the reality. Boutt et al (2007) proposed a hydraulic radius to cope with this problem. For the DEM calculation, the original particle radius is employed in the computations. While dealing with the fluid-particle interactions (see IMB scheme in Section 2.3.2), a reduced value of the real particle size, i.e. the hydraulic radius, is used. In the fluid-particle two-way coupling, a hydraulic radius R_h, rather than the actual radius of a particle R, is adopted for a cylinder to give an equivalent drag force to a sphere,

$$\frac{\mathbf{F}_D(cylinder)}{\mathbf{F}_D(sphere)} = \frac{\frac{1}{2}\rho \mathbf{v}_r{}^2 A(cylinder) C_d(cylinder)}{\frac{1}{2}\rho \mathbf{v}_r{}^2 A(sphere) C_d(sphere)} = \frac{C_d(cylinder)}{C_d(sphere)} \cdot \frac{R_h / R}{0.25\pi} = 1. \quad (4.4)$$

Hence,

$$\frac{R_h}{R} = 0.25\pi \cdot \frac{C_d(sphere)}{C_d(cylinder)}, \quad (4.5)$$

where $\mathbf{F}_D$ denotes the drag force applied to a particle. $\mathbf{v}_r$ is the speed of particle relative to fluid. A is the reference area. C_d is the drag coefficient only related to particle Reynolds number, and is taken as 0.44 according to Zhu et al. (2007). Therefore, the hydraulic radius is adopted as

$$\frac{R_h}{R} = 0.785. \quad (4.6)$$

Hence, fluid channels are generated artificially between particles.

4.3.3 Flow chart of DEM-LBM computations

Figure 4.3 illustrates the flow chart of DEM-LBM computations. Initially, the time steps for both DEM and LBM are determined, and hence a sub-cycling strategy is set up for the DEM-LBM simulation. In each LBM calculation cycles, the soil particles are mapped onto the LBM framework, and the local solid/fluid ratio (see Section 2.3.2) is calculated at each lattice node. Based on the local solid/fluid area ratio, the LBM governing equation with IMB and LES (2.57) is solved, and the fluid force and torque applied to each particle are obtained from (2.59) and (2.60) as well. The fluid force and torque obtained is then regarded to be constant and used in (2.3) and (2.4) during the

DEM calculations followed. While in the DEM sub-cycles, contact detection is performed firstly to find out the particles/walls that might be in contact. This process is shown in Section 5.5. Once the contact has been identified, the contact force calculations are conducted. Afterwards, the calculated force and torque are used for updating the position and rotation of each particle as shown in (2.7) and (2.8). The above procedures are repeated till the end of the simulation.

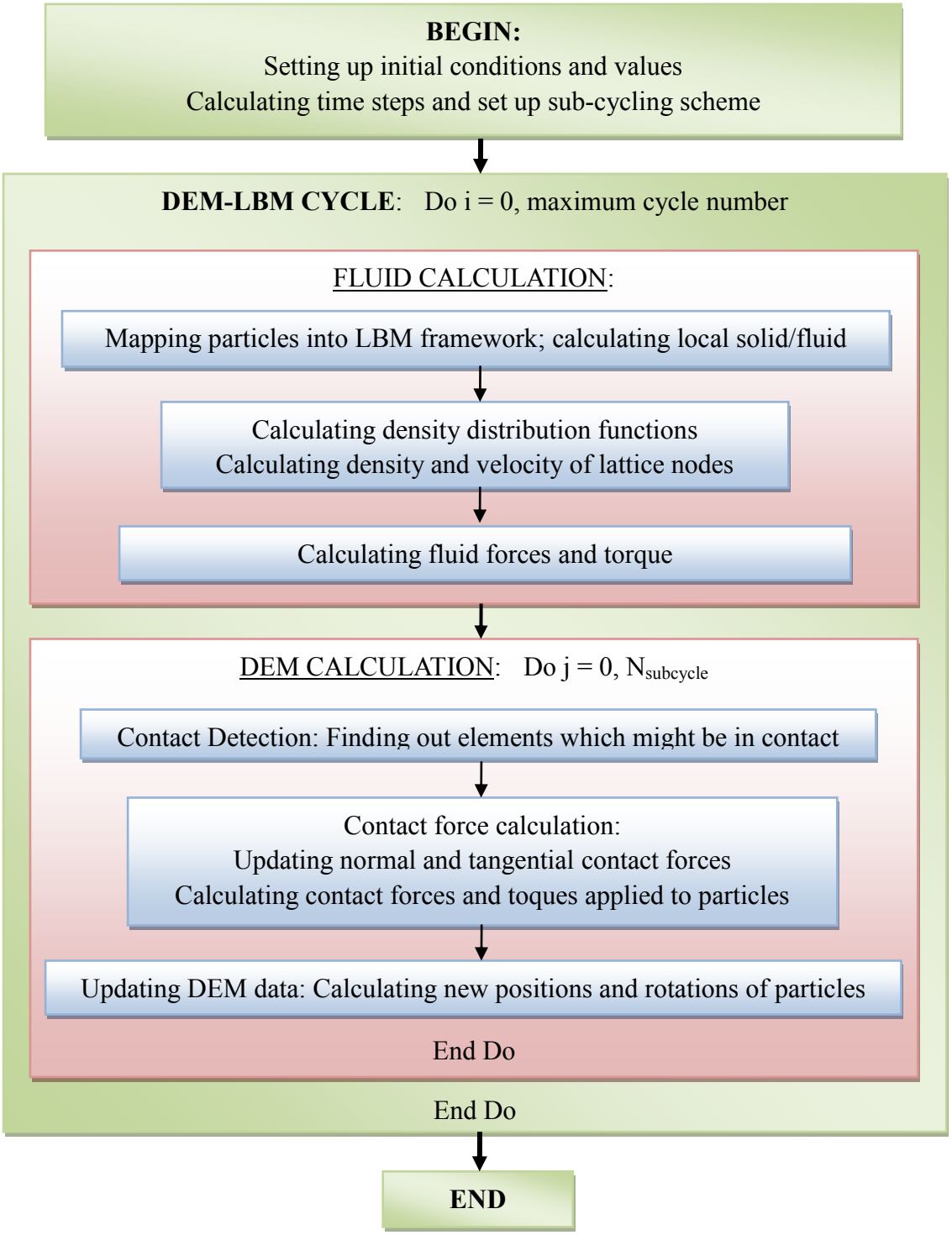


Figure 4.3: Flow chart of DEM-LBM computations

4.4 DEM-DFF Implementation

Since the focus is on the combination of DEM-LBM and DEM-DFF, the permeability is assumed to be homogeneous and can be simplified as $k_{xx} = k_{yy} = k$ and $k_{yx} = k_{xy} = 0$, where k is the macroscopic permeability of the soil. According to Section 2.4.2, it implies that only a one-way coupling between DEM and DFF is implemented in this thesis, as the change in permeability and seepage by particle motions are not taken into account. The DFF governing equation can then be written as

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = 0 . \tag{4.7}$$

Using the second-order central difference approach, we have

$$\frac{p_{x_0-1,y_0} + p_{x_0+1,y_0} - 2p_{x_0,y_0}}{\Delta x^2} + \frac{p_{x_0,y_0-1} + p_{x_0,y_0+1} - 2p_{x_0,y_0}}{\Delta y^2} \cong \frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = 0 , \tag{4.8}$$

where p_{x_0,y_0}, p_{x_0-1,y_0}, p_{x_0+1,y_0}, p_{x_0,y_0-1}, p_{x_0,y_0+1} are the excessive pressures at Node (x_0, y_0) and its four neighbouring nodes. Δx and Δy are the node spacing in the x and y directions. If $\Delta x = \Delta y$,

$$p_{x_0,y_0} = \frac{p_{x_0-1,y_0} + p_{x_0+1,y_0} + p_{x_0,y_0-1} + p_{x_0,y_0+1}}{4} . \tag{4.9}$$

In DEM-DFF, the value of permeability used in (2.71) is required to be determined prior to calculations. The value is required to be consistent with that in LBM. A simple way is to carry out a DEM-LBM test on the soil sample under a pressure driven flow. Once the

steady state is reached, i.e. the inlet flow rate is equal to the outlet one, and the permeability can be calculated according to (2.71), and subsequently be applied to the DFF calculations.

4.5 Modification of gravitational acceleration

As the body force of the fluid is not included in both the LBM and DFF governing equations (i.e. (2.57) and (2.66)), the hydrostatic pressure is not taken into account. Hence, in the DEM calculations, the buoyancy force is considered by modifying the gravitational acceleration as (Feng, 2010)

$$g^* = (1 - \rho / \rho_s) g, \tag{4.10}$$

where g and g^* are the original and modified gravitational accelerations, respectively.

CHAPTER 5: PARALLELISATION WITH DOMAIN DECOMPOSITION

5.1 Introduction

As stated in Section 1.3, a domain decomposition scheme for DEM-LBM is required to be proposed in *FPS-BHAM* for simulating the leakage-soil interaction. In this research, blocked partitioning domain decomposition is employed in order to facilitate parallel computing of the combined system. Full detail of the parallelisation strategy is explained in this chapter. Section 5.2 shows the domain decomposition scheme used and the sub-domain partitioning. The parallelised fluid and DEM calculations are presented in Sections 5.3 and 5.4, respectively. Two related computing issues, i.e. the DEM contact detection algorithm, and the modification to the fluid force calculation are shown in Section 5.5 and Section 5.6, respectively. They are followed by a flow chart of the whole parallel computing process in *FPS-BHAM*, as shown in Section 5.7. The performance of proposed domain decomposition scheme is evaluated in Section 7.3

5.2 Domain decomposition

In domain decomposition, data of each sub-domain is expected to be stored within the local memory of corresponding processors. The calculations in each processor can be carried out independently without interfering with one another frequently. Hence, the time spent on communication among processors is kept at the minimum level. Working

load shall be balanced for each processor to avoid idle waiting of processors.

Depending on computing resources, one processor may be occupied by a single sub-domain or be shared by several sub-domains. For the former case, the fluid data array, the ball data array, and the contact array are set up for each sub-domain to store local data, which is illustrated in this chapter. For the latter case, the ball and contact arrays can be set up for each processor. In such a case, less data exchange is required, which leads to an easier implementation.

Combination of DEM-LBM and DEM-DFF could lead to satisfactory performance in terms of accuracy and computational cost, which is an important feature of this research. Therefore, the blocked partitioning domain decomposition scheme (Schepke et al, 2009) is adopted. It provides an efficient way of applying two different numerical techniques in appropriate areas, i.e. applying DEM-LBM to the area with intensive fluid-particle interactions while DEM-DFF to the area with low Reynolds number and less fluid induced particle movement. Both the fluid and DEM calculations share the same decomposition scheme, so that there would be no extra data communication among processors when considering fluid-particle interactions.

One fact should be noted beforehand that CPU time required for DEM-LBM and DEM-DFF differs widely, which is demonstrated in Section 7.4.2. In order to balance

the working load, DEM-LBM and DEM-DFF sub-domains are distributed in the way that the number of each type of sub-domains is almost identical in each processor. For this research, the number of sub-domains for each technique is fixed while a variable number and size of sub-domain can be considered in the future. For instance, as shown in Figure 5.1, the domain is divided into 4×4 sub-domains, with 8 DEM-LBM and 8 DEM-DFF sub-domains. If 4 processors are used, each one shall contain 2 DEM-LBM and 2 DEM-DFF sub-domains, respectively. If 3 processors are adopted, it is impossible to evenly distribute these sub-domains. In such a case, the optimum distribution shall be 3 DEM-LBM and 3 DEM-DFF sub-domains in the first two processors.

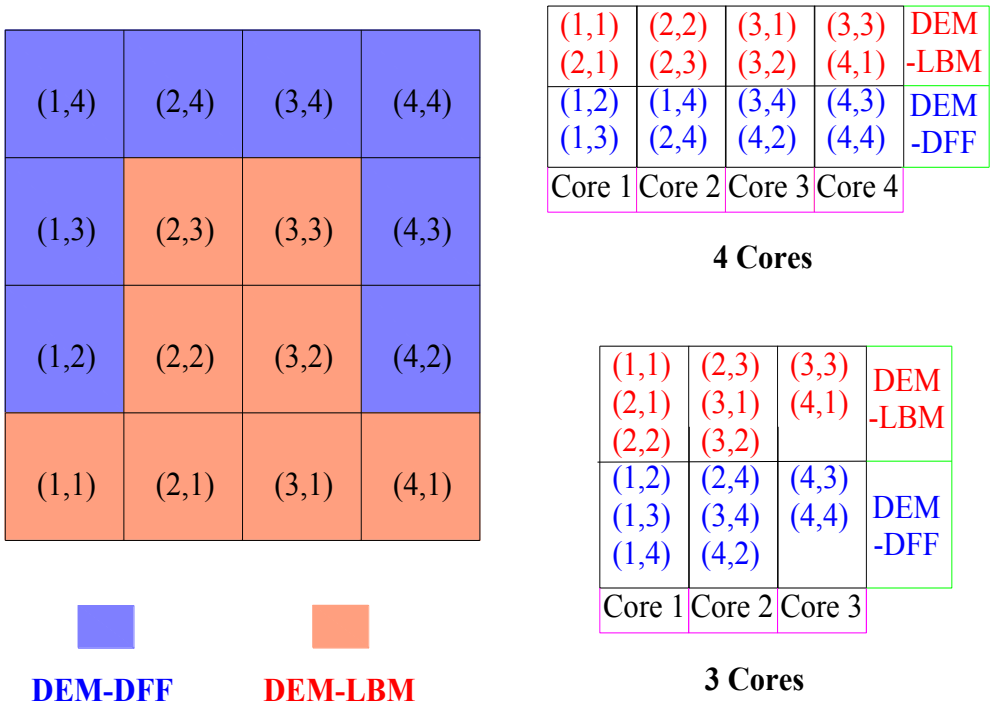


Figure 5.1: Sub-domain allocation

5.3 Parallelised fluid calculation

During a single calculation cycle, the governing equation for the fluid field in each sub-domain (i.e., (2.57) in DEM-LBM sub-domains or (2.66) in DEM-DFF sub-domains) is solved independently with its corresponding boundary conditions. At the interfaces, it is necessary to guarantee the continuity conditions:

$$\rho_{interface,sub-domain\_A} = \rho_{interface,sub-domain\_B}, \tag{5.1}$$

$$\mathbf{u}_{interface,sub-domain\_A} = \mathbf{u}_{interface,sub-domain\_B}, \tag{5.2}$$

$$p_{interface,sub-domain\_A} = p_{interface,sub-domain\_B}, \tag{5.3}$$

where ρ, **u**, p are the density, velocity, and pressure of the fluid, respectively.

Virtual nodes are set up as the sub-domain boundary nodes, which overlap the corresponding inner nodes in a neighbouring sub-domain (see Figure 5.2). Data at virtual nodes can be obtained from calculations in the corresponding inner nodes from the neighbouring sub-domains, and are used for applying boundary conditions. Between sub-domains with a same numerical technique applied, only data exchange by assignment operation is necessary (see Figure 5.3). However, for sub-domains with different numerical techniques applied, it is not straightforward since different data types are used. In Chapter 6, implementations applied to the combination strategy between DEM-LBM and DEM-DFF will be presented in detail.

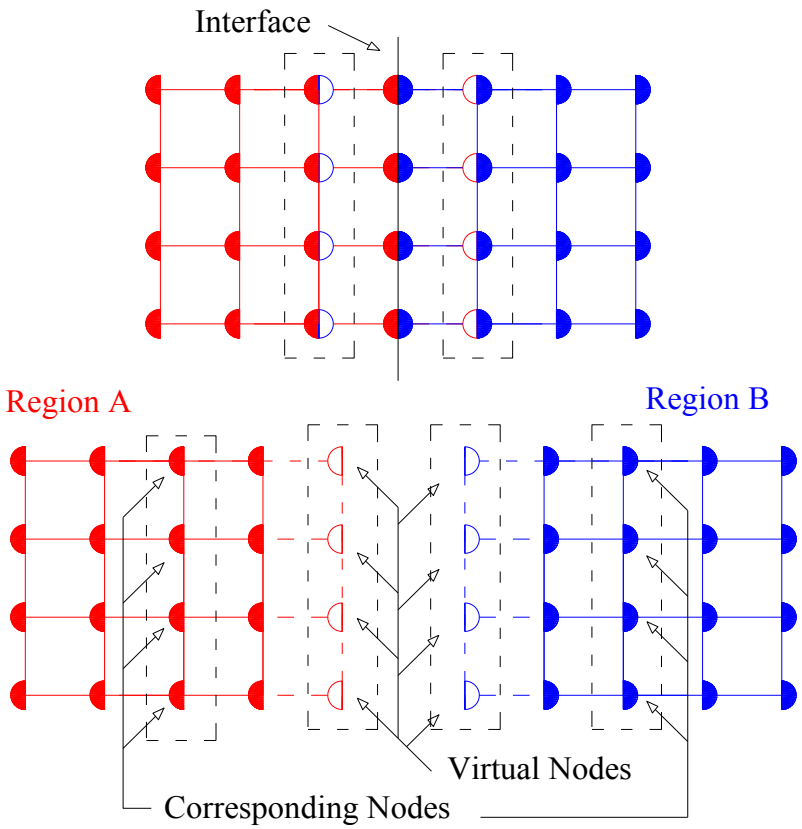


Figure 5.2: Virtual nodes

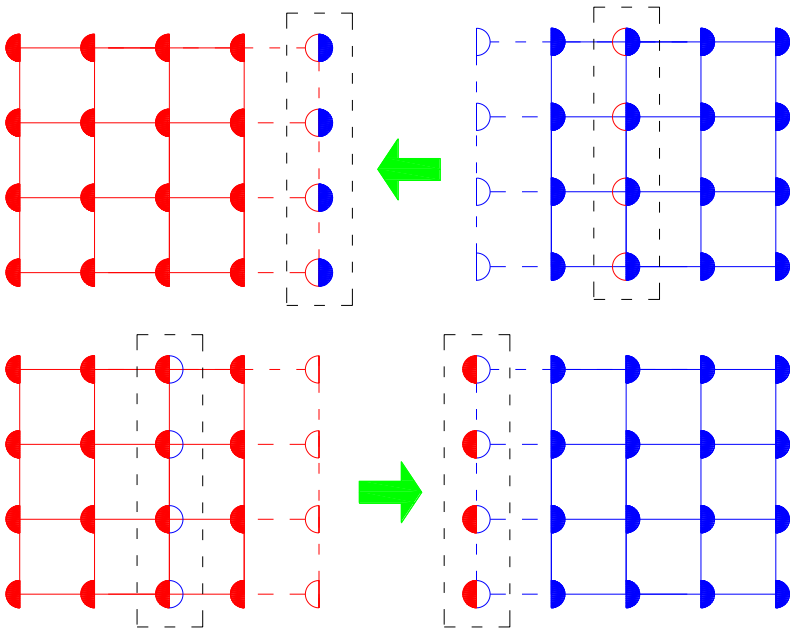


Figure 5.3: Fluid data exchange

If two sub-domains belong to different processors, message passing is required to facilitate communication. There are many parallel programming libraries that provide message passing option. The Message Passing Interface Standard (MPI) (Blaise, 2012a) for distributed memory systems, and OpenMP (Blaise, 2012b) for shared memory systems are widely used nowadays. Since *FPS-BHAM* has been implemented on the distributed memory cluster BlueBEAR (Reed, 2007) in the University of Birmingham, MPI is adopted for the data communication among processors.

MPI is a message passing library standard based on the consensus of the MPI Forum, and the detailed documentation is available at Blaise (2012a). In this thesis only point-to-point communication strategy is employed, in which one processor sends data to only one destination processor with a single command. The implementation is briefly introduced in this section.

The standard point-to-point communication in MPI is a two-way message passing, in which both the sender and receiver call MPI routines for data communication. Two commonly-used types are blocking communication and non-blocking communication. In the blocking communication, neither the sender nor the receiver is continuously processing until data communication is completed. However, if two processors are sending data to each other at the same time, both processors will wait for the partner to receive the data, which would form a deadlock. Although it can be avoided by

programming with a special attention, this is still a challenge for the situation where multiple communications exist (Blaise, 2012b).

To avoid this case, non-blocking communication is employed in this thesis. By this communication type, processors are allowed to send and receive data at the same time. The routine for sending a message is *MPI_Isend* and the one for receiving a message is *MPI_Irecv* (Blaise, 2012b):

MPI_Isend (buf., count., datatype., dest., tag., comm., request., ierr.) ,

MPI_Irecv (buf., count., datatype., source., tag., comm., request., ierr.) ,

buf. is the address of the first element of data. *count.* and *datatype.* are the number and data type of all the elements, respectively. *dest.* and *source.* are the IDs of the processors receiving and sending data, respectively. *tag.* works as a tag to distinguish among various messages. *tag.* is set as (5.4), so messages can be identified during multiple communications.

$$tag. = 8 * i_{sub-domain} + bl \,, \tag{5.4}$$

where $i_{sub-domain}$ is the sub-domain index in the destination/source processor and bl is the boundary location, (There are eight boundary locations: left, right, upper, lower, upper left, upper right, lower left, lower right, which are denoted by 0 to 7, respectively) . *comm.* represents the group of processors with communication. *request.* is the request to show a successful communication. *ierr.* indicates the type of error happened during communication. For multiple non-blocking communications, *MPI_wait*

(*request., status.*) is employed for a final check of the completeness of all communications.

The type and location of each sub-domain are stored in all processors, therefore all processors know the type of data to send and receive as well as the ID of the processor to communicate. After fluid calculation in each cycle, the routine *MPI_Barrier (comm., ierr.*) is invoked to synchronise all processors before data communications. Then, *MPI_Isend* is employed to send the data to the corresponding destination processors. At the same time, each processor prepares to receive data from the corresponding source processors by the command *MPI_Irecv*.

5.4 Parallelised DEM calculation

Particles are initially mapped onto different sub-domains according to the location of their centres. Each of them is expected to be stored and calculated only in the local sub-domain. However, a particle straddling different sub-domains might contact particles in the neighbouring sub-domains, and such contacts would have to be considered. In order to identify these contacts, overlapping areas with the width identical to the maximum particle size are set around each sub-domain. A criterion is set up that a particle is considered for a particular sub-domain during DEM calculation if its location is either inside this sub-domain or in the overlapping areas of this sub-domain. This is

because for the particles belong to the neighbouring sub-domains, only those located in the overlapping areas of the local sub-domain have the chance to touch the local particles.

However, this treatment can lead to duplicated consideration of contact force. For instance, there are three particles in contact as shown in Figure 5.4, Particles 1 and 3 are located in the overlapping area of Sub-domain B, and Particle 2 is located in the overlapping area of Sub-domain A. Contacts exist between particles 1 and 3 (designated as C13) as well as particles 1 and 2 (designated as C12). As all the three particles are taken into consideration in both sub-domains, the two contacts are detected twice which could lead to duplicated consideration of the contact force. In order to avoid this, the contact force calculated is only applied to the particles located inside a particular sub-domain. In this case, contact force in C12 is applied to Particle 1 in Sub-domain A, and to Particle 2 in Sub-domain B. Although C13 can be detected in both sub-domains, the calculation of the corresponding contact force will not be considered in sub-domain B, as the contact force is only used in Sub-domain A.

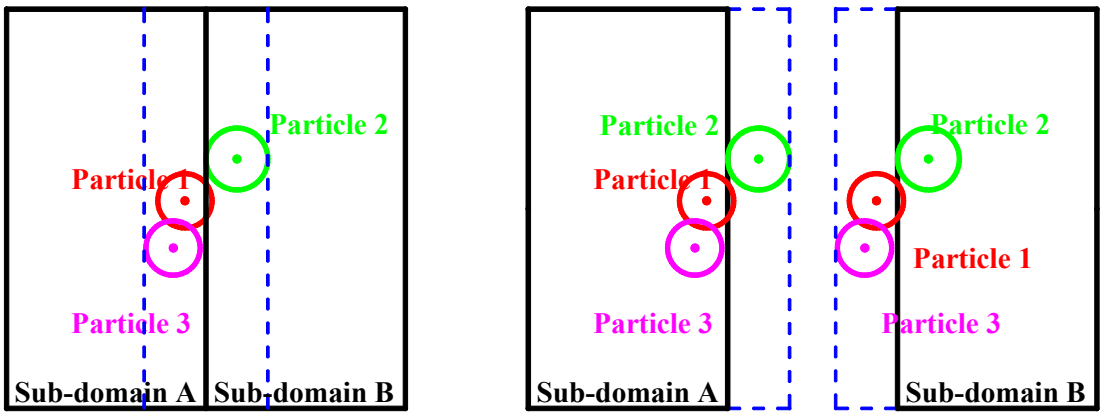


Figure 5.4: Particles in overlapping areas

In this way, each particle inside a particular sub-domain obtains the correct value of contact force. Calculations followed lead to the correct solution to the particle locations. Before the data of these particles are ready for the next calculation cycle, some of it shall be transferred to replace the values in adjacent sub-domains, including:

(i) particles moved out of the local sub-domain, or

(ii) particles located in the overlapping areas of the adjacent sub-domains.

Similar with fluid parallelisation, the data communication among processors is also required if the sub-domains involved in data exchanges are in different processors. However, the number of particles of which the data needs to be transferred cannot be determined by the destination processor in advance. Hence, the data communication is conducted in two steps. Firstly, the destination processor is informed of the amount of data. Accordingly, a dynamic array can be set up as a temporary place for storing the

data to be received. Afterwards, the data is transferred and stored in the place prepared. Once all communications are finished, the data stored in the temporary places can be unpacked for use.

The contact array of a sub-domain with particle immigration needs to be updated as well. As stated in Section 4.2, contact array is quite complicated. The addresses of contacts and chains of contact array may suffer from great changes when receiving new contact data from other sub-domains. A simple implementation can be performed in the following way to avoid the changes: a contact is allocated in the contact array when its related particles are in the overlapping areas, while the calculation of this contact can be omitted. Once either of the particles enters the sub-domain, the contact information related to this particle shall be updated in this sub-domain. The updating information is sent from the source sub-domain. Based on this principle, all the contacts which are required to be updated or sent are checked over sub-domains after data of particles are transferred.

5.5 Contact detection with blocked partitioning domain decomposition

In DEM simulations with a great number of contacts involved, contact detection usually consumes a considerable amount of time. Many researchers have developed different schemes for contact detection in order to improve the DEM computational efficiency,

and a brief literature review can be found in Munjiza (2004). In this section, all particles are assumed to be with similar size. In the case where this assumption fails, additional memory space is needed and modifications are required, which can also be found in Munjiza (2004).

Among various detection schemes, Direct Mapping Scheme (DMS) (Munjiza, 2004), which is a linear one, is capable of providing the best CPU efficiency. In this method, the domain is divided into boxes. The spacing of a box is equal to the largest particle size, so that each particle can be fitted into a single box. Each particle is mapped onto the corresponding box according to its centre coordinates. A memory space is required for each box to store the ID of the particle occupying the box. Once a particle is mapped, contact detection is carried out. As one particle will be in contact with particles in the neighbouring boxes, contact searching is only performed in the neighbouring boxes. However, in such a scheme, a huge amount of computer memory is required if the dimension of computational domain is relatively large compared with particle size, which adversely limits its applications.

Another scheme called "No Binary Search" (NBS) was proposed by Munjiza and Andrews (1998), in order to reduce the memory cost for DMS. The fundamental philosophy is to divide the whole domain into columns, of which the width equals the maximum particle size. Particles are mapped onto the columns based on their centre

coordinates. Afterwards, contact searching is sequentially performed on each of the two neighbouring columns by DMS until the whole domain is checked. Once the checking between two related columns is finished, the memory used for a former column can be released and used for a new column, so that the memory required for the whole domain is reduced to that for only two columns. However, as blocked partitioning domain decomposition is used (see Section 5.2), such scheme is not suitable for the work in this thesis. This is because a single column is crossing different sub-domains, requiring extra data transfer between different processors. Therefore, in this section, a modified contact detection scheme with blocked partitioning domain decomposition is proposed.

Since the particles have already been mapped onto sub-domains with the overlapping areas in this thesis (see Section 5.4), DMS can be directly executed within each sub-domain with its overlapping areas. With a similar idea as NBS, once the checking within a sub-domain is finished, the memory used can be released and used for a new one. For a LBM fluid sub-domain, the memory cost of each lattice node is at least 10 times more than the memory required for storing a particle index in a box. Furthermore, for the simulations in this thesis, the average size of the box is five times of the size of a lattice spacing in one dimension, which leads to the memory used in the process of mapping can be neglected compared with that used in LBM (i.e. less than $1/(5^2 \times 10) = 1/250$). Thus, this contact detection scheme can be implemented without special consideration of memory cost.

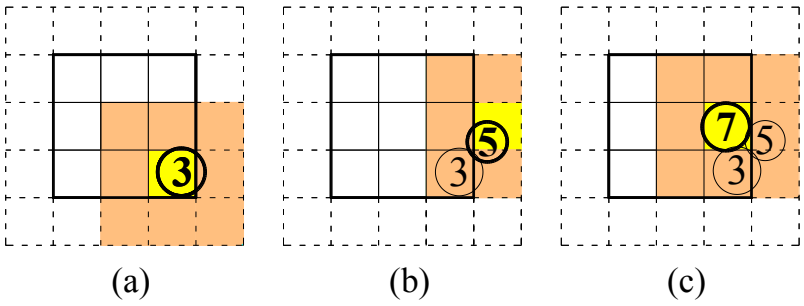


Figure 5.5: Contact detection

As an example shown in Figure 5.5, Particles 3, 5, and 7 are located in a particular sub-domain or in an overlapping area of this sub-domain. Firstly, Particle 3 is mapped onto this sub-domain. Contact searching is performed on the neighbouring boxes and no contact is detected (see Figure 5.5(a)). While after Particle 5 is introduced into the sub-domain, the contact between Particles 3 and 5 is detected (see Figure 5.5(b)). Finally, after Particle 7 is mapped, contacts between 3 and 7, and between 5 and 7 can be found through detection.

It can therefore be concluded that, the computational cost used in the contact detection is linearly related to the number of particles in this scheme. Furthermore, this scheme can be directly implemented without much consideration of memory cost.

5.6 A particle straddling different sub-domains

With the proposed combination strategy, calculation of the fluid force applied to a particle straddling different sub-domains requires special attention. In DEM-DFF, this force is determined by the pressure gradient force and hydrodynamic force on the cell level (see Section 2.4.2). While, in DEM-LBM, the fluid force applied to a single particle are based on the IMB scheme (see Section 2.3.2), which follows the philosophy of momentum conservation between the solid particle and the surrounding microscopic fluid particles. The different treatments of fluid force between DFF and LBM may lead to a problem which is discussed in Section 6.3.2. In *FPS-BHAM*, if the particle is covering both DEM-DFF and DEM-LBM sub-domains, the fluid force is determined by averaging the values calculated from those DEM-DFF sub-domains.

5.7 Flow chart of computations with parallelisation

Figure 5.6 shows the flow chart of the complete computational strategy with consideration of parallelisation. In the beginning, the domain is divided into different sub-domains with different numerical techniques applied according to simulation arrangement, so that the computational load can be evenly distributed to different processors, as shown in Section 5.2. Particles are then mapped onto the corresponding sub-domains according to their positions. During each computing cycle, the calculations of each sub-domain are performed independently in the corresponding processors. The

main structure of the loop in a single computing cycle is followed, which is illustrated in Section 4.3.3, but with data exchanges among sub-domains in the following situations: a) fluid data exchange at sub-domain interfaces to apply boundary conditions for fluid calculation (see Section 5.3); b) modification of the fluid force applied to a particle straddling sub-domains with both DEM-LBM and DEM-DFF (see Section 5.6); and c) exchange of particle and contact information among sub-domains after each DEM sub-cycle (see Section 5.4). Before each data exchange, synchronisation between processors is required to guarantee an accurate and successful data transfer.

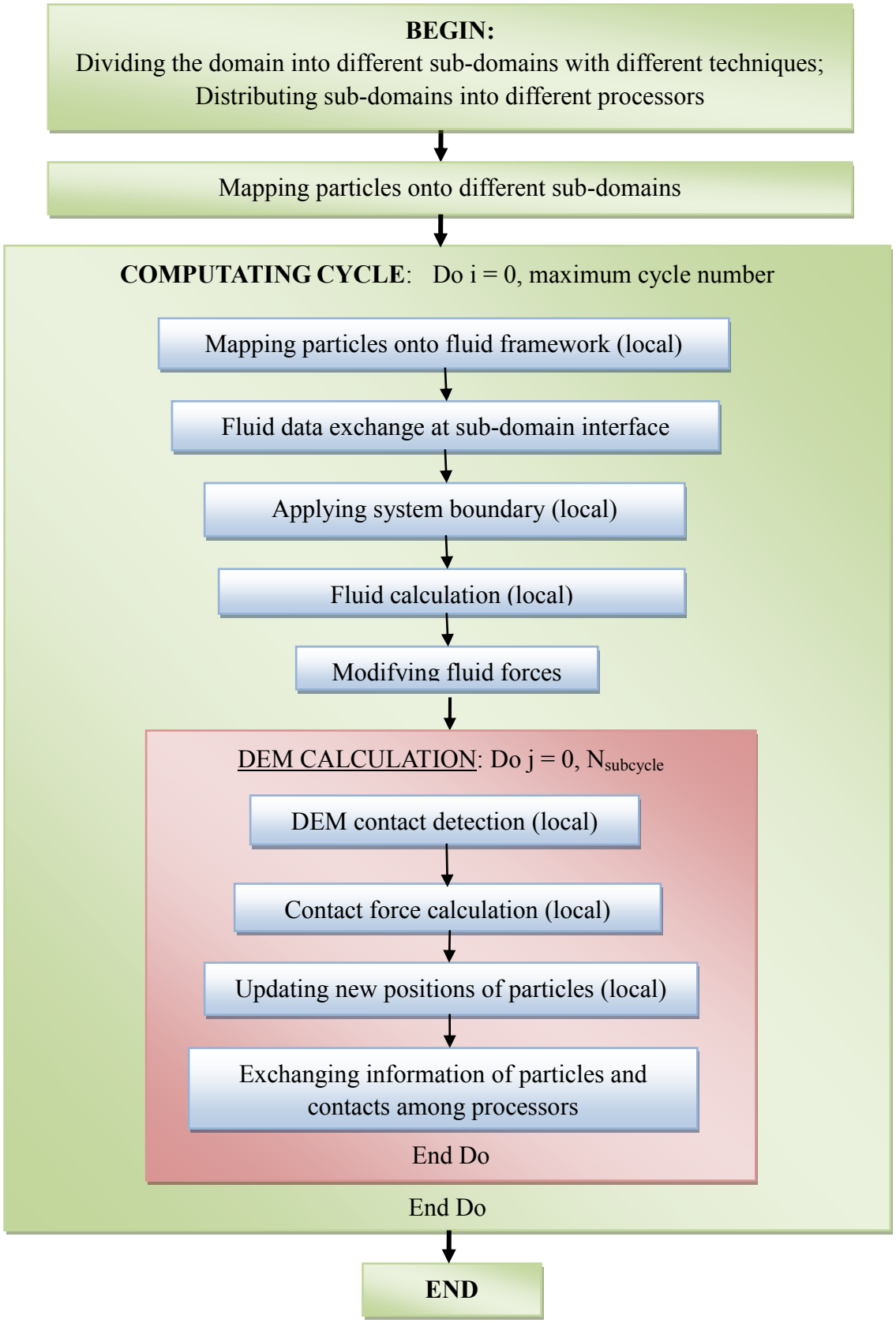


Figure 5.6: Flow chart of computations with parallelisation

CHAPTER 6: COMBINATION OF DEM-LBM AND DEM-DFF

6.1 Introduction

As the blocked partitioning domain decomposition is adopted for parallel computing (see Chapter 5), the DEM-LBM and DEM-DFF calculations can be conveniently performed independently on separate sub-domains. However, a difficulty is encountered during data exchange due to different types of variable used in the DEM-DFF and DEM-LBM techniques, as mentioned in Section 5.3. Moreover, different treatments to fluid-particle interactions also require special consideration in dealing with sub-domain interfaces. Therefore, in this chapter, main focus is on solving the problems mentioned above in order to provide an accurate and smooth solution across the sub-domain interfaces. The conversion of variable types between DFF and LBM is then explained in Section 6.2. Thereafter, issues regarding incorporation of the DEM with the combined system are discussed in Section 6.3. The validation of the proposed combination strategy in this chapter can be found in Section 7.4.

6.2 Combination of between DFF and LBM

6.2.1 Introduction

As the DFF calculations involve macroscopic variables while LBM is implemented using microscopic ones, information transfer between the sub-domains of the different

numerical techniques applied is not straightforward. Thus, it is necessary to develop an efficient two-way conversion scheme between different data types.

The relationship between the density distribution functions and macroscopic variables is presented in (2.53)-(2.55). By a direct summation, conversion from density distribution functions in LBM to pressure in DFF can be achieved. However, there is no such direct way to converse the data inversely, as the number of independent unknowns is greater than that of the conservation equations. In order to close the system, extra constraints are required.

An example is taken for the case of a node at the left boundary of a LBM sub-domain, with a DFF sub-domain adjacent to it on the left, as illustrated in Figure 6.1.

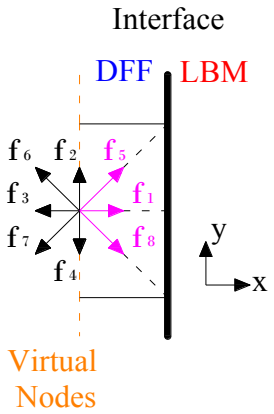


Figure 6.1: Interface between LBM and DFF

After the streaming step, the density distribution functions f_0, f_2, f_3, f_4, f_6 and f_7 can be determined while f_1, f_5 and f_8 are unknown. The key issue is to obtain the values of f_1, f_5 and f_8 so as to meet the conservation requirements. However, based on (2.53)-(2.55), only two equations can be derived, i.e., (6.1) and (6.2). Therefore, one more constraint is required, which is discussed in the following sub-sections.

$$f_5 - f_8 = \rho u_y - f_2 + f_4 - f_6 + f_7, \tag{6.1}$$

$$f_1 + f_5 + f_8 = \rho - (f_0 + f_2 + f_3 + f_4 + f_6 + f_7), \tag{6.2}$$

where ρ and u_y are the macroscopic density and velocity in the y direction, respectively.

6.2.2 Extrapolation scheme

In 1996, Chen et al proposed the extrapolation scheme, in which it is assumed that the density distribution functions are linearly distributed within the domain.

The illustration is presented for Node E in Figure 6.2, which is based on the same node as shown above. Considering the ease of implementation, it is assumed that there is one additional layer of nodes (A-D-G) beyond the virtual ones (see Figure 6.2(a)). For each calculation cycle before the streaming operation, the distribution functions at the nodes on A-D-G can be calculated using a second order extrapolation, based on the values of the distribution functions at the nodes on B-E-H and C-F-I,

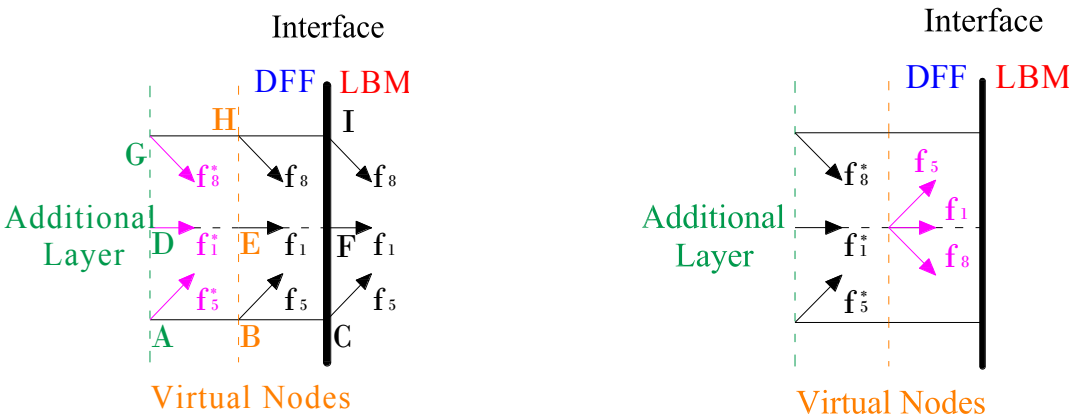
$$f_{1-D}^{*} - f_{1-E} = f_{1-E} - f_{1-F}, \tag{6.3}$$

$$f^*_{5-A} - f_{5-B} = f_{5-B} - f_{5-C}, \tag{6.4}$$

$$f^*_{8-G} - f_{8-H} = f_{8-H} - f_{8-I}, \tag{6.5}$$

where f_i^* are predetermined values ready to be streamed to Node E in the figure.

After streaming (see Figure 6.2(b)), the distribution functions at the virtual nodes can be enforced by the equilibrium distribution functions (2.48) in the collision step, with the density and velocity values obtained from DFF sub-domain through (2.55) and (2.71).



(a) Before streaming (b) After streaming

Figure 6.2: Extrapolation scheme

In this way, the distribution functions at Node E can be uniquely determined. However, as reported in Section 7.4.2, the extrapolation scheme cannot be applied to the simulations with a low relaxation number. In order to overcome this limitation, the non-equilibrium bounce-back scheme is investigated in the following sub-section.

6.2.3 Non-equilibrium bounce-back scheme

A non-equilibrium bounce-back scheme for LBM model was proposed by Zou and He in 1997. It assumed that the transverse velocity can be neglected at the interface where density distribution functions have to be obtained from the specified macroscopic variables. This scheme is based on the idea that the non-equilibrium part of the density distribution is bounced back only in the direction normal to the interface. Therefore, in the previous case, there is

$$f_1 - f_1^{eq} = f_3 - f_3^{eq}, \tag{6.6}$$

where f_i^{eq} is the equilibrium density distribution function in the i^{th} direction. Combining (6.1), (6.2) and (6.6), f_1, f_5 and f_8 can be obtained.

6.3 Incorporation of DEM with the combined system

6.3.1 Fluid density at the particle-covered nodes

As IMB is adopted in the LBM framework (see Section 2.3.2), the information of the fluid nodes fully covered by particles is not obtained explicitly, for instance, Node A in Figure 6.3. However, such information is required at the interface between the DFF and LBM sub-domains, as it is required to be transferred across the interface. In this work, the problem is solved by determining the fluid density at the particle-covered nodes from those at the neighbouring fluid nodes using a smoothing function.

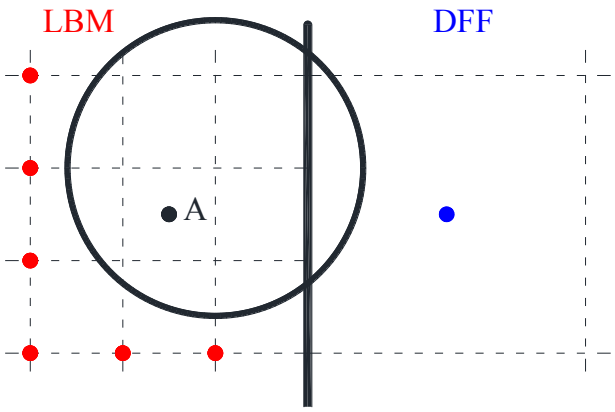


Figure 6.3: A node fully covered by particle

The smoothing function is an approximating function based on interpolation of a set of values in an arbitrary manner, and it is widely used in mesh-free methods, such as Smoothed Particle Hydrodynamics (SPH) (Liu and Liu, 2010). The values can be those at the neighbouring nodes within a predefined cut-off distance. Many researchers have considered the smoothing function construction, and special attention has been paid to the pattern of interpolation and definition of the cut-off distance of the influencing area. Using the smoothing function, the density at a particle-covered node ρ_p can be obtained by

$$\rho_p = \frac{\Sigma(\omega_i \cdot \rho_{fi})}{\Sigma\omega_i}, \tag{6.7}$$

where ρ_{fi} is the density at the i^{th} fluid node within the cut-off distance, and ω_i is its corresponding weighting factor.

In this thesis, the bell-shaped function (Lucy, 1977) is employed, and it is expressed as

$$\omega_i = (1+3\frac{r_i}{h})(1-\frac{r_i}{h})^3, \tag{6.8}$$

where r_i is the distance between the particle-covered node and the i^{th} fluid node. h is the cut-off distance of the influencing area, which is chosen as the maximum particle diameter in simulations.

6.3.2 Fluid force treatment

In a DEM-LBM sub-domain, the particles are treated as moving solid boundaries, and fluid force applied to a single particle is calculated based on momentum conservation at the microscopic scale. Accurate results can only be obtained when calculations are performed on all the lattice nodes surrounding the particle. While in a DEM-DFF sub-domain, the particles are submerged in large fluid cells. The fluid force is obtained from (2.70) at the cell scale. Even if part of a particle is outside the sub-domain, the force applied to the particle can still be estimated with the aid of extrapolation, which is implemented on the fluid pressure and velocity. Therefore, if a particle is located at the LBM/DFF interface, as shown in Figure 6.4, the fluid force adopts the value which is calculated in the DFF sub-domain.

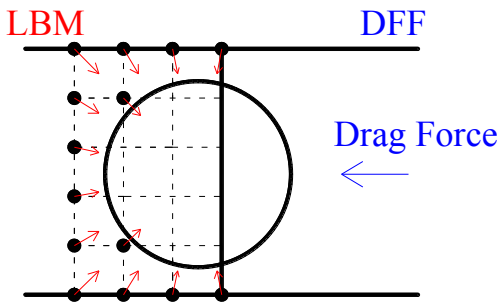


Figure 6.4: Fluid force applied to a particle at the LBM/DFF interface

CHAPTER 7: VERIFICATION AND VALIDATION

7.1 Introduction

In this chapter, verification and validation are carried out to evaluate the algorithms and implementation presented in Chapters 4-6. Since this research began with the modelling leakage-soil interaction using DEM-LBM, this technique is firstly tested in Section 7.2, including: a) whether or not DEM-LBM has been correctly implemented in *FPS-BHAM*; and b) the capability of DEM-LBM in modelling the leakage-soil interaction in a pipe leakage problem. For the parallel computing strategy proposed in Chapter 5, the verification can be easily achieved by comparing the numerical results with and without parallelisation. Therefore, in Section 7.3, the main focus is on the assessment of parallel performance with the blocked partitioning domain decomposition. Finally, in Section 7.4, the validation is conducted on the combination strategy between DEM-LBM and DEM-DFF. The following objectives shall be accomplished during this section: a) to ensure the DEM-DFF technique has been correctly implemented in *FPS-BHAM*; b) to check whether the proposed combined system works properly; and c) to evaluate the benefit obtained from the combination strategy.

7.2 Verification of DEM-LBM in *FPS-BHAM*

The work carried out in this section is used to verify DEM-LBM implementation and to evaluate the capability of DEM-LBM in modelling leakage-soil interaction. Since DEM-LBM mainly consists of DEM, LBM and the two-way coupling between DEM and LBM (i.e. IMB), the work in this section is presented in the following sub-sections:

1) verification of DEM code,

2) verification of LBM with IMB code, and

3) evaluation of DEM-LBM capability in modelling leakage-soil interaction.

7.2.1 Verification of DEM code

The contact force calculation of the DEM part in *FPS-BHAM* is extracted from the well-developed *TRUBAL* code which is maintained by Colin Thornton and his team. The following comparison studies are carried out to examine if the DEM calculations, especially the contact force calculation subroutine, works properly.

One-Ball Drop Test

As shown in Figure 7.1, a ball with no initial velocity is located 5m above the ground. Due to the body force, the ball dropped freely. After touching the ground, the ball bounces back. All parameters are listed in Table 7.1.

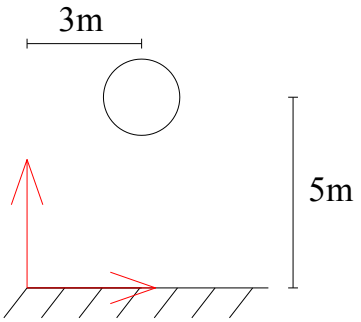


Figure 7.1: One-ball drop test

Body force (N)	(1.0, -9.81, 0.0)
Radius (m)	1.0
Density (kg/m3)	2700
Young's modules (Pa)	6.895×10^{8}
Poisson's ratio	0.3
Friction coefficient	0.3
Global damping	0.01
Ball-Ball contact damping	0.016
Wall-Ball contact damping	0.032
Time step (s)	0.0001

Table 7.1: Parameters of one-ball drop test

Case 1: A non-adhesive ball is used with the elastic contact theory. The simulation is performed on both *TRUBAL* and *FPS-BHAM*. Table 7.2 records the particle location at 1.0s (10,000 DEM calculation cycles), which is a short time after colliding with the wall.

The results are compared with those from *TRUBAL* in Table 7.2 as well.

	TRUBAL	*FPS-BHAM*	Relative difference
X-coordinate	3.478647	3.478688	4×10^{-5}
Y-coordinate	1.503515	1.502916	5×10^{-4}
Rotation	-4.954123×10^{-2}	-4.924805×10^{-2}	5×10^{-5}

Table 7.2: Comparison between the results from *TRUBAL* and *FPS-BHAM* in Case 1

Case 2: An adhesive particle is adopted to collide with the wall. The same set of DEM parameters is used except the particle surface energy value given as 5.0J/m^2. Table 7.3 shows the comparison between the results from different codes.

	TRUBAL	*FPS-BHAM*	Relative difference
X-coordinate	3.478651	3.478722	7×10^{-5}
Y-coordinate	1.499619	1.498504	1×10^{-3}
Rotation	-4.953366×10^{-2}	-4.916298×10^{-2}	6×10^{-5}

Table 7.3: Comparison between the results from *TRUBAL* and *FPS-BHAM* in Case 2

Two-Ball Contact Test

As shown in Figure 7.2, both Ball A and Ball B are with no initial velocity. Ball A is located at 5m above the ground, while Ball B is rest on the ground. In the beginning of

the test, Ball A started to drop freely under gravity. All parameters are listed in Table 7.4. The simulation is performed on both *TRUBAL* and *FPS-BHAM*. Table 7.5 shows the comparison of ball positions and rotations after 10,000 cycles.

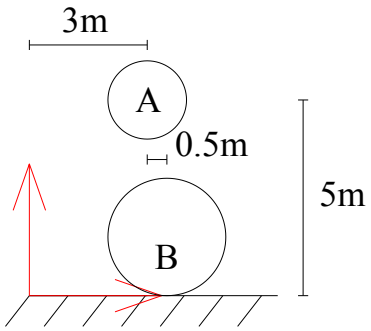


Figure 7.2: Two-ball contact test

Parameters	Ball A	Ball B	Wall
Body force (N)	(0.0, -9.81, 0.0)		
Radius (m)	1.0	1.5	-
Density (kg/m3)	2700	5000	-
Young's modules (Pa)	6.895×10^8	8.895×10^8	10.895×10^8
Poisson's ratio	0.30	0.31	0.32
Friction coefficient	0.3	0.4	0.5
Surface energy (J/m^2)	5.0		
Global damping	0.01		
Ball-Ball contact damping	0.016		
Wall-Ball contact damping	0.032		
Time step (s)	0.0001		

Table 7.4: Parameters of two-ball contact test

		TRUBAL	*FPS-BHAM*	Relative difference
Ball A	X-coordinate	2.377825	2.378652	8×10^{-4}
	Y-coordinate	3.707772	3.699688	8×10^{-3}
	Rotation	0.5343704	0.532251	3×10^{-4}
Ball B	X-coordinate	3.560187	3.563979	4×10^{-3}
	Y-coordinate	1.476991	1.476910	8×10^{-5}
	Rotation	-3.6935519×10^{-2}	-3.489850×10^{-2}	3×10^{-4}

Table 7.5: Comparison between the results from *TRUBAL* and *FPS-BHAM* in two-ball contact test

Since the contact force subroutine used in *TRUBAL* is treated as a black box, and be transferred as a whole, the difference in the contact force calculation may be due to reasons other than the coding within the contact force subroutine and there are a number of possibilities which may cause the difference in the above tests:

(i) Truncation error: In *TRUBAL*, the single precision 'real type' variables are used during calculations. However, in *FPS-BHAM*, the 'double precise type' variables are adopted.

(ii) Procedural difference: In *TRUBAL*, the contact forces are calculated at the end of each DEM cycle, after the particle positions being updated. While in *FPS-BHAM*, the

contact forces are firstly calculated, and then followed by updating the particle positions. Hence, there will be almost a cycle difference between the two codes, which induces subtle difference in the simulation results.

Although differences exist, from Tables 7.2, 7.3, and 7.5, the differences are still considered to be sufficiently small compared with the dimensions of simulation. It indicates that the DEM part works properly with the contact force calculation successfully transferred from the original *TRUBAL* code.

7.2.2 Verification of LBM with IMB code

In Cook et al (2004), the Couette flows are simulated using LBM with IMB. By comparing with the analytical solution, LBM with IMB was proved to be capable of describing fluid behaviour with moving boundaries with acceptable accuracy. In their tests, a laminar flow of a viscous fluid is in a channel between two parallel walls. When one wall is moving relative to the other, the flow is driven by the viscous drag force applied to the fluid. In this section, similar tests, including a plane Couette flow and a cylindrical Couette flow are modelled and the results are compared with analytical solutions, respectively.

Plane Couette Flow

In a plane Couette flow, two plane walls are located at the top and bottom boundaries of

a rectangular domain. Hence, the local solid/fluid ratio (see Section 2.3.2) is 0.5 for all nodes at the interface between fluid and walls. Since the analytical solution is derived based on the assumption of a large aspect ratio, the size of the computational domain is chosen to be 0.3m × 0.05m. The fluid density $\rho = 1000\text{kg/m}^3$, and the kinematic viscosity $\upsilon = 10^{-6}\text{m}^2/\text{s}$. The lattice spacing is 1mm, and the time step $\Delta t_{LBM} = 1\times10^{-4}\text{s}$ which gives the corresponding lattice speed $C = 10\text{m/s}$. Periodic boundary conditions are applied to the left and right sides. Initially, the fluid is at the equilibrium state and the top wall is moving rightwards with a constant velocity $u_0 = 1\text{m/s}$. The analytical solution at the steady state satisfies

$$u = u_0 \cdot \frac{y}{d}, \tag{7.1}$$

where u is the fluid velocity, d is the distance between the two walls, and y is the distance between a particular node to the stationary wall at the bottom.

The global difference is defined as (Cook et al, 2004)

$$E_v = \frac{\sqrt{\sum (u_x' - u_x)^2 + (u_y' - u_y)^2}}{\sqrt{\sum (u_x^2 + u_y^2)}}, \tag{7.2}$$

where u_x and u_y are the velocity components of the analytical solution in the x and y directions, u_x' and u_y' are the values given from the numerical tests. The summation is performed over the entire lattice. At the steady state, the global difference is 1.6% and the velocity distribution from the numerical result is shown in Figure 7.3.

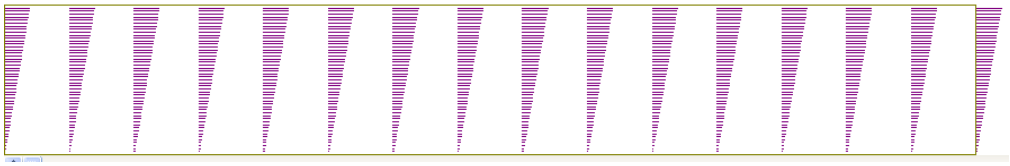


Figure 7.3: Velocity distribution from numerical result on a plane Couette flow

Cylindrical Couette Flow

A cylindrical Couette flow is also simulated for the following reasons: a) In this flow model, the local solid/fluid ratio varies among nodes, thereby the calculations of the local solid/fluid ratio can be verified; and b) not only the velocity field, the torque applied to the walls can also be obtained from the analytical solution, indicating the fluid force applied to a moving boundary can be verified as well. In this test, two concentric cylindrical walls with radius of 60mm and 40mm, respectively, are located in the centre of a computational domain. The domain contains 160×160 lattices with spacing equal to 1mm. The cylindrical walls are rotating with different angular velocities, i.e. 0.0001 rad/s at the outer one and 0.0002 rad/s at the inner one. All other parameters are identical to those in the previous test. For the velocity field between the two walls at the steady state, the analytical solution is given by (7.3)-(7.5), and the torque applied to the walls can be obtained by (7.6).

$$u(r) = Ar + \frac{B}{r}, \tag{7.3}$$

$$A = \frac{\Omega_2 r_2^{\,2} - \Omega_1 r_1^{\,2}}{r_2^{\,2} - r_1^{\,2}}, \tag{7.4}$$

$$B = \frac{(\Omega_1 - \Omega_2) r_1^{\,2} r_1^{\,2}}{r_2^{\,2} - r_1^{\,2}}, \tag{7.5}$$

$$T = \frac{-4\pi\rho\upsilon(\Omega_1 - \Omega_2)r_2^2 r_1^2}{r_2^2 - r_1^2}, \tag{7.6}$$

where r_1, r_2, Ω_1 and Ω_2 are radii and angular velocities of the inner and outer cylinders, respectively. r is the distance between a particular node and the centre of the cylinders. ρ and υ are the density and kinematic viscosity of the fluid, respectively.

At the steady state, the global velocity difference is found to be 1.1% and the relative difference of the torque is calculated as 1.5% . The velocity distribution from the numerical result is shown in Figure 7.4.

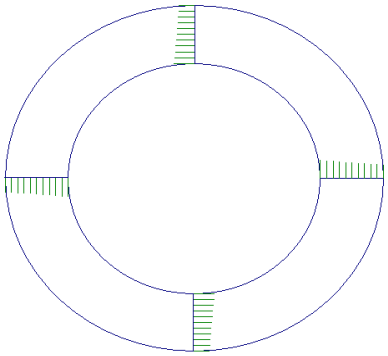


Figure 7.4: Velocity distribution from numerical result on a cylindrical Couette flow

From the Couette flow tests, the simulation results show good agreement with the analytical solutions, indicating LBM with IMB has been successfully implemented in *FPS-BHAM.*

7.2.3 Evaluation of DEM-LBM capability in modelling leakage-soil interaction

An internal fluidisation is found to be the mechanism underlying the leakage-soil interaction (Alsaydalani, 2010). Therefore, an internal fluidisation model, which is based on the experiment conducted by Alsaydalani (2010), is set up to evaluate the DEM-LBM capability in modelling leakage-soil interaction, by comparing the simulation results with the experimental findings.

Comparison between numerical results and experimental findings

An experiment was performed at the University of Southampton to investigate the internal fluidisation in a sand bed (Alsaydalani, 2010). A two-dimensional experimental study (Alsaydalani, 2010) was conducted on a sand bed contained in a tank. It was connected to an inlet water pipe through a slot-shaped orifice. The dimensions of the tank is 600mm × 420mm × 146.6mm, as shown in Figure 7.5. During the test process, a sand bed with 300mm in height was fully submerged under water. The size of sand particles ranged from 0.6mm to 1.2mm with Beta accumulative distribution function. A slot opening along the width direction was chiselled in the middle bottom of the tank, through which a fluid was injected into the sand bed, and activated a two-dimensional soil fluidisation. With progressively increased flow rate, internal fluidisation was identified. Observations in the soil bed were made with the aid of Particle Image Velocimetery (PIV) which was a widely-used image analysis technique. Pore pressures were also monitored using standpipes attached to the centre line of the tank.

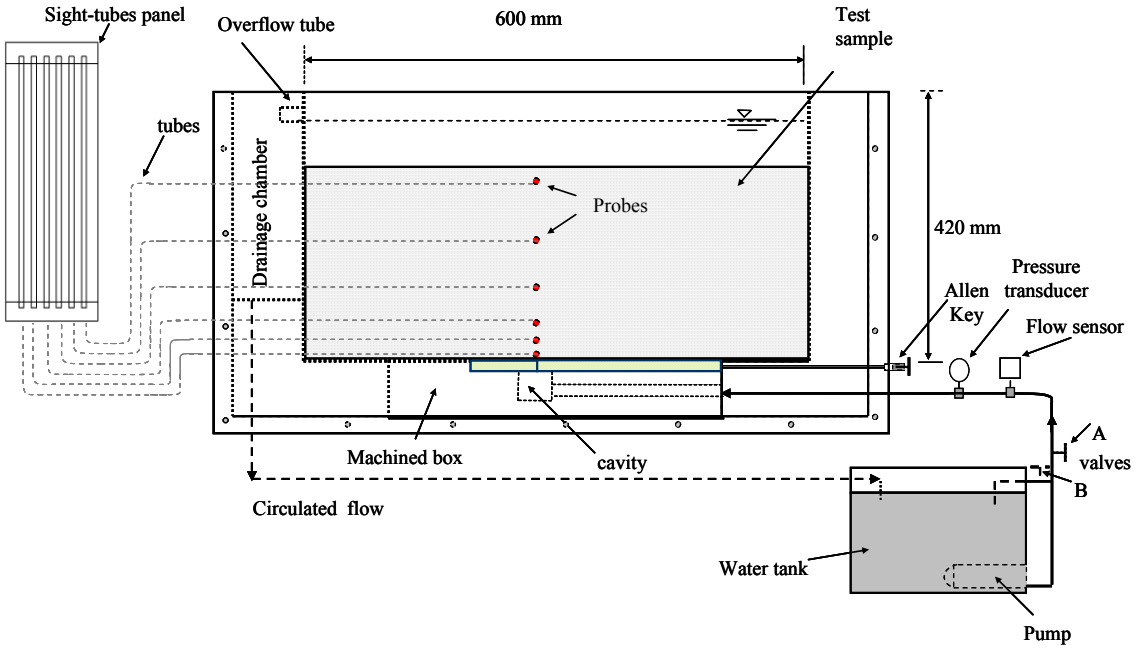


Figure 7.5: Sketch of the experimental setup (Alsaydalani, 2010)

In order to produce comparable numerical results, a two-dimensional numerical model is set up using DEM-LBM technique, which has the same dimensions with those in the experiment. Non-adhesive circular particles are adopted to reproduce the sand bed. The bed is packed by dropping particles freely and settling them down for a sufficient period. The resulting dimensions are of 600mm in length and 290mm on average in height. As shown in Figure 7.6, the particles are coloured in layers so that the deformation in the bed due to the injected fluid can be clearly identified. The whole computational domain is divided into a 600×400 lattice grid, and hence the bed is fully submerged. An orifice with a width of 3mm is located in the middle of the base, connected with an inlet pipe which spans half length of the bed, so as to obtain a locally injected fluid. Wall boundary conditions are applied to the left and right boundaries of the sand bed in both DEM and

LBM calculations. At the top boundary of the computational domain, constant zero pressure is applied. During each test, a constant flow rate is applied as a controlled condition, which ranges from 0.25 l/s to 6.0 l/s among tests.

In consideration of the computational cost, the particle sizes are set to be 3.25mm, 3.50mm, 3.75mm, 4.00mm, 4.25mm, 4.50mm, 4.75mm, 5.00mm, 5.25mm, 5.50mm, 5.75mm, and 6.00mm, of which the numbers are 198, 547, 826, 1035, 1174, 1244, 1244, 1174, 1035, 826, 547, and 198, respectively. It can be noted that the particles are larger than those in a real sand.

Although the particle size, the orifice width, and the flow rate are not directly comparable to the experimental model used by Alsaydalani (2010), it is recognised that a consistent mechanism underlying the onset of fluidisation is still achievable. In an effort to demonstrate the validity of the numerical model, the computational output is compared with experimental findings, then followed by a comparison with the analytical solution. The parameters used in the numerical tests are shown in Table 7.6.

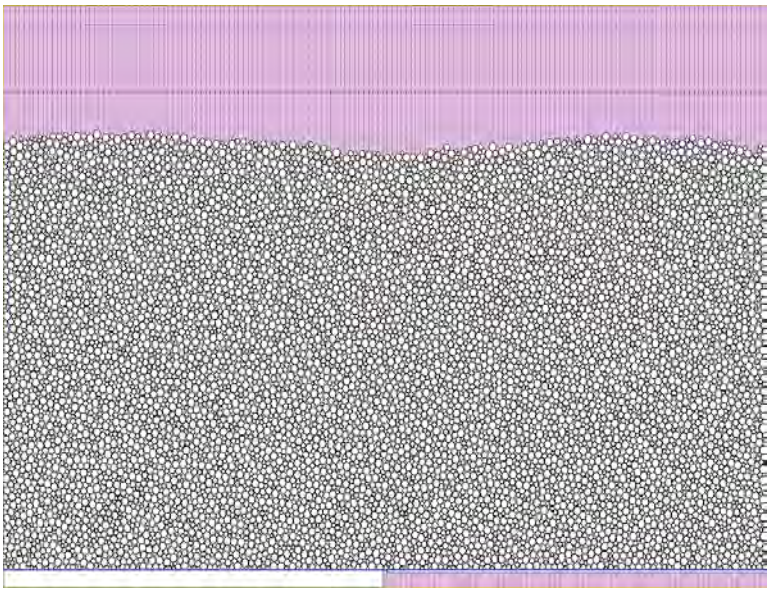


Figure 7.6: Numerical setup

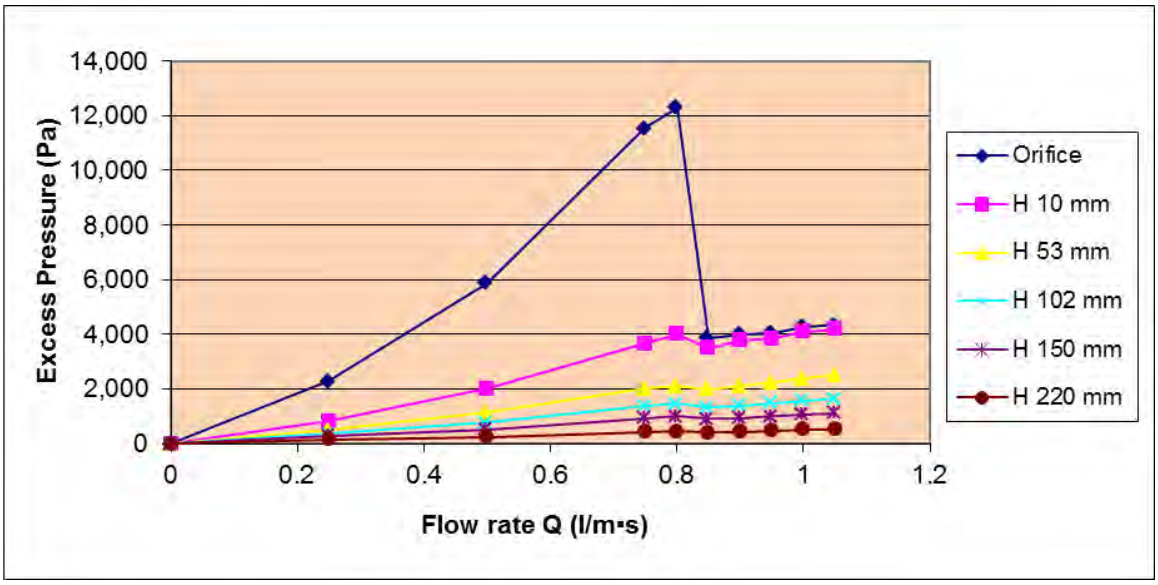
Fluid kinematic viscosity	$\upsilon = 1\times10^{-6} m^2/s$
Lattice spacing	$\Delta h = 1.0\times10^{-3} m$
Time step in LBM	$\Delta t_{LBM} = 1.0\times10^{-4} s$
Fluid density	$\rho = 1\times10^{3}\,\mathrm{kg/m^3}$
Lattice speed	$C = 10 m/s$
Particle density	$\rho_s = 2700 kg/m^3$
Time step in DEM	$\Delta t_{DEM} = 2.5\times10^{-5} s$
Friction coefficient	0.3
Young's modulus	69 MPa
Poisson's ratio	0.3

Table 7.6: Parameters used in numerical simulations

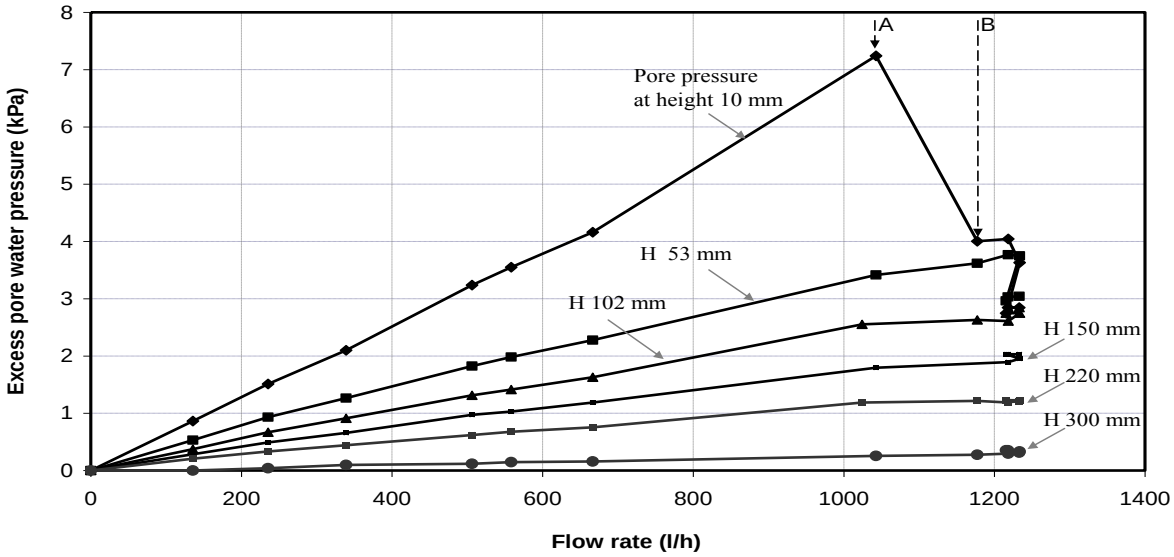
From the numerical results, it has been found out that the pore fluid pressure is accumulated as the flow rate is increasing before fluidisation (see Figure 7.7(a)). Once fluidisation is activated, which is observed as the formation of a small internal cavity, the excess pressures inside the cavity undergo a reduction. If the flow rate is further increased, the pressures then tend to level off. This finding agrees well with the experimental data as shown in Figure 7.7(b).

Figure 7.8 shows the excess pressure distribution along the bed height right above the opening. The reduction in the excess pressure near the opening with the increase in the flow rate indicates the activation of fluidisation. At the onset of fluidisation, the actual excess pressure gradient is much greater than unity, which is the critical hydraulic gradient derived for seepage flows, as shown in Figure 7.8(a). This is due to the localisation of injected fluid involved in this problem. Good qualitative agreement with the experiment data is also achieved, as shown in Figure 7.8(b).

As suggested by Alsaydalani (2010), the soil mass above the fluidised zone was uplifted in a form of an inverted tapered block, and the particles outside it remained stable (see Figure 7.9(a)). The inclination angle of the wedge in the experiment was measured from $63.2°$ to $64.8°$, which was stated to be dependent on neither the opening size or the bed height. This uplifting phenomenon can also be observed from the numerical tests, in which the inclination angle is around $63°$.



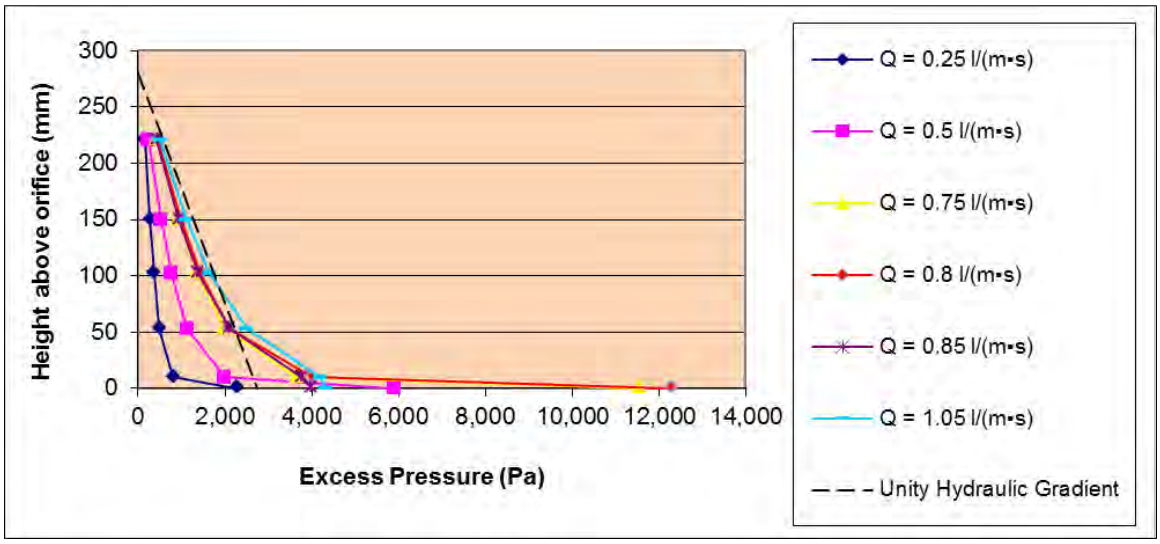
(a)



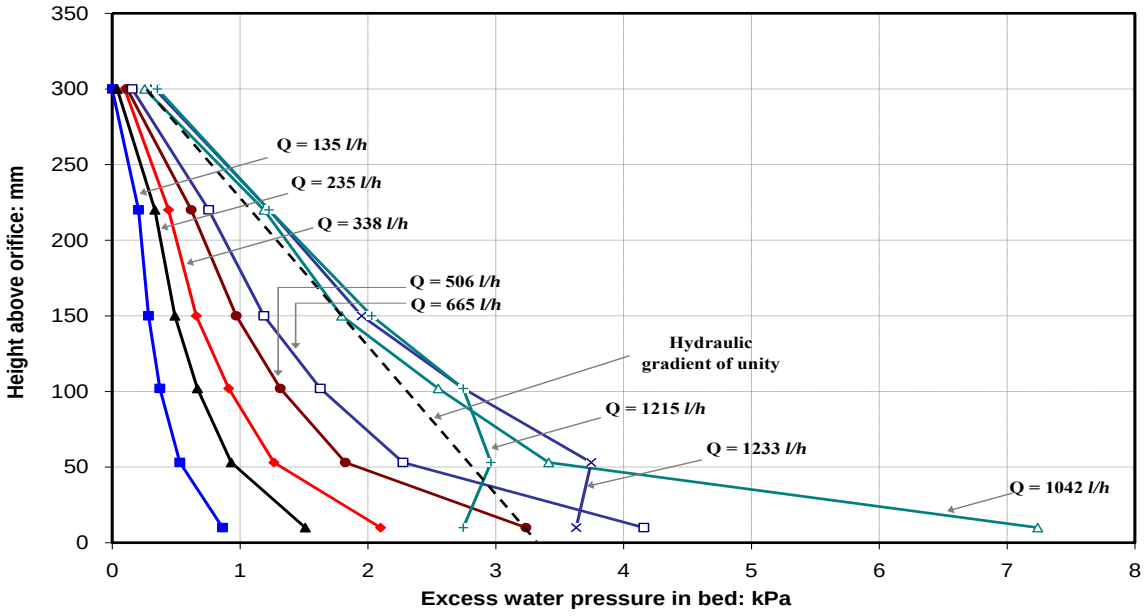
(b)

Figure 7.7: Excess pore pressures in bed with various flow rates

(a) numerical results; and (b) experimental results (Alsaydalani, 2010)



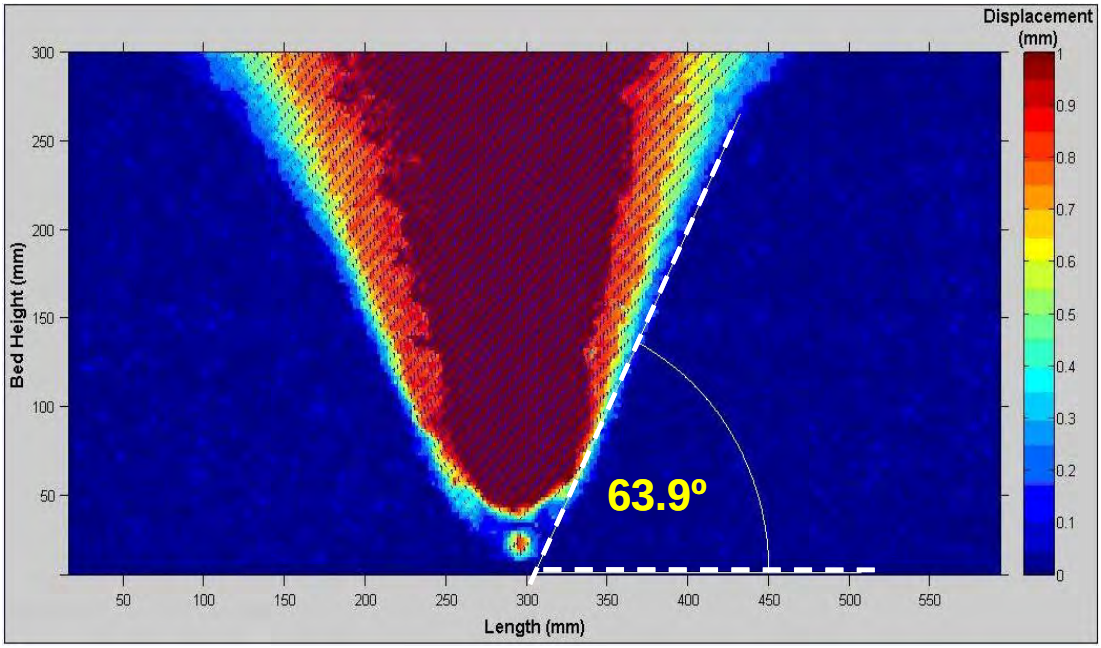
(a)



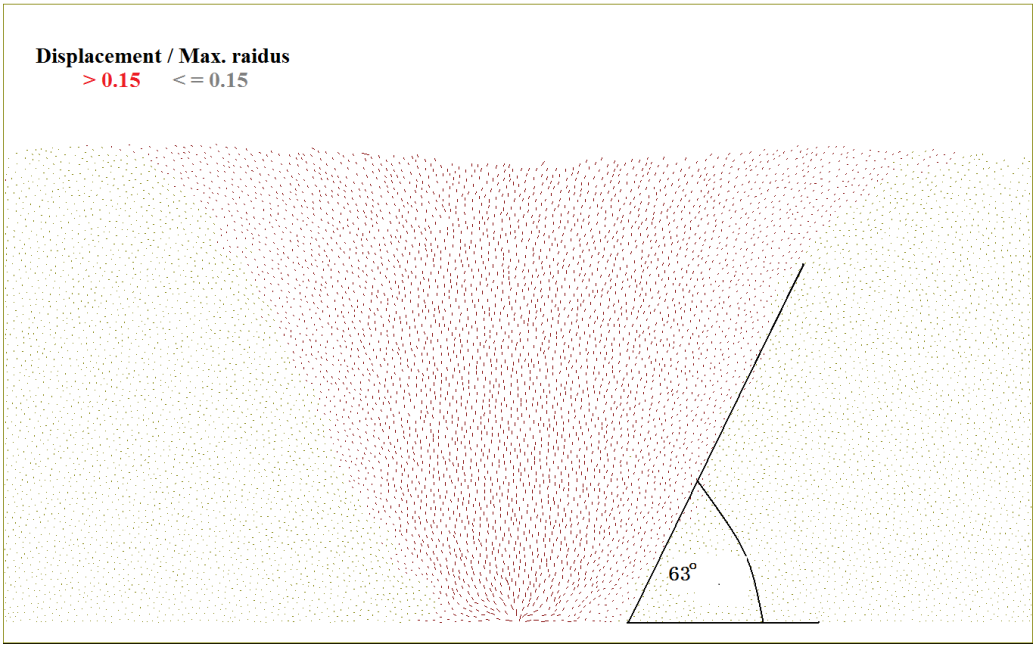
(b)

Figure 7.8: Vertical distribution of excess pore pressures in bed

(a) numerical results; and (b) experimental results (Alsaydalani, 2010)



(a)



(b)

Figure 7.9: Inverted tapered block and wedge angle

(a) experimental result (Alsaydalani, 2010); and (b) numerical result

The above comparisons indicate that a good qualitative description can be achieved using the DEM-LBM technique in simulating the leakage-soil interactions. In order to validate the capability of DEM-LBM in providing quantitative analysis in this problem, a validation against the analytical solution is conducted as shown in the following section.

Validation of the numerical results against analytical solution

At the onset of fluidisation, force equilibrium is achieved in the uplifted soil mass between the seepage force and the effective weight of the soil mass (Alsaydalani, 2010). Based on this force balance, the fluidising pressure can be derived as (see Cui, 2012)

$$\Delta p_f = \frac{\ln(\frac{2H}{L_0})(L_0 + H\cot\alpha)\cdot W}{\pi - 2\alpha}, \tag{7.7}$$

where H, L_0, α are the bed height, the opening size, and the wedge angle, respectively. W is the effective volumetric weight of the soil.

A set of tests are conducted using the numerical model on the soil bed with different heights. The validation of the numerically measured fluidising pressure against the predicted values is shown in Figure 7.10. It can be seen that the results match well the analytical solution.

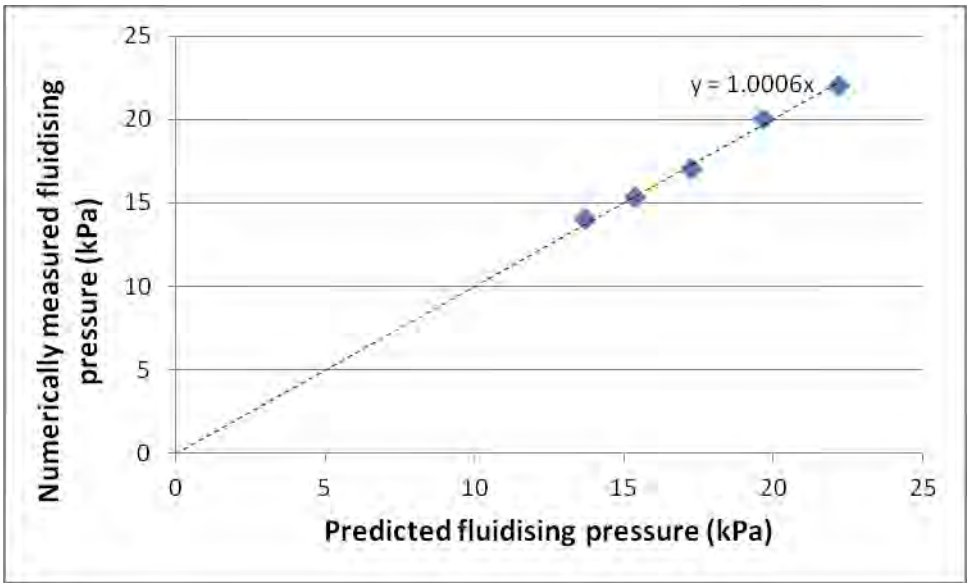


Figure 7.10: Numerically measured and predicted fluidising pressures

In summary, the DEM-LBM technique is a promising tool in simulating the leakage-soil interactions as it is capable of providing both qualitative description and quantitative analysis of the overall behaviours.

7.3 Assessment of parallelisation strategy of DEM-LBM

As mentioned in Section 7.1, the main focus in this section is on the assessment of the parallel performance with the blocked partitioning domain decomposition proposed in Chapter 4. The parallel performance is assessed using the parallel efficiency, which is defined as (Kumar et al, 1994)

$$E = \frac{1}{N_p}\frac{T_{seq}}{T}, \tag{7.8}$$

where T_{seq} is the runtime of the best sequential algorithm. N_p is the runtime of the

parallel algorithm. P is the number of processors used. T_{seq} could be influenced by the number of sub-domains adopted. Hence, such an influence is explored prior to the parallel performance assessment.

A similar leakage-soil interaction model to that presented in Section 7.2.3 is set up with 119,000 particles located in the LBM framework. The domain size is divided into a 2400×2400 lattice grid. All other parameters are the same as those used in Section 7.2.3. Various numbers of sub-domains are sequentially adopted, with same numbers in each dimension (e.g. 2×2, 3×3, 4×4, etc). All the computations are firstly carried out on a single processor.

Among the tests, identical results are found to be obtained, including both the particle movements and the flow field. The CPU time consumed is plotted versus the number of sub-domains in each dimension, as shown in Figure 7.11 below. As more sub-domains are used, more duplicated DEM calculations are performed in the overlapping areas, which leads to more CPU time consumed in the DEM calculation. For LBM, more virtual nodes are also needed and calculated with more sub-domains adopted, thus more CPU time is expected as well. However, it is worth noting, from the test results, that the CPU time of LBM keeps reducing when the number of sub-domains is increasing until it reaches 32×32. Such a phenomenon is mainly attributed to a sharp reduction of CPU time in the streaming process of LBM. As it is also observed and studied by Satofuka and

Nishioka (1999), the reason was proved to be the 'pseudo-vector processing capacity', indicating that the reduction of the array inner loop size causes a save in runtime in the array operation for an equal amount of data. This feature boosts the parallel performance of LBM in the domain decomposition scheme.

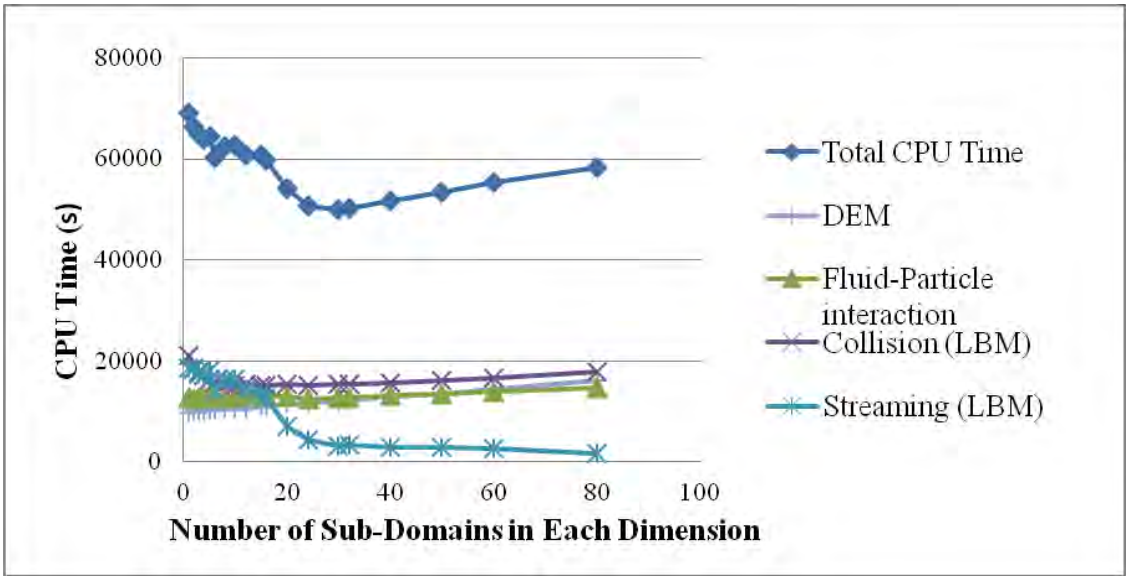


Figure 7.11: CPU time verses number of sub-domains in each dimension

The minimal computation time used in a single processor, which is 52,364 seconds, is obtained using 32×32 sub-domains. This time is therefore used as a benchmark, T_{seq}, in the analysis of the parallel performance of DEM-LBM code.

With the same numerical model, simulations are then carried out with parallel computing. The parallel efficiency is plotted versus the number of processors, as shown in Figure 7.12. It is seen that with 32 processors, the efficiency of DEM-LBM is 0.72. It

is considered as a good parallel behaviour of DEM-LBM with the blocked partitioning domain decomposition proposed in this thesis. It is also noted that the LBM parallel efficiency is higher than that of DEM, indicating the overall performance is dependent on the ratio of computing load of LBM to DEM. Therefore, in the cases where the DEM sub-cycle number is relatively low or the porosity of the medium being modelled is relatively high, the computing load ratio of LBM to DEM tends to be relatively large, and a relatively high parallel efficiency can be achieved.

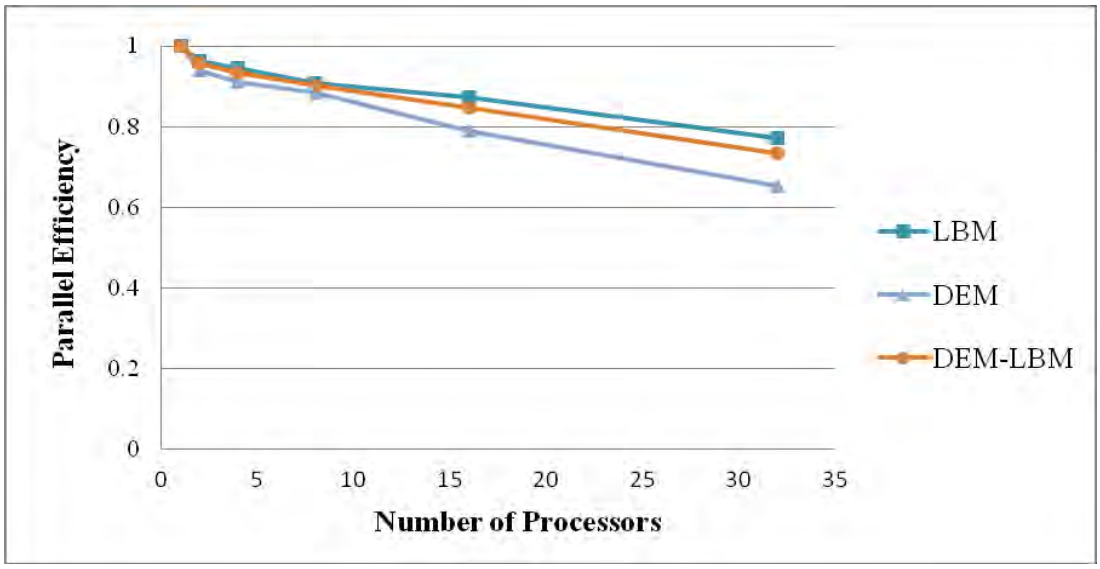


Figure 7.12: Parallel efficiency verses number of processors

7.4 Validation of the combination strategy between DEM-LBM and DEM-DFF

In this section, the validation of the combination strategy between DEM-LBM and DEM-DFF (see Chapter 5) is conducted. Prior to the validation of the combination

strategy, the verification of pure DEM-DFF computation is carried out in Section 7.4.1. The validation of the combination strategy is performed in two steps: a) the validation of the combination of LBM and DFF, which is shown in Section 7.4.2; and b) the validation of the combination strategy with the presence of particles is presented in Section 7.4.3.

7.4.1 Verification of DEM-DFF computation

A numerical test on a one-dimensional pressure-driven flow through a porous medium is performed to verify the computation of DEM-DFF. The porous medium has the size of 0.16m×0.15m, as shown in Figure 7.13. It consists of 1,574 circular particles with diameter of 4mm. The domain is divided into 16×15 DFF cells, with around 6 particles in each cell. The periodic condition is adopted at the left and right boundaries. At the bottom and top boundaries, pressure inlet and outlet boundary conditions are applied, with the pressure gradient across the porous medium calculated as $\Delta P / \Delta y = 22753\text{Pa/m}$. Stationary DEM walls are applied to all the boundaries in order to prevent the particles from moving out of the domain. After the steady state is reached, the pressure distribution contour is plotted in Figure 7.14. The global pressure difference is computed as 9×10^{-8} using (7.9), which indicates the result agrees well with the analytical solution.

$$E_v = \frac{\sqrt{\sum (p' - p)^2}}{\sqrt{\sum p^2}}, \tag{7.9}$$

Where p' and p is the pressure from numerical test and analytical solution,

respectively. p is calculated as

$$p = \frac{H - y}{H} \Delta p \,, \tag{7.10}$$

where H is the height of the porous medium, and y is the distance to the bottom boundary.

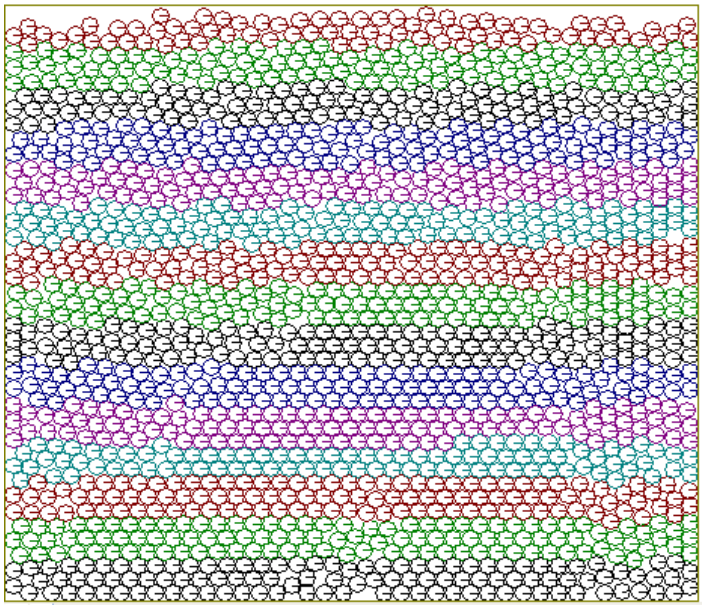


Figure 7.13: A DEM-DFF model of a one-dimensional pressure-driven flow through a porous medium

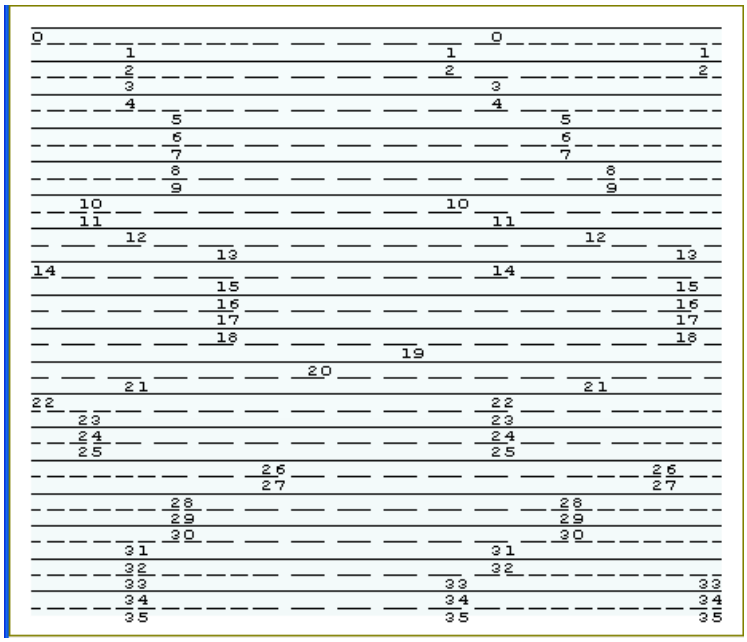


Figure 7.14: DEM-DFF result: pressure distribution contour in a one-dimensional pressure-driven flow through a porous medium

In order to verify the calculation of the fluid force applied to the particles, a comparative DEM-LBM test is carried out. The lattice spacing in DEM-LBM is chosen to be 0.001m. The time step used is $1.67\times10^{-5}s$. The lattice speed C is 60m/s and the relaxation time $\tau = 0.50005$. Different sets of pressure gradient in DFF and density gradient in LBM are employed, respectively, and the conversion relationship is shown as

$$\frac{\Delta p}{\Delta y} = C_s^2 \frac{\Delta \rho}{\Delta y}. \tag{7.11}$$

Since it is a one-dimensional flow, the fluid force generated in the x direction is considered as negligible. After the steady state is reached, the total fluid force in the y direction applied to all particles is plotted against the pressure gradient in Figure 7.15. It

is firstly noted that the fluid force is approximately proportional to the pressure gradient. This indicates the hydrostatic force $\mathbf{f}_b^p$, which is linearly related to fluid velocity, is of key importance in this case. It can also be seen that the fluid force calculated in DEM-LBM is slightly less than that calculated in DEM-DFF, and the relative difference ranges from 4.19%~5.32%. This is mainly because the fluid force equation in DEM-DFF (5.14) cannot accurately describe the effect caused by the presence of other particles, even though the porosity of the soil bed is considered (Zhu et al, 2007). While such an effect can be taken into account when calculating the fluid force in DEM-LBM, therefore, the value calculated in DEM-DFF is a bit higher than that calculated in DEM-LBM.

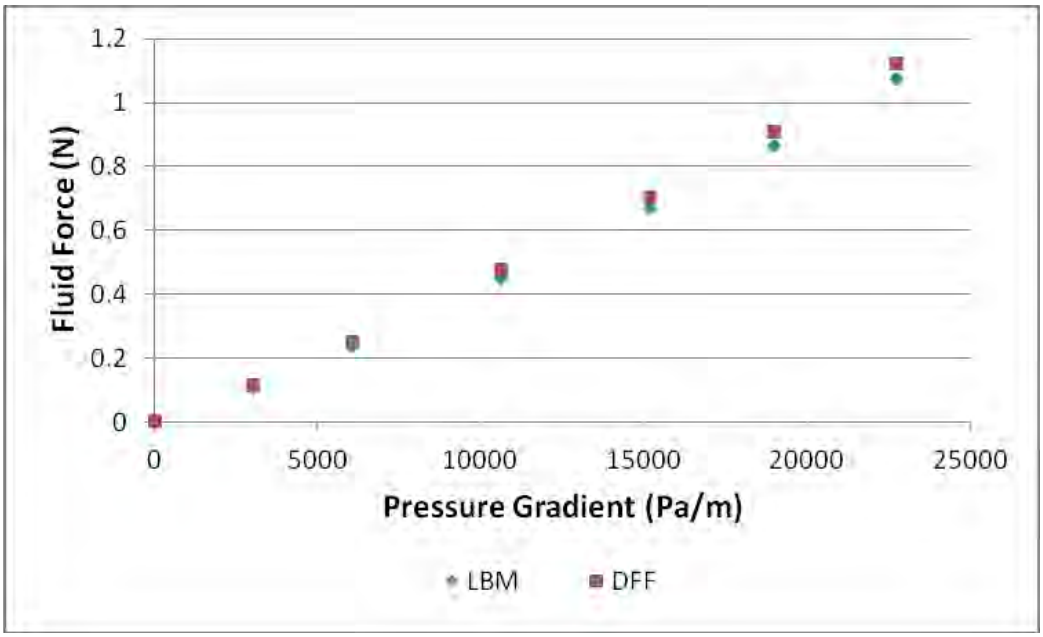


Figure 7.15: Pressure gradient verses fluid force

From the test results presented in this sub-section, it can be concluded that computation of DEM-DFF is capable of providing descriptions to both the fluid field and the fluid force applied to particles with acceptable differences.

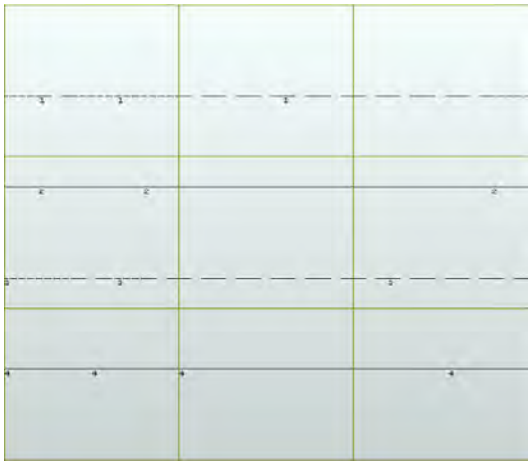
7.4.2 Validation of the combination strategy between DFF and LBM

Numerical tests demonstrated in this sub-section aim to validate the combination strategy between DFF and LBM, which is proposed in Section 5.3. Both one- and two-dimensional pressure-driven flows are simulated, and the test setup and results obtained are presented as below, respectively.

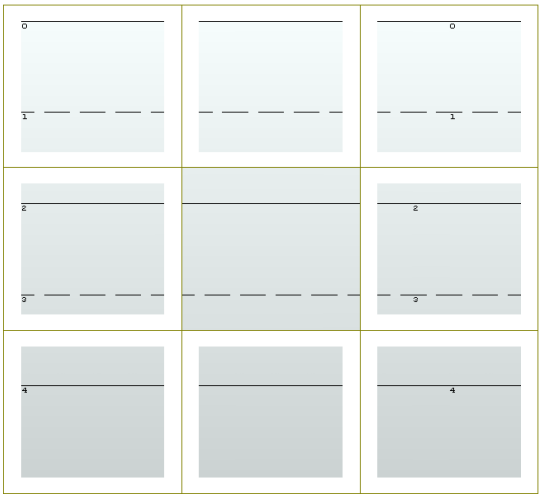
One-dimensional pressure-driven test

A fluid domain is divided into 3×3 sub-domains with each of them contains a 10×10 grid. Three fluid models are employed by: i) pure LBM, ii) the combined system with LBM in the middle sub-domain, and iii) pure DFF. The lattice speed C is 10m/s and the kinematic viscosity is $\upsilon = 1\times10^{-3}\,\mathrm{m}^2/s$. The relaxation time is calculated as $\tau = 0.8$. The time step used in DFF is identical to that in LBM, which is 1×10^{-4}s. Periodic boundary condition is applied to the left and right boundaries. At the bottom and top boundaries, pressure inlet and outlet boundary conditions are applied with pressure gradient $\Delta P/\Delta y = 333\mathrm{Pa/m}$ across the domain, which is sufficiently low to maintain a small Mach number during the test (i.e. within 10^{-3}). Both the extrapolation scheme (see Section 5.3.2) and the non-equilibrium bounce-back scheme (see Section 5.3.3) are

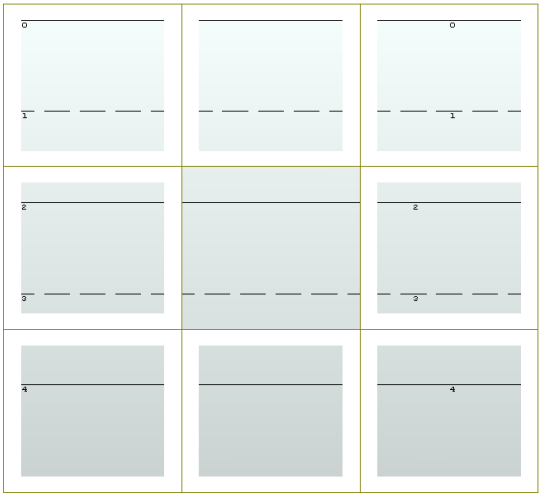
employed and the pressure distribution contour at the steady state is shown in Figure 7.16. The global difference of all tests is within 1×10^{-6}, which indicate that both the extrapolation scheme and the non-equilibrium bounce-back scheme show a good performance in data transfer between DFF and LBM in this case. The ratio of computing time on a single processor is found to be 126:22:9 (for LBM only: the combined system: DFF only) and the memory cost ratio is 117:21:9, which suggests applying combination strategy makes the computation much more economical.



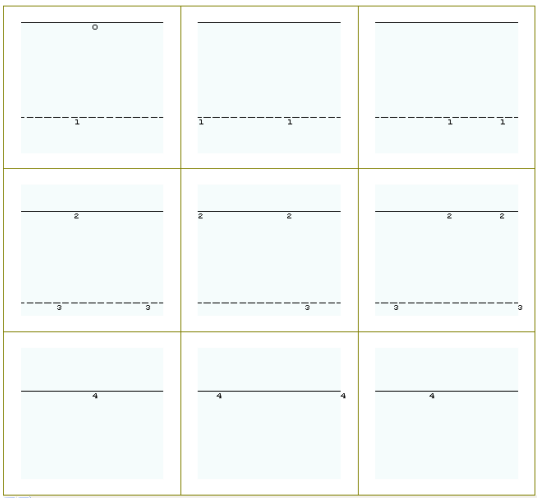
(a)



(b)



(c)



(d)

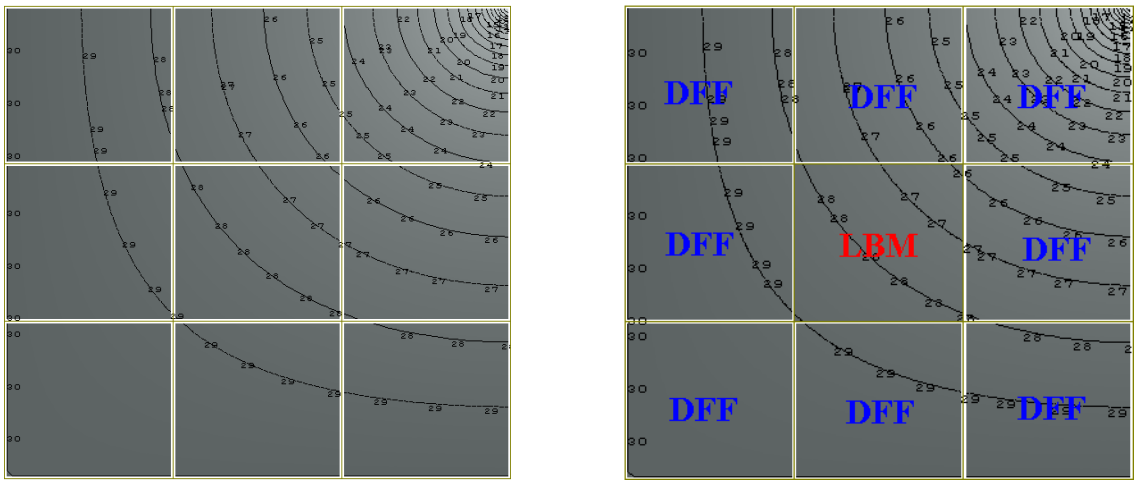
Figure 7.16: One-dimensional pressure-driven flow: pressure distribution contour at the steady state by (a) LBM only; (b) combined system with extrapolation scheme; (c) combined system with non-equilibrium bounce-back scheme; and (d) DFF only

If the kinematic viscosity is adopted as $\upsilon = 1\times10^{-6}\,\mathrm{m}^2/s$, which is the value for water at room temperature. The relaxation time reduces to $\tau = 0.5003$. In this case, the non-equilibrium bounce-back scheme generates an acceptably accurate pressure contour

at the steady state with the global difference within 1×10^{-6}. However, the extrapolation scheme fails to give a stable solution during the test. This failure is due to its assumption that a linear relationship exist for the density distribution function, which has been validated in the cases with a relatively high relaxation time (≥ 0.7) only (Chen et al, 1996). Therefore, in this thesis, the non-equilibrium bounce-back scheme is adopted as water is simulated.

Two-dimensional pressure-driven test

In order to validate the two-dimensional performance of the combined system, a two-dimensional pressure-driven test is carried out by i) pure DFF, and ii) the combined system with LBM in the middle sub-domain. Zero excess pressure is applied to the top-right corner node and kept fixed during the test. And the left and bottom boundaries are defined to have a fixed higher pressure. The pressure difference between the top-right corner to the left/bottom boundary is $\Delta P = 100\text{Pa}$. At the top and right boundaries, the pressure gradient is held constant to be zero across the boundaries. All other parameters in both DFF and LBM are identical to that in the previous case with $\tau = 0.5003$. The pressure distribution contour at the steady state is shown in Figure 7.17.



(a) (b)

Figure 7.17: Two-dimensional pressure-driven flow: pressure distribution contour at the steady state by (a) pure DFF; and (b) combined system

The pressure distribution along the line from the top-right corner to the left-bottom corner is shown in Figure 7.18. After 0.005s, some difference in the pressure distribution appears in the middle sub-domain. Such a difference is due to the different forms of the governing equations. In LBM which is based on the hyperbolic Partial Differential Equation (PDE), convective behaviour is modelled. While in DFF which is based on the parabolic PDE, only diffusion behaviour is modelled. This difference is further explored in Chapter 8. As time goes, such the difference gradually extends to the whole computational domain. In spite of this, after a short period (at around 0.5s), the difference disappears as the steady state is reached and the two curves nearly overlap. The global difference at the steady state calculated over the whole domain is 1.3×10^{-6}, which indicates a good agreement with each other.

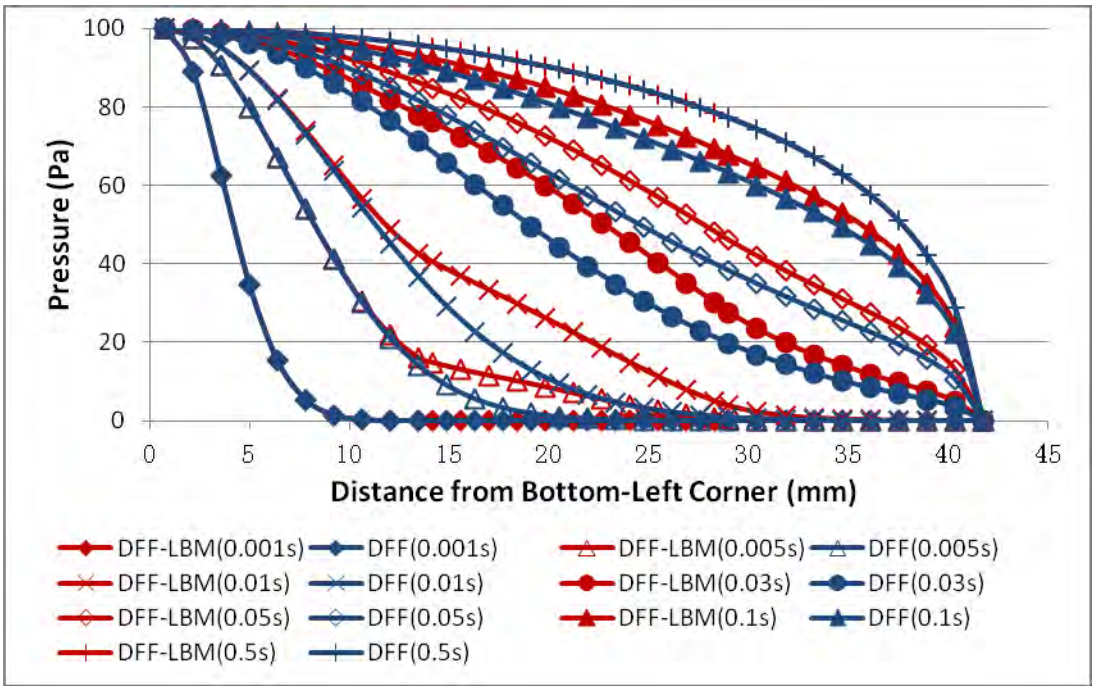


Figure 7.18: Pressure distribution along top-right corner to left-bottom corner

In this sub-section, both one- and two-dimensional pressure-driven flows are simulated in the combined system. Numerical results demonstrate the non-equilibrium bounce-back scheme provides successful combination of DFF and LBM. However, the extrapolation scheme achieves good performance only with a relatively large relaxation time. Therefore, in this thesis, only the non-equilibrium bounce-back scheme is employed for the interface treatment.

7.4.3 Validation of the combined system with the presence of particles

Further work is conducted to validate the combination strategy with the presence of particles. In order to carry out the comparison, a pipe leakage test with the DEM-LBM technique in Cui (2012) is reproduced in the combined system.

A two-dimensional numerical model is set up with its configuration shown in Figure 7.19. The size of the domain is 600mm×400mm. A fully submerged bed consisting of non-adhesive circular particles is adopted to produce a sand bed. By dropping particles freely and settling them down for a sufficient period, a sand bed with 600mm in length and 290mm on average in height is formed. The particle sizes are set to be 3.25mm, 3.50mm, 3.75mm, 4.00mm, 4.25mm, 4.50mm, 4.75mm, 5.00mm, 5.25mm, 5.50mm, 5.75mm, and 6.00mm, of which the numbers are 198, 548, 828, 1040, 1179, 1256, 1256, 1179, 1040, 828, 548, and 198, respectively. In the figure, the particles are coloured in layers so that the deformation in the bed due to the injected fluid can be clearly identified. Wall boundary conditions are applied to the left and right boundaries of the sand bed in both DEM and LBM calculations. At the top boundary of the computational domain, constant zero pressure is applied. A pipe with 30mm diameter is placed at the bottom of the soil bed with a 4mm-wide orifice opening located in the middle of the pipe top surface, from which pipe water injects into the soil bed. A constant pressure of 16.67 kPa is applied to both the left and right ends of the pipe. Other parameters used in the DEM-LBM test are adopted and listed in Table 7.7.

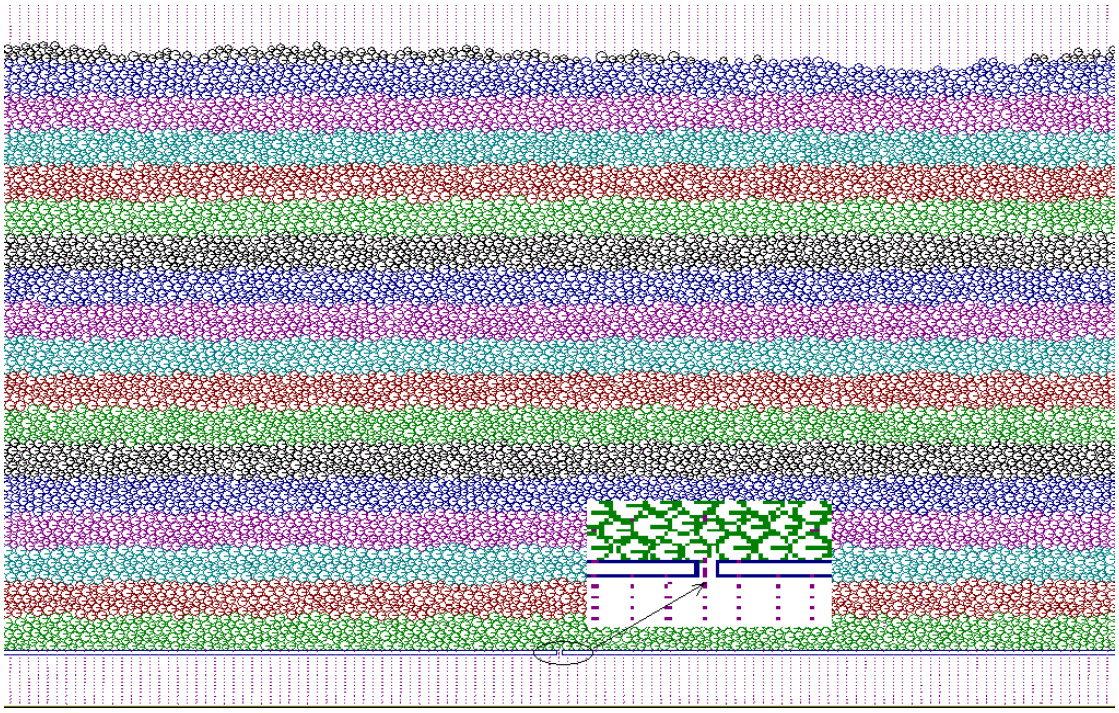


Figure 7.19: Setup of the pipe leakage model

Particle density (kg/m^3)	2700
Friction coefficient in DEM	0.3
Young's modulus (MPa)	69
Poisson's ratio	0.3
DEM time step (s)	2.0×10^{-5}
Fluid density (kg/m^3)	1000
Fluid kinematic viscosity (m^2/s)	1.0×10^{-6}
Lattice spacing (m)	1.0×10^{-3}
LBM time step (s)	1.0×10^{-4}
Dimensionless relaxation time	0.5003

Table 7.7: Parameters in the pipe leakage test

In order to implement the combination strategy, the domain is divided into 3×5 sub-domains. As shown in Figure 7.20, the DEM-DFF technique is used in the sub-domains with seepage flows only, which has been observed from the previous DEM-LBM test (Cui, 2012). While in the remaining sub-domains, the DEM-LBM technique is employed. The cell size of a DFF sub-domain is 10mm, which is 10 times larger than the lattice spacing in a LBM sub-domain. From a preliminary test on the same soil sample under a one-dimensional pressure driven flow, the averaged permeability of the soil is calculated as 4.67×10^{-4} m/s, which is then used for calculating the drag force in DEM-DFF.

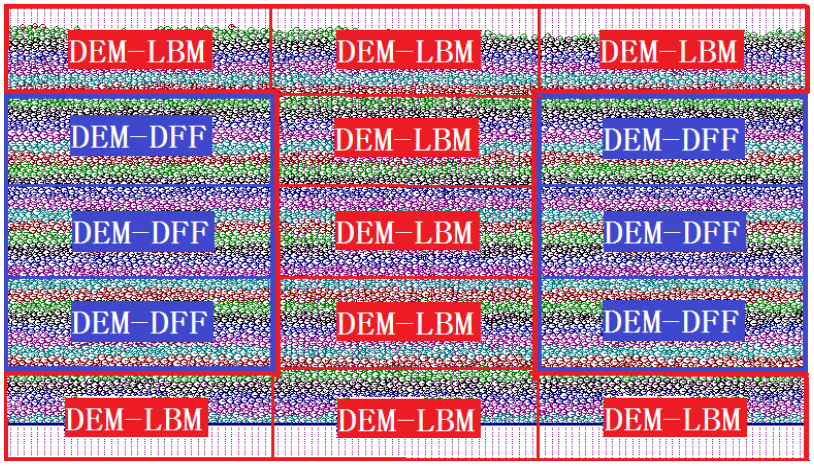
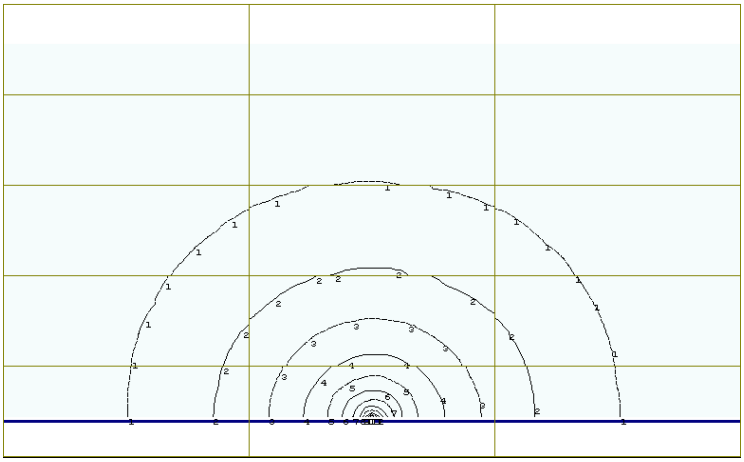


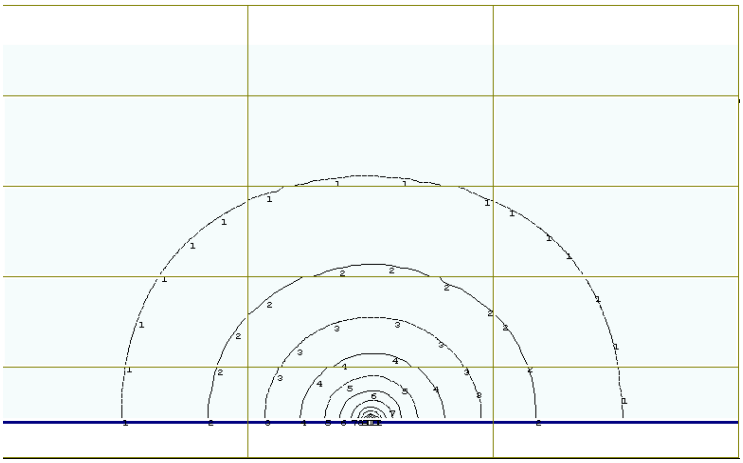
Figure 7.20: Sub-domain arrangement for a pipe leakage problem

In the first 4 seconds, the particles are kept fixed in order to observe the performance of the combination strategy for a steady flow. Afterwards, the calculation of DEM is switched on to validate the combination strategy in the dynamic process when a cavity develops at the leaking area. The pressure distribution contours at 4s are presented in

Figure 7.21, from which a general good agreement is achieved.



(a)



(b)

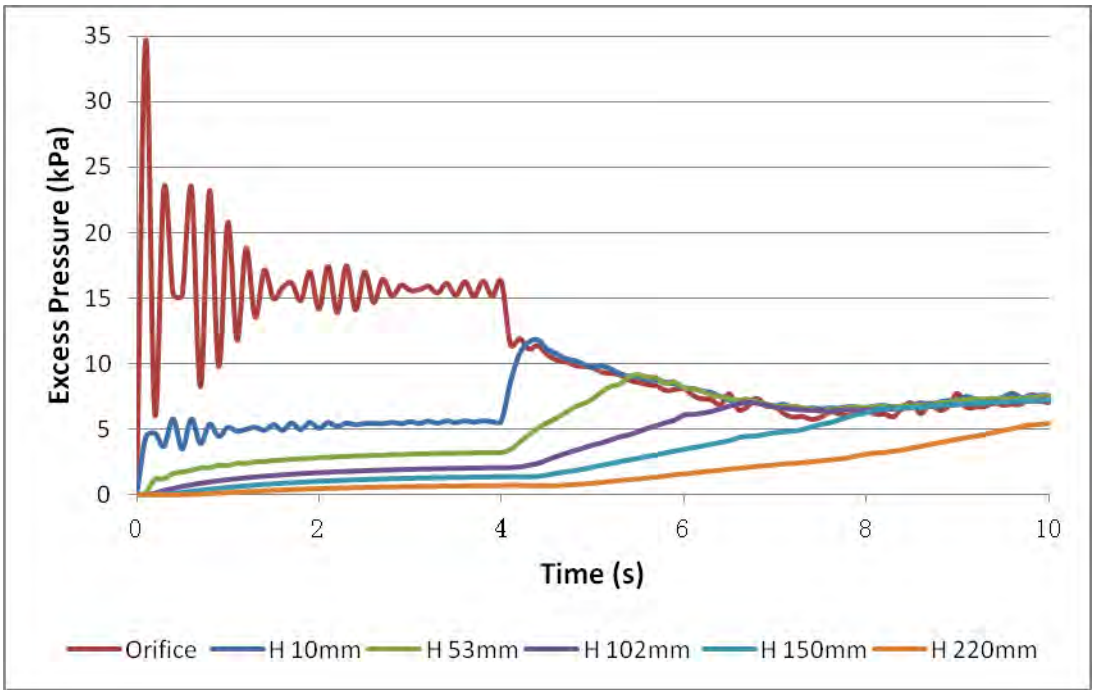
Figure 7.21: Pressure distribution contour at 4 seconds:

(a) DEM-LBM results; and (b) combined system results

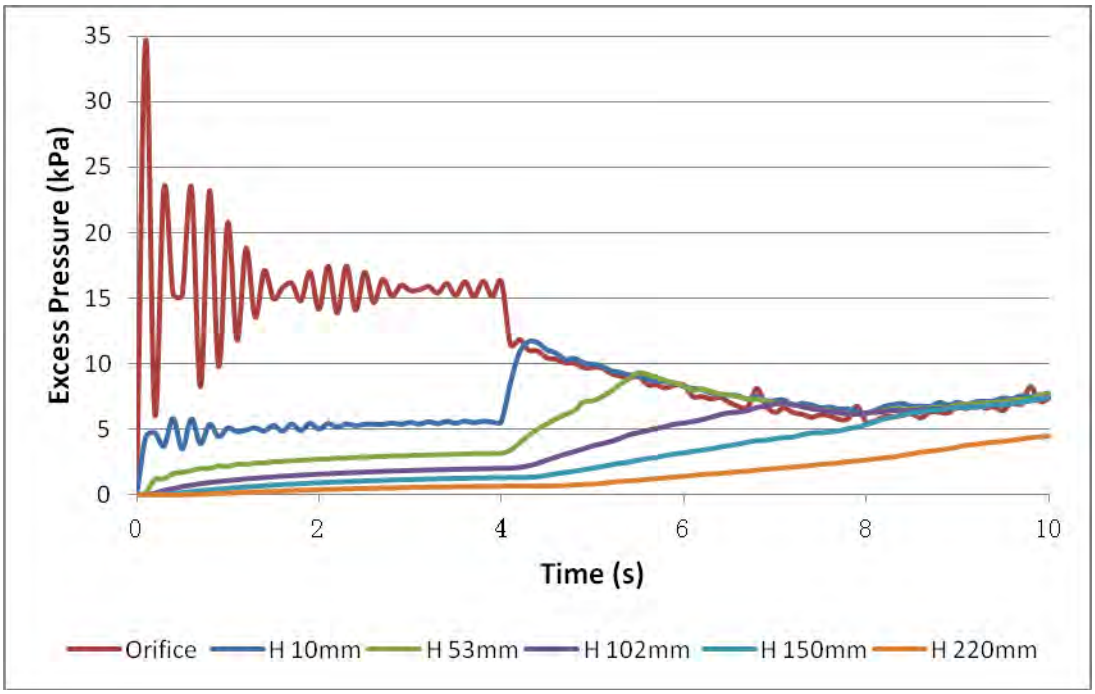
The individual pressure time p-t curves within the soil bed at different heights are presented in Figure 7.22, and it can be found out that the same general behaviour of the pressure evolution is exhibited. The comparison of these p-t curves at different heights

within the soil bed is shown in Figure 7.23. From calculations, the global differences are found to be 2.38%, 2.44%, 3.24%, and 4.5%, respectively, which are thought to acceptable for an engineering problem.

The comparison of the velocity evolution at the orifice and the cavity evolution are shown in Figures 7.24 and 7.25, respectively. The global difference is calculated to be 2.52% and 3.46%. A quite satisfied agreement is achieved from the above comparison, which indicates the successful combination is implemented in this thesis.



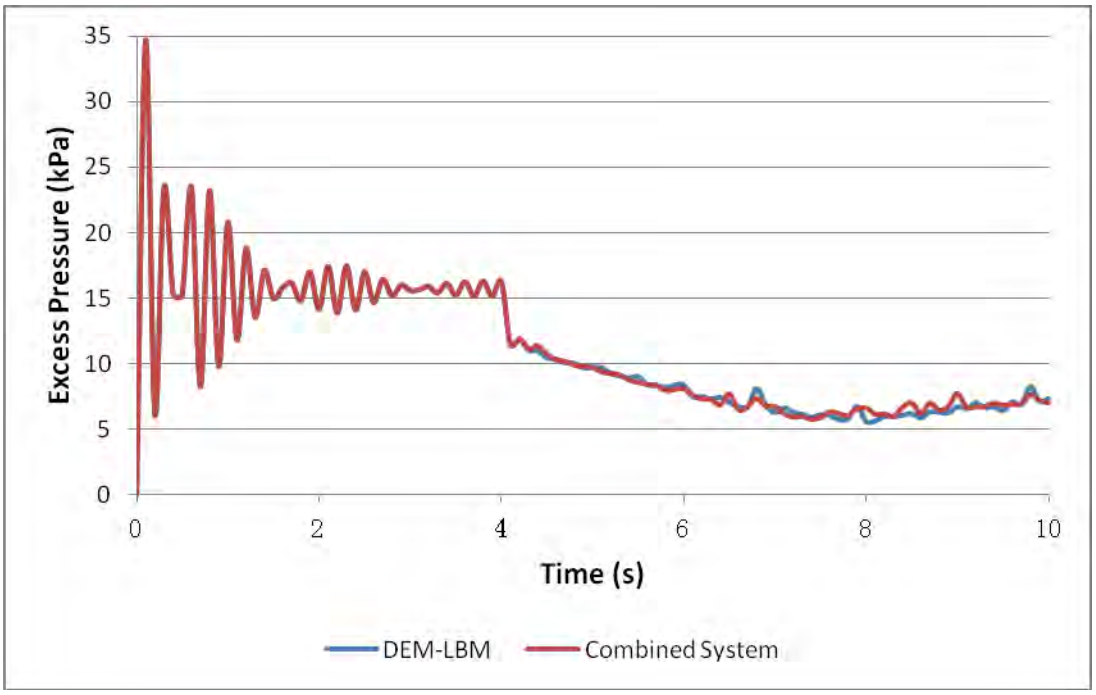
(a)



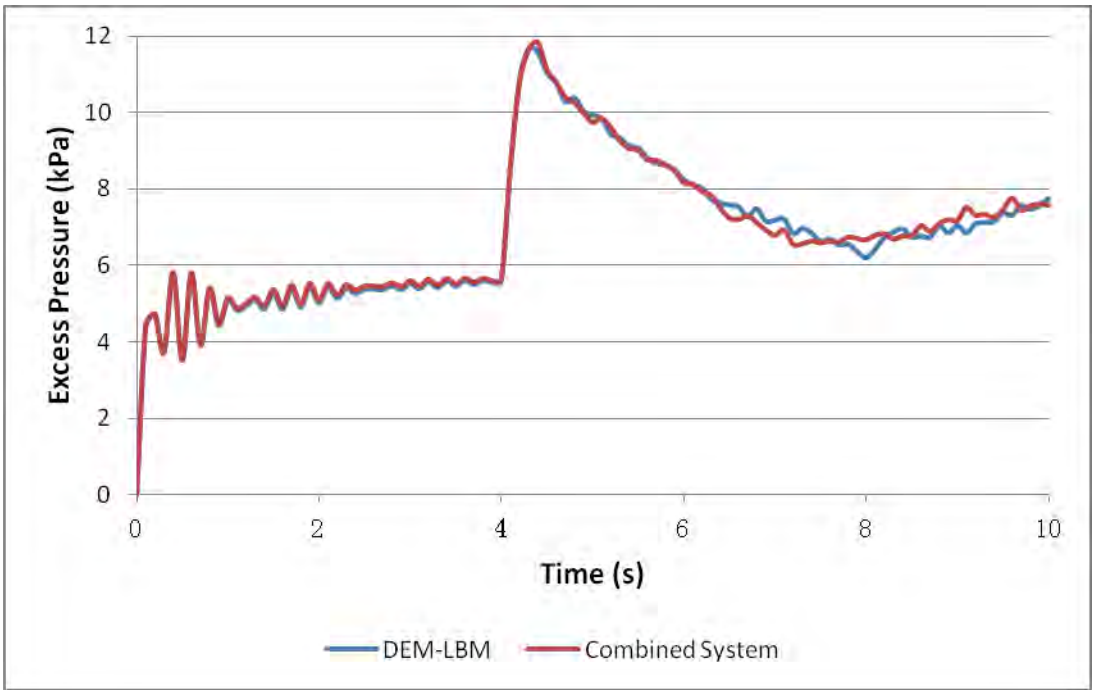
(b)

Figure 7.22: p-t curves within soil bed right above the orifice:

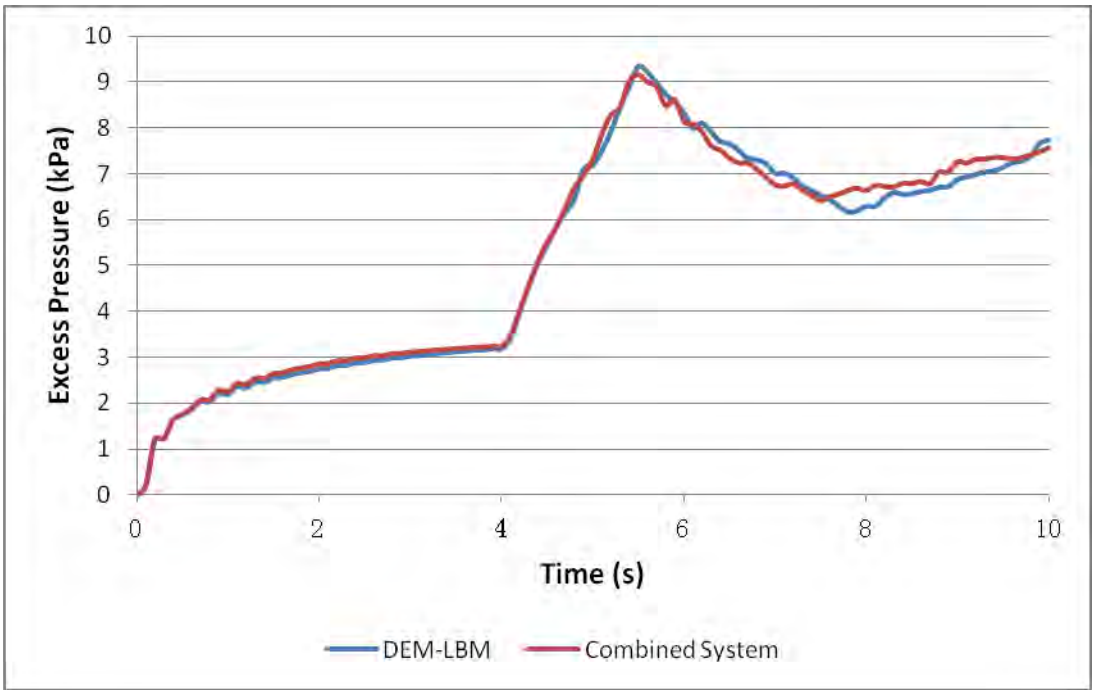
(a) DEM-LBM results; and (b) combined system results


Excess Pressure (kPa)
35
30
25
20
15
10
5
0
0
2
4
6
8
10
Time (s)
DEM-LBM
Combined System

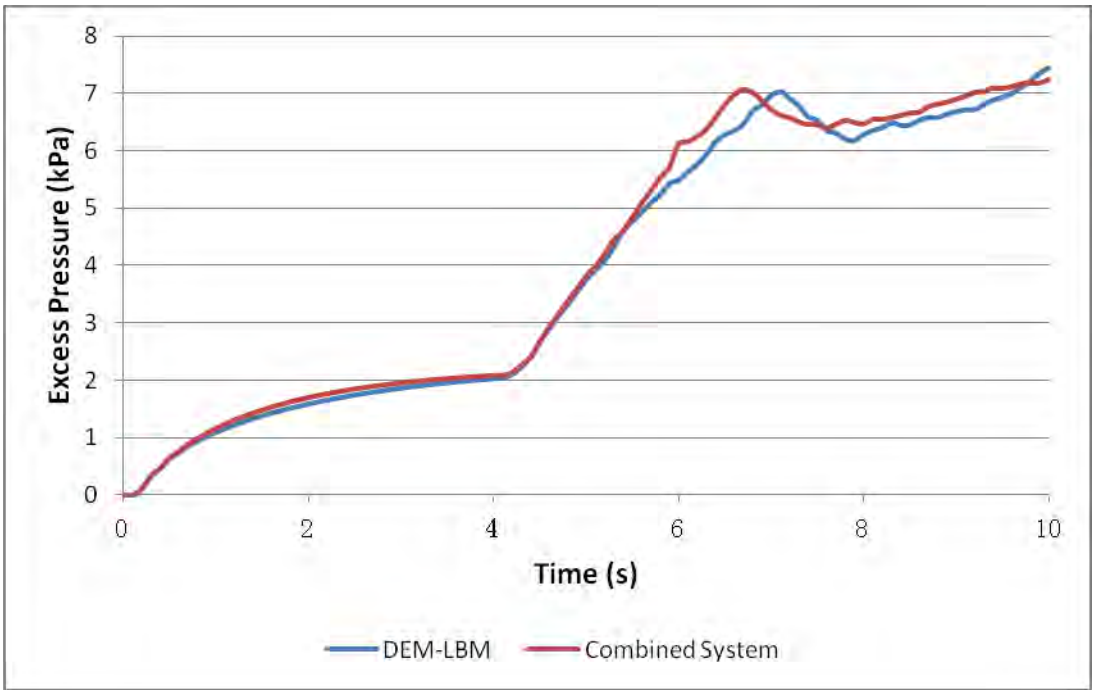
(a)


Excess Pressure (kPa)
12
10
8
6
4
2
0
0
2
4
6
8
10
Time (s)
DEM-LBM
Combined System

(b)



(c)



(d)

Figure 7.23: A comparison of p-t curves between the results by DEM-LBM and combined system at different heights: (a) 0mm; (b) 10mm; (c) 53mm; and (d) 102mm.

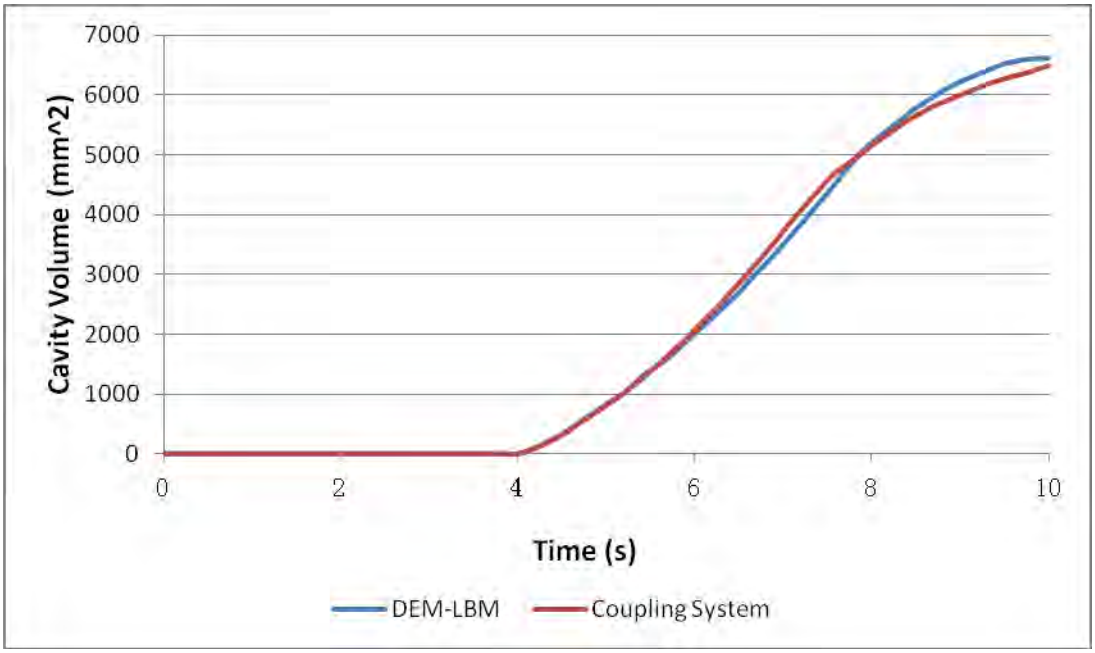


Figure 7.24: A comparison of the velocity evolution at the orifice between results by DEM-LBM and combined system

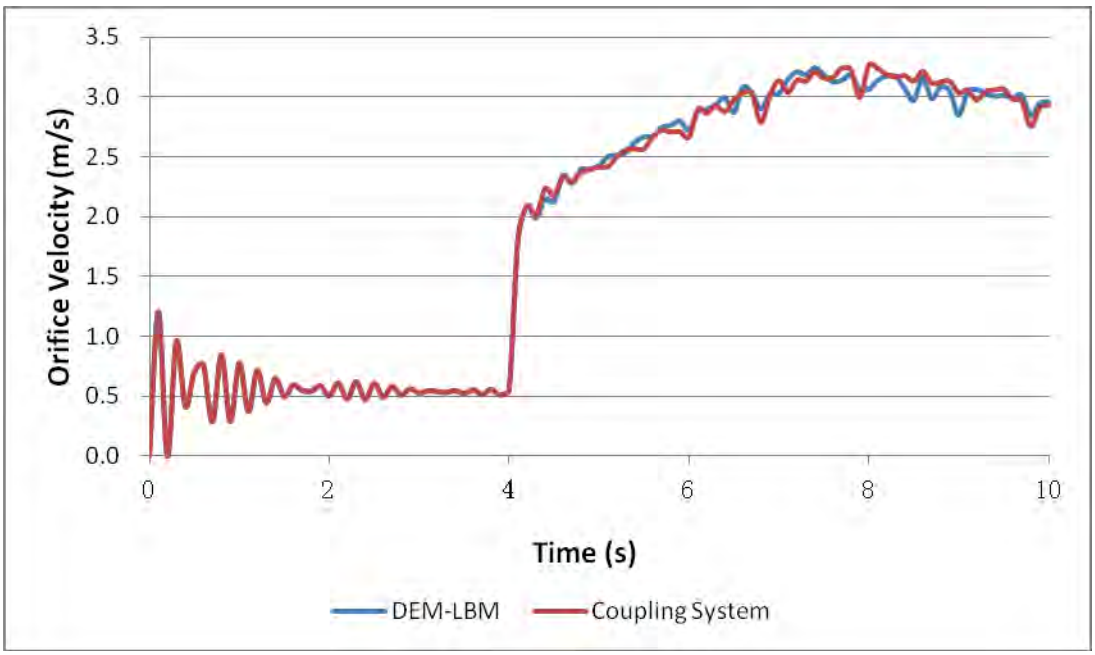
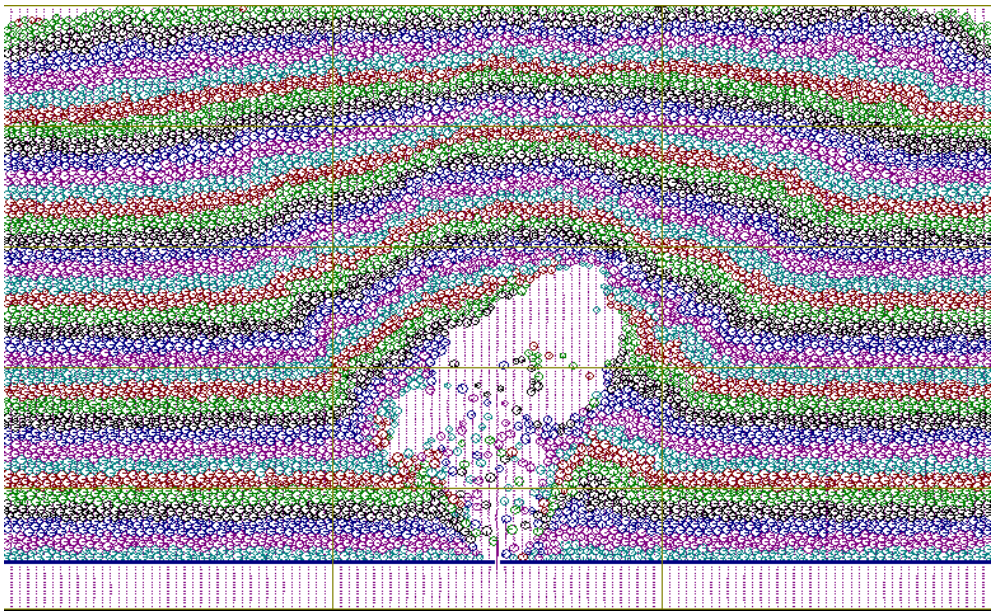


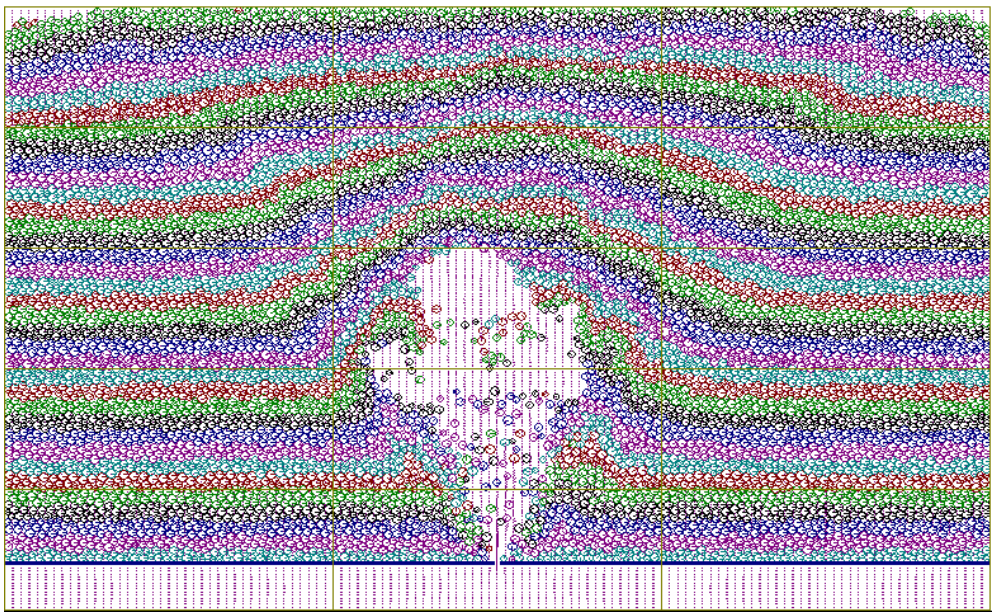
Figure 7.25: A comparison of the cavity evolution between results by DEM-LBM and combined system

From Figure 7.23 – Figure 7.25, it can be observed that the major component of the difference happens in the later stage of the simulation, which may be attributed to the different cavity shapes developed in the two tests. The particle configurations at the end of tests (i.e. at 10s) are provided in Figure 7.26. The cavity shape in the DEM-LBM simulation shows a more asymmetrical pattern compared with that in the combined system. This could be due to the different simulation scales of DEM-LBM and DEM-DFF. In DEM-LBM, the fluid-particle interactions are described with a fine mesh, which is based on microscopic momentum conservation. While in DEM-DFF, the cell size is larger and the force calculated is a local average, which may not be able to capture the subtle changes within a cell. Therefore, DEM-DFF with coarse mesh is not as sensitive as DEM-LBM to the local variety of the particle configurations, and a more symmetrical cavity is formed in the combined system.

The total CPU time spent by the DEM-LBM simulation has been 21,353 seconds, while it takes only 12,460 seconds by the combined system with the same single processor used. Hence, a 42% save in the computational time is achieved in the combined system, which indicates a good computational benefit in the proposed combination strategy.



(a)



(b)

Figure 7.26: The particle configurations at 10s:

(a) DEM-LBM results; and (b) combined system results

7.5 Summary

In this Chapter, verifications are carried out to evaluate the algorithms and implementation in Chapters 4-6.

Firstly, the DEM-LBM implementation is verified and its capability in modelling leakage-soil interaction is evaluated. As the contact force calculation of the DEM part in *FPS-BHAM* is extracted from the well-developed *TRUBAL* code, comparisons are made between the *FPS-BHAM* results and the *TRUBAL* results regarding the DEM calculations. It has been proved the DEM part works properly in *FPS-BHAM*. Simulations of a plane Couette flow and a cylindrical Couette flow are conducted and results are compared with analytical solution. Good agreement is achieved which indicates successful implementation of LBM with IMB in *FPS-BHAM*. It is then followed by the simulation of an internal fluidisation problem, and the results are validated against previous experimental findings. Good qualitative agreements are achieved. Furthermore, the fluidising pressures measured from the numerical tests match well the predicted ones from the analytically derived equation. All the above verifications indicate the DEM-LBM technique has been successfully implemented in *FPS-BHAM* and is capable of providing both qualitative description and quantitative analysis of the overall behaviour in the leakage-soil interaction problem.

The blocked partitioning domain decomposition proposed for this thesis is under test

with the parallelised computing in Section 7.3. The 'pseudo-vector processing capacity' boosts the parallel behaviour of LBM in domain decomposition. A parallel efficiency of 0.72 is achieved with 32 processors, which indicates a good parallel behaviour of DEM-LBM with the blocked partitioning domain decomposition. It is also noted that the parallel efficiency of LBM is higher than that of DEM, suggesting the overall performance is dependent on the ratio of the computing load of LBM to DEM.

In Section 7.4, the verification of pure DEM-DFF computation is firstly carried out. Simulation results from a one-dimensional pressure-driven flow through a porous medium agree well with the analytical solution and the DEM-LBM solutions, as presented in Section 7.4.1. It is then followed by a validation of the combination strategy between DFF and LBM by conducting both one- and two-dimensional pressure-driven flow tests. Numerical results at the steady state demonstrate the non-equilibrium bounce-back scheme provides successful combination between DFF and LBM, while the extrapolation scheme achieves good performance only with a relatively large relaxation time. In addition, before the steady state is reached, it is noted that a difference in pressure distribution appears in the middle sub-domain. Such a difference is due to the different forms of the governing equations in DFF and LBM, which is further explored in Chapter 8. With the presence of particles, a previous pipe leakage test with the DEM-LBM technique in Cui (2012) is reproduced using the combined system. In this test, the non-equilibrium bounce-back scheme is adopted. At

the steady state, a general good agreement on the flow field is achieved between results by DEM-LBM and the combined system. During the dynamic process when a cavity develops at the leaking area, the consistent evolution behaviours of the excess pressures, the orifice velocity, and the cavity size are obtained. However, it is found that the major component of the difference happens in the later stage of the simulation, which may be attributed to the different cavity shapes developed in the two tests. This is because DEM-DFF is not as sensitive as DEM-LBM to the local variety of the particle configurations, and a more symmetrical cavity is formed in the combined system. In spite of the difference, a 42% savings in the computational time is achieved in the combined system for the example used with coarser mesh used for DFF, which indicates a good computational benefit in the proposed combination strategy and a promising capability in simulating the real-scale geotechnical problem with local detailed descriptions.

CHAPTER 8: PARAMETRIC STUDY ON DIFFERENCE BETWEEN LBM AND DFF BEHAVIOURS

8.1 Introduction

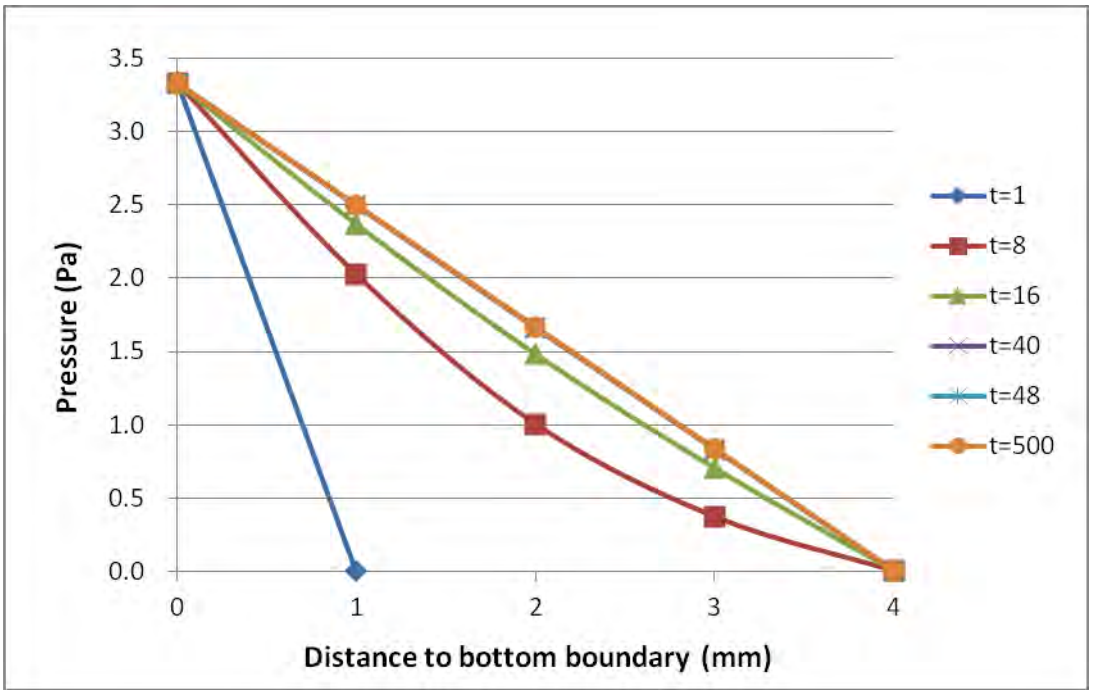
As pointed out in Section 7.4, a different behaviour is observed in the combined system during the dynamic propagation to the steady state, which is mainly caused by the different forms of the flow equations in LBM and DFF. This issue is further explored in this chapter. In Section 8.2, an introduction to the Partial Differential Equations (PDEs) used in LBM and DFF are presented respectively and their different behaviours are demonstrated. In Section 8.3, the standard difference is introduced to describe such a difference. It is then followed by the work conducted to explore the LBM and DFF parameters affecting the standard difference in Sections 8.4 and 8.5, respectively.

8.2 Difference between hyperbolic and parabolic PDEs

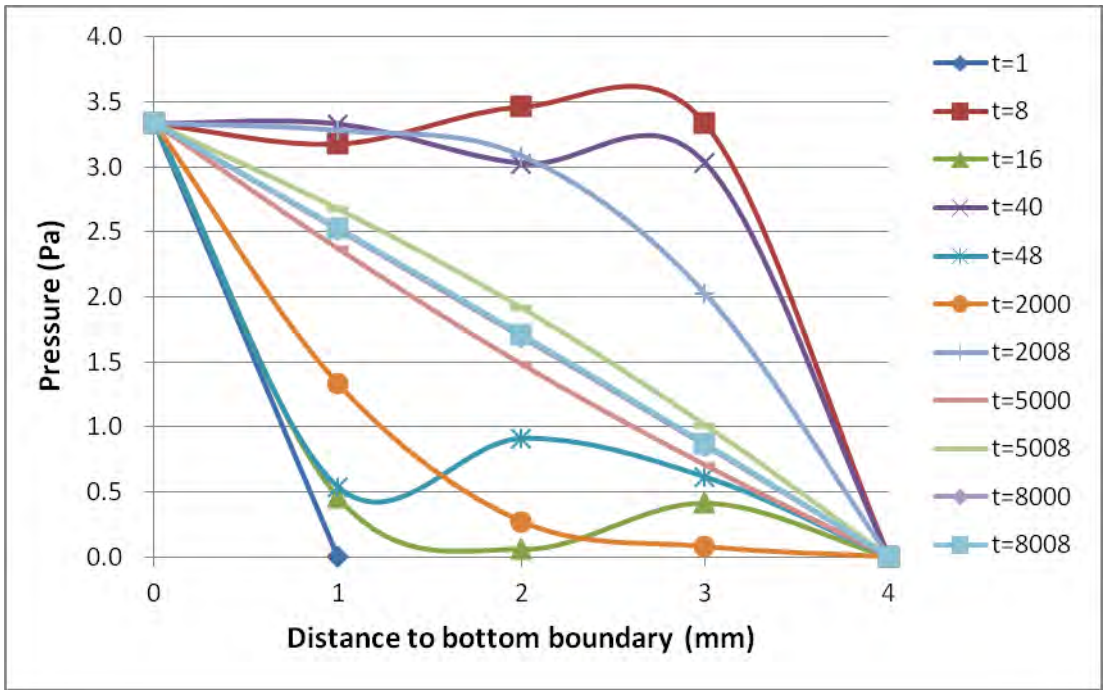
In order to investigate the effect of different PDEs on the flow behaviour, a numerical test is carried out to simulate a one-dimensional pressure-driven flow. The domain size is 0.004m×0.004m, which is divided into a 4×4 grid. A pressure difference $\Delta p = 3\text{Pa}$ is applied across the vertical direction. Pressure gradient across the left and right boundaries are kept constant at zero. Two tests with pure LBM and pure DFF employed are respectively conducted. In the LBM test, the time step is 1×10^{-4}s, which gives the corresponding lattice speed of 10m/s. The kinematic viscosity is 1×10^{-6} m^2/s. And in the

DFF test, the same time step is used so as to make the test results comparable.

The results at various calculation cycles are shown in Figure 8.1, from which it can be seen the dynamic propagations to the steady state are different in DFF and LBM. In the DFF calculations, the flow pattern is controlled by (4.7), which is a parabolic PDE and exhibits diffusion behaviour due to the presence of the concentration gradient (i.e. the applied pressure gradient in the vertical direction). The convergence to the steady solution is very fast, which only takes around 40 calculation cycles. However, in the LBM calculations, the flow pattern is governed by (2.47), which originates from a hyperbolic PDE (i.e. the Boltzmann equation). Such a PDE describe the flow driven by force acting upon it, exhibiting the convection behaviour. From Figure 8.1(b), it is identified that a pressure wave between the top and bottom boundaries propagates with a period of around 16 calculation cycles. During each period, the pressure distribution consecutively reaches its wave crest and the wave trough. Due to the dissipation of the kinetic energy by the fluid viscosity, the amplitude of the wave keeps decreasing until the steady state is achieved. Compared with the DFF results, it can be observed that the number of cycles to reach the steady state is much greater in the LBM calculations. The resulting difference in the pressure distribution before the steady state cannot be ignored, as it would affect the performance of the combination strategy proposed in this thesis. Therefore, further investigation is conducted in the following sections.



(a)



(b)

Figure 8.1: Dynamic propagation to the steady state at different calculation cycles: (a) DFF results; and (b) LBM results

8.3 Quantifying the difference between LBM and DFF behaviours

In order to analyse the difference between LBM and DFF behaviours, a standard measure of difference using a root mean square value is adopted, which is defined as

$$\sigma = \sqrt{\frac{\sum_{i=1}^{i=n} (p_{i,LBM} - p_{i,DFF})^2}{n}}, \tag{8.1}$$

where n is the number of the nodes, $p_{i,LBM}$ and $p_{i,DFF}$ are the pressures at Node i obtained from LBM and DFF calculations, respectively. From the test results demonstrated in Section 8.2, the evolution of the standard measure of difference for the first 50 calculation cycles is calculated and shown in Figure 8.2. It can be found the difference oscillates with a period of around 8 calculation cycles, which is half of that in the wave oscillation. The peak values are taken from both the wave crests and the troughs. In an effort to further investigate the trend of the standard difference, only these peak values are plotted against their corresponding calculation cycles as presented in Figure 8.3. With a much longer time, it is seen that the standard difference undergoes a reduction until it approaches to zero with the steady solution is achieved.

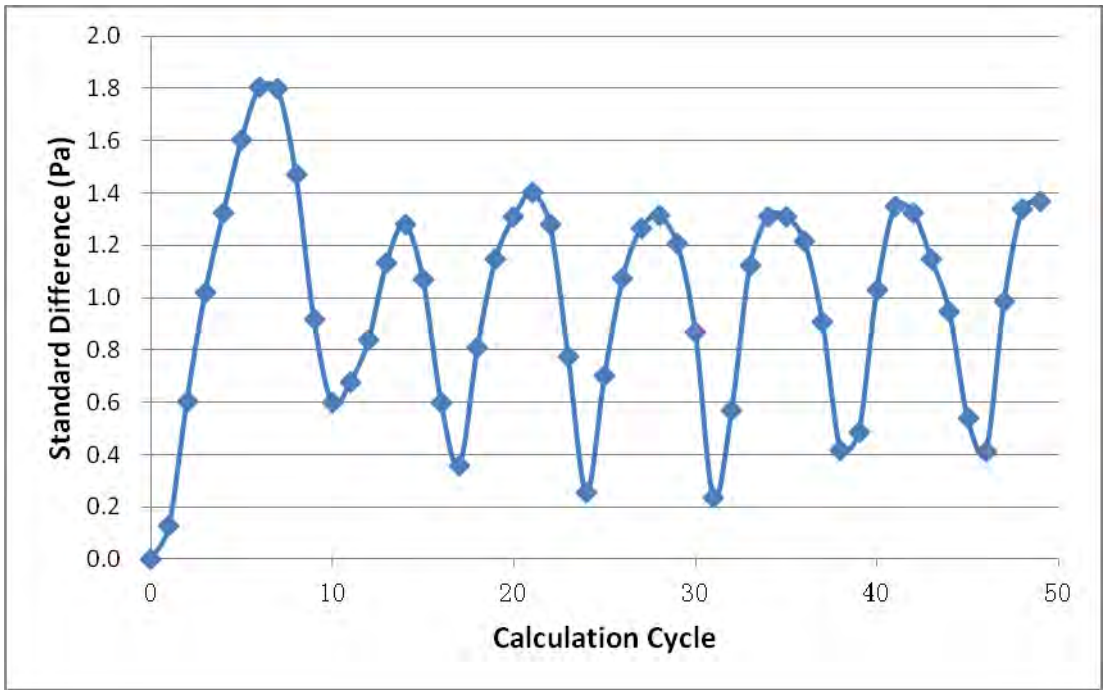


Figure 8.2: Time evolution of the standard difference for the first 50 calculation cycles

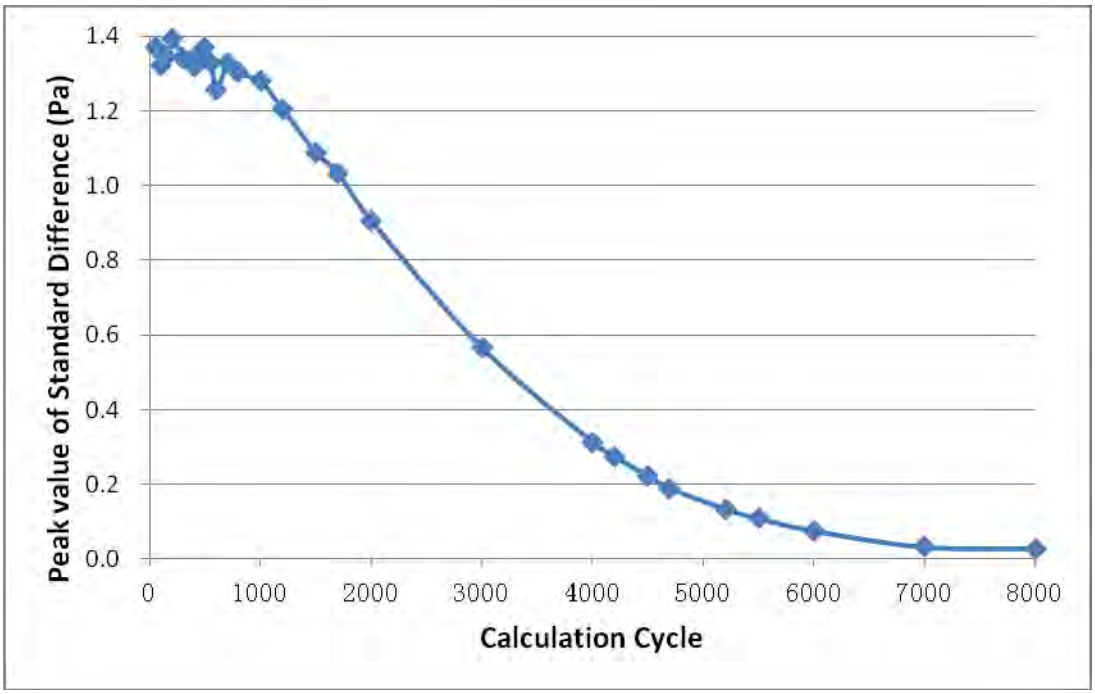


Figure 8.3: Time evolution of peak values in the standard difference

8.4 Parameters in LBM affecting its behaviour and the standard difference

The parameters which may be crucial in determination of the LBM behaviour include: kinematic viscosity, lattice speed, time step, lattice spacing, and relaxation time. In this section, the effects of the above parameters on the LBM behaviour and hence the standard difference are explored.

Kinematic viscosity

In order to investigate the influence of the kinematic viscosity on the convection behavior in LBM, the following tests are performed with the viscosity chosen to be $8\times10^{-6}m^2/s$, $4\times10^{-6}m^2/s$, $2\times10^{-6}m^2/s$, and $1\times10^{-6}m^2/s$, respectively. All other parameters are the same as those in the test reported in Section 8.2. The pressure distribution around 0.1004s are shown in Figure 8.4. For each viscosity, the curve includes both the crest (at 0.1s) and the trough (at 0.1008s). It can be observed that, with the increase in the kinematic viscosity, the amplitude of the wave reduces. This indicates that, for the fluid with a higher viscosity, the kinetic energy is dissipated faster, so that less number of calculation cycles is required to approach to the steady solution. This can also be seen from Figure 8.5 as below.

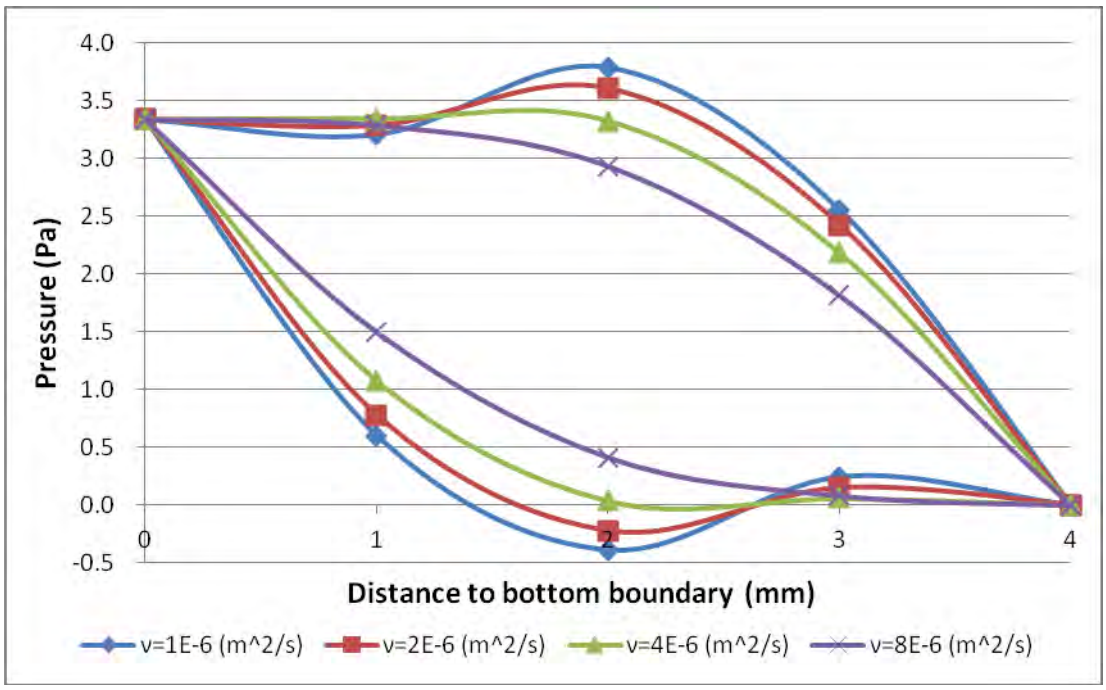


Figure 8.4: Pressure distribution around 0.1004s

The time evolution of the peak values in the standard difference is shown in Figure 8.5. It can be observed that the viscosity influences the rate of reduction in the peak values. With higher viscosity, the standard difference decreases faster as the kinetic energy is dissipated at a higher rate. It is also worth noting from Figure 8.6, the standard difference is almost linearly with the increase in the viscosity at a given time.

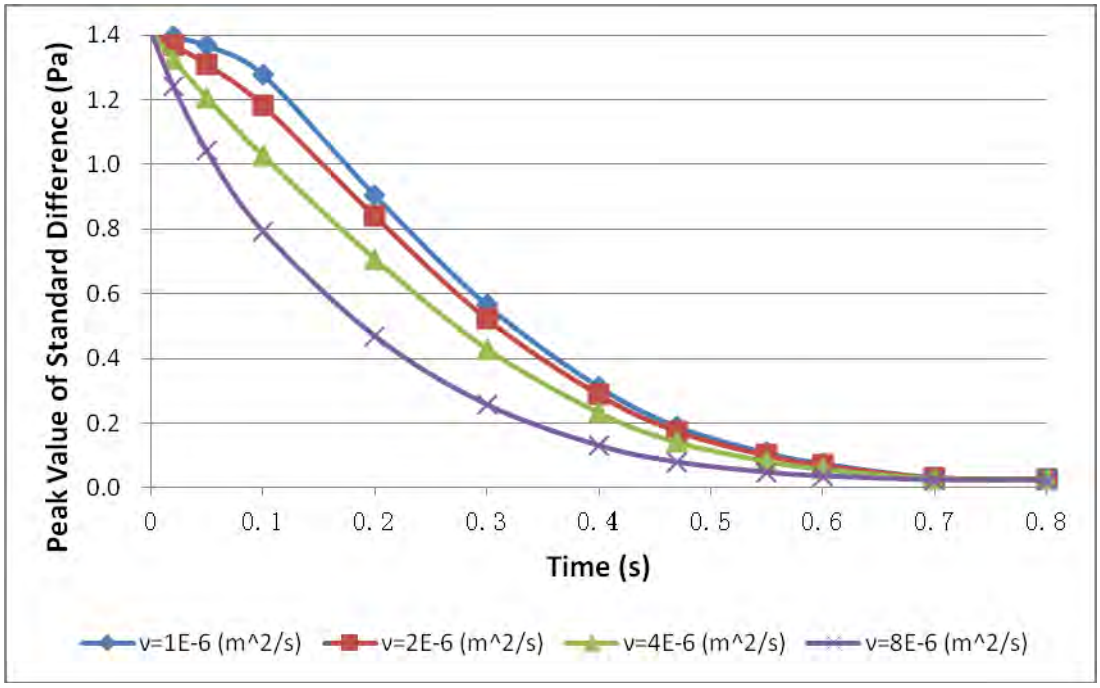


Figure 8.5: Time evolution of peak values in the standard difference with different kinematic viscosities

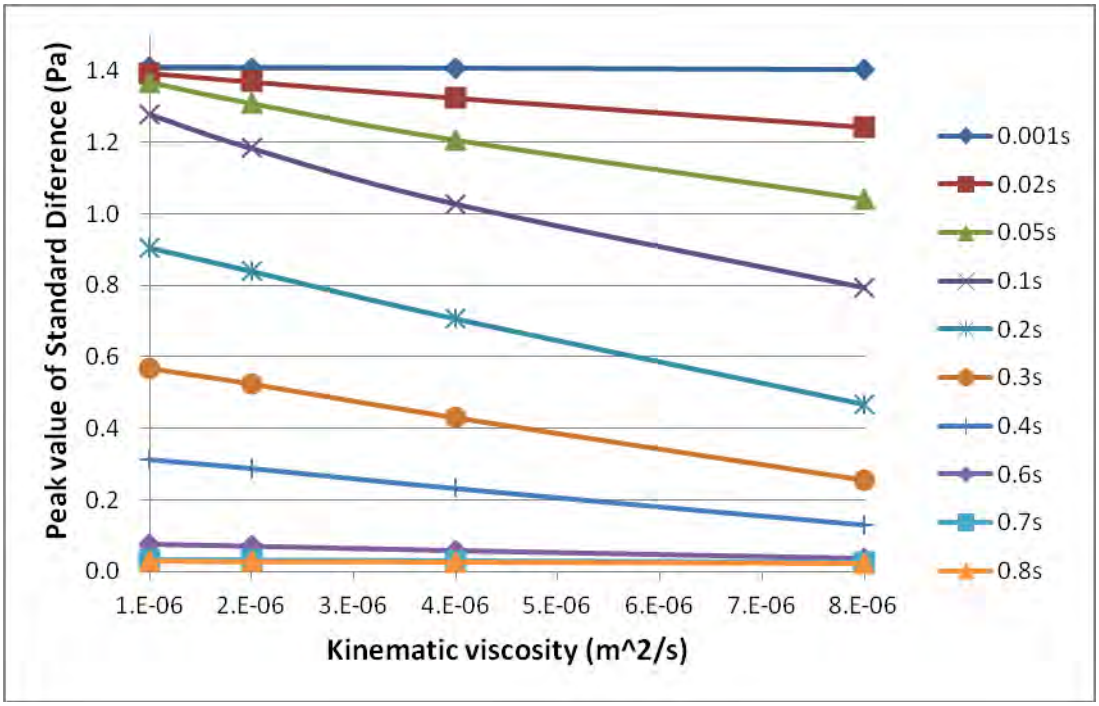


Figure 8.6: Peak value of standard difference verses kinematic viscosity

Lattice speed and time step

From (2.45), it can be identified that, for a given lattice spacing, the value of the lattice speed determines the time step used in the LBM caluclations, and vice versa. Therefore, the influence of these two parameters on the LBM behaviour is investiged together in the following test. The lattice speed is set as 10m/s, 20m/s, 40m/s and 80m/s, so as to give the corresponding time step of 1×10^{-4}s, 0.5×10^{-4}s, 0.25×10^{-4}s, and 0.125×10^{-4}s, respectivley. All other paratmers are the same as those in the test reported in Section 8.2. The results at same cycles (i.e. t = 56 cycles and t = 64 cycles) are shown in Figure 8.7, from which it can be found the identical results are obtained regardless of the variety of lattice speed. It indicates that the energy dissapation within each LBM calculation cycle is independent of the lattice speed for a given lattice spacing. However, as the time steps used are different, the results at a given time are different from one another. Figure 8.8 shows the pressure distribution around 0.1s with different lattice speeds adopted. It is found that the flow behavior changes significantly due to the change in lattice speed. This can be interpreted that, with a higher lattice speed, more periods of wave oscillation occurs during a given time, and hence more energy is disspated so that the steady solution is quicker to be achieved. The peak values in the standard difference is shown in Figure 8.9. It can be found that the lattice speed has a great impact on the standard difference. A higher lattice speed leads to a more sharply reduciton in the standard difference, this is also attributed to the quicker dissipation of the kinetic energy. Compared with the results shown in Figure 8.5, it can be concluded that the reduction in

the difference of LBM and DFF behaivours is more sensitive to the change in the lattice speed rather than in the kinematic viscosity.

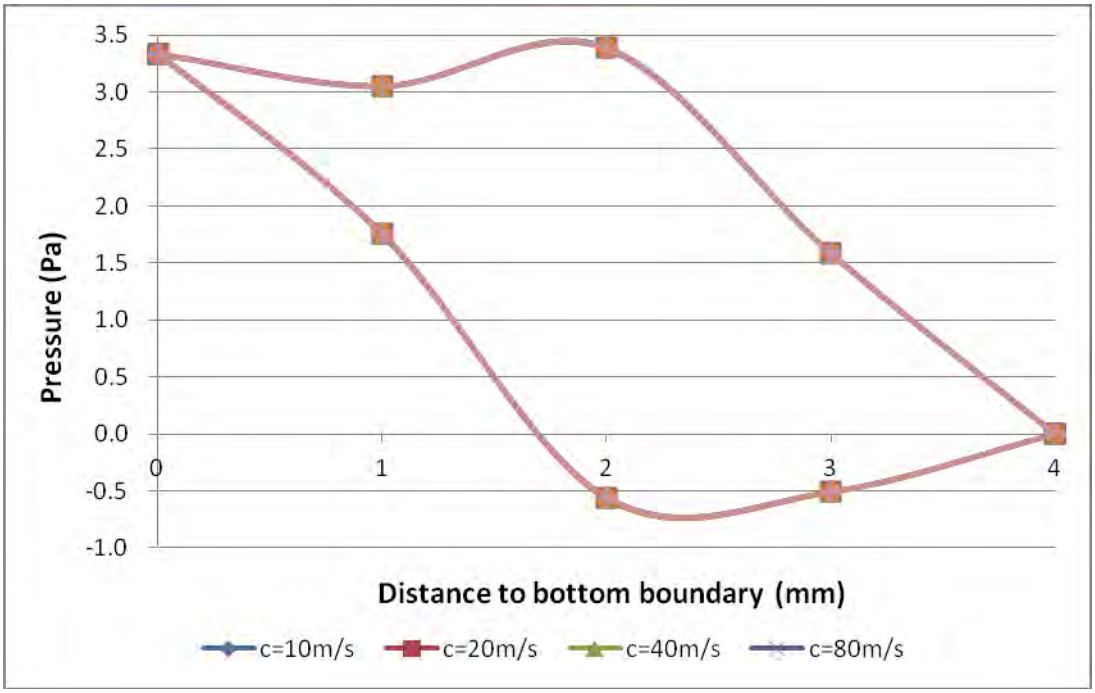


Figure 8.7: Pressure distribution at t = 56 cycles and t = 64 cycles with different lattice speeds

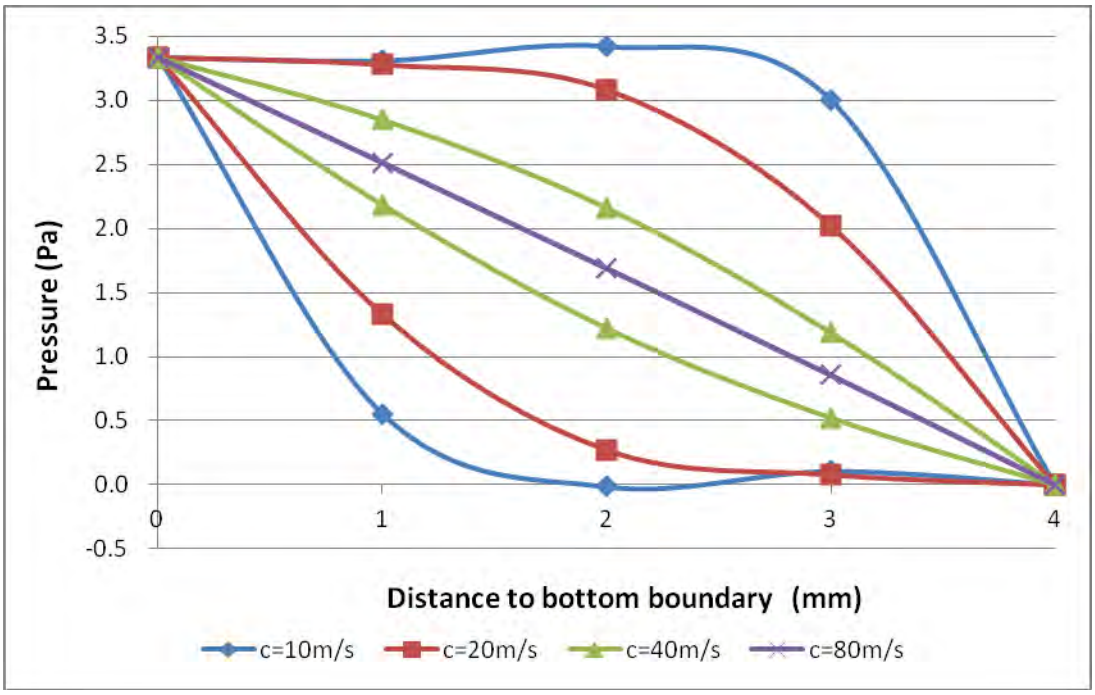


Figure 8.8: Pressure distribution around 0.1s with different lattice speeds

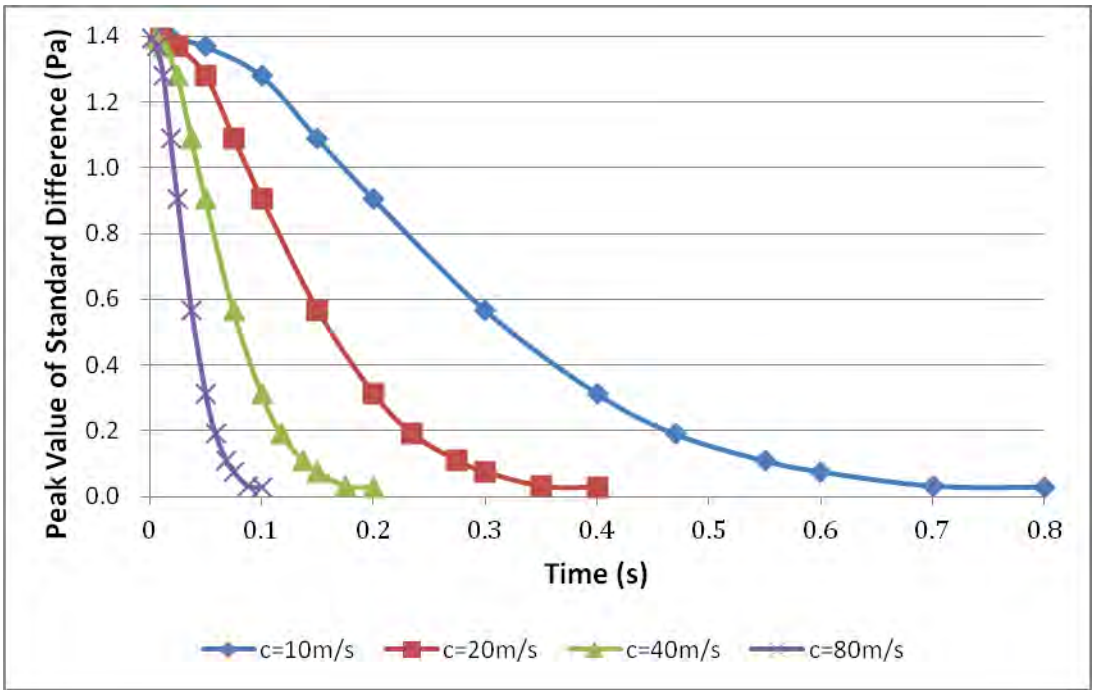


Figure 8.9: Peak values in the standard difference with different lattice speeds

Lattice spacing

The following test is conducted to explore the effect of lattice spacing. The lattice spacing is set to be 0.125mm, 0.25mm, 0.5mm and 1.0mm, respectively. For a given domain size which is same as that in the test reported in Section 8.2, more lattice nodes are adopted for a case with a smaller lattcie spacing for LBM. The time step is also changed correspondingly in order to keep an identical lattice speed, which is 10m/s for all these cases. The pressure distribution around 0.4s with different lattice spacings is shown in Figure 8.10, from which no significant difference can be observed, indicating the lattice spacing has little influence on the convection behavior.

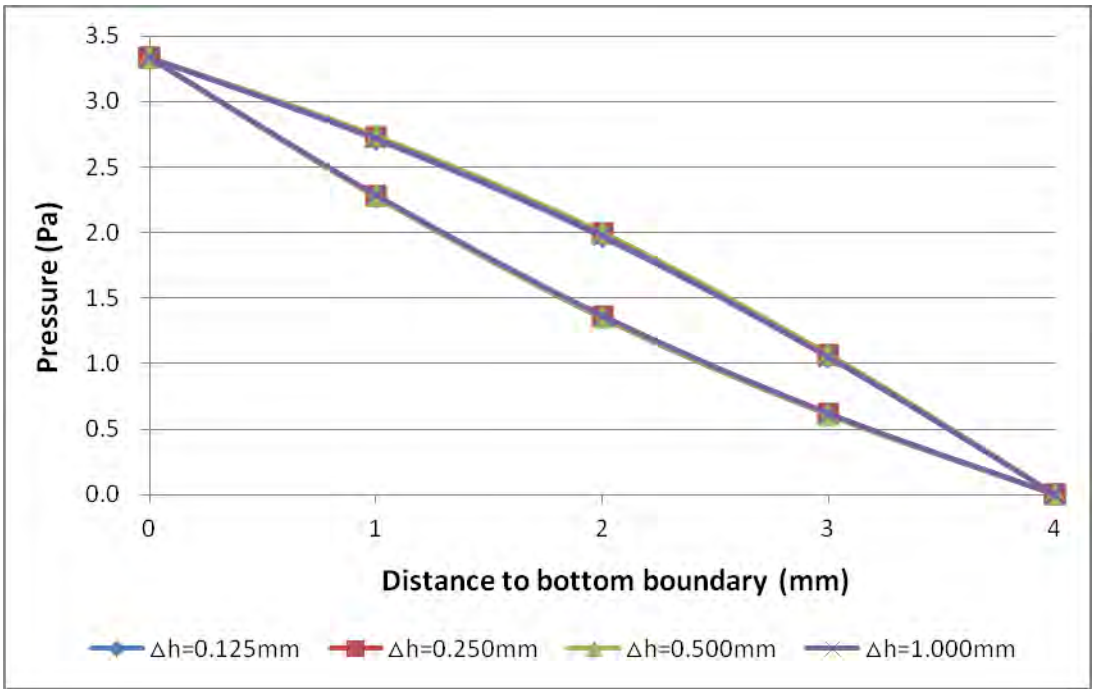


Figure 8.10: Pressure distribution at around 0.4s with different lattice spacings

Relaxation time

From Figure 8.5, it can be noticed that for a given lattice speed and lattice spacing, a quicker reduction in the standard difference is obtained with a higher relaxation time, according to (2.52). However, from Figure 8.9, it is found out for a given kinematic viscosity and lattice spacing, a quicker reduction in the standard difference is achieved with a lower relaxation time. Hence, it can be concluded that there is no direct relationship between the relaxation time and the convection behaviour in LBM.

8.5 Parameters in DFF affecting its behaviour and the standard difference

For the simple form of DFF adopted in this thesis (see (4.7)), the cell size is the only parameter that may affect its diffusion behaviour. In this test, the cell size is set to be 1mm, 0.5mm, 0.25mm and 0.125mm, respectively. By changing the number of cells and the time step, the same domain size and lattice speed are kept as those in the test reported in Section 8.2. The pressure distribution at around 0.04s with different cell sizes is shown in Figure 8.11. Unlike LBM, a significant impact can be observed from the results. With a smaller cell size, the diffusion speed is lower and hence a longer time is required to obtain a stable solution. The evolution of the standard difference during the first 50 calculation cycles is shown in Figure 8.12. It is seen that less violent oscillations in the standard difference are obtained in the test with a larger cell size. From the calculations, the average of the standard difference for each case is 0.97 Pa,

1.03 Pa, 1.1 Pa and 1.2 Pa, respectively. This indicates that with a smaller cell size, a larger difference between LBM and DFF behaviours is, in general, resulted.

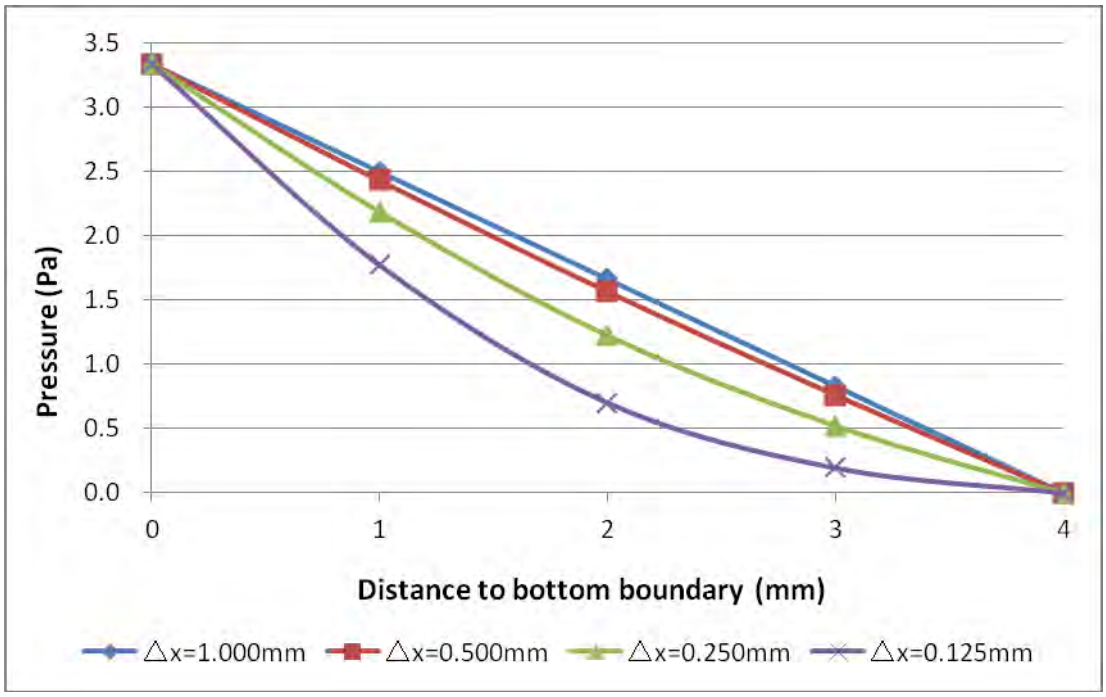


Figure 8.11: Pressure distribution at around 0.04s with different cell sizes

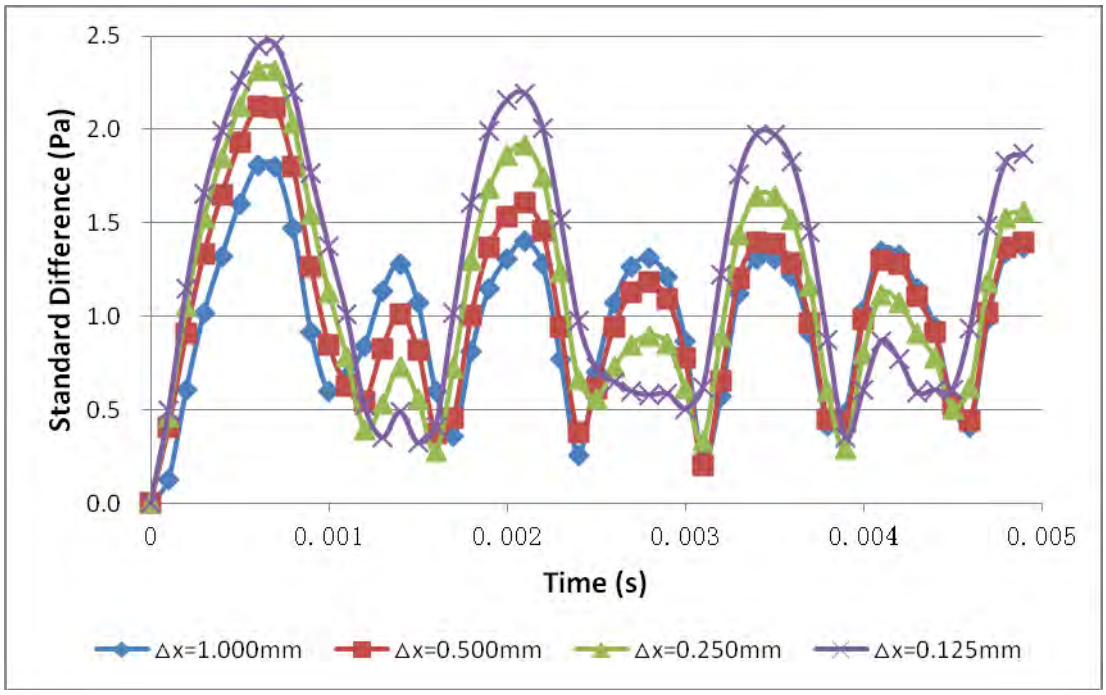


Figure 8.12: Evolution of the standard difference during the first 50 calculation cycles

8.6 Summary

This chapter explores the different behaviours in the combination strategy during the dynamic propagation to the steady state. In the DFF calculations, the flow pattern is controlled by a parabolic PDE and exhibits diffusion behaviour. While in the LBM calculations, the flow pattern is governed by a hyperbolic PDE. Such a PDE describes a flow motivated by force acting upon it, exhibiting the convection behaviour. Compared with the DFF results, it can be observed that the number of cycles to reach the steady state is much greater for the LBM calculations.

The standard difference is adopted to analyse the difference between LBM and DFF

behaviours. It can be found the standard measure of difference oscillates with a period of around 8 calculation cycles, which is half of that in the wave convection. The peak values are taken from both the wave crests and the hollows. With a much longer time, it is seen the standard difference undergoes a reduction until it approaches to zero with the steady solution is achieved.

For a fluid with a higher viscosity, the kinetic energy is dissipated faster, so that less calculation cycles are required to approach to a steady solution, and hence the standard difference decreases faster. It is also found that the standard difference is almost linearly reduced with the increase in the kinematic viscosity at a given time. Moreover, the identical results are obtained after the same number of calculation cycles regardless of the variety of lattice speed. With a higher lattice speed, more periods of wave oscillation are performed during a given time, and hence more energy is disspated so that the steady solution is quicker to be achieved. From the result, it is also noted that the reduction in the difference between LBM and DFF behaivours is more sensitive to the change in the lattice speed rather than in the kinematic viscosity. On the other hand, the lattice spacing has little influence on the LBM convection behavior. And there is no direct relationship being observed between the relaxation time and the convection behaviour, either. For DFF, with a smaller cell size, a larger difference between LBM and DFF behaviours is resulted.

From the work in this chapter, it is suggested that for the combination between DFF and LBM simulation, a fluid with a higher kinematic viscosity has less differernce during the dynamic propagation to the steady state. In addition, by increasing the lattice speed, the difference can be controlled in a most effeicient way.

CHAPTER 9: CONCLUSIONS AND FUTURE WORK

9.1 Conclusions of this thesis

The thesis is motivated by developing a numerical tool, *FPS-BHAM*, in exploring the large-scale fluid-particle phenomenon with the local behaviours being captured, such as a soil bed subjected to a local leakage. Although it is regarded as one of the most efficient techniques in a small-scale modelling, DEM-LBM is very expensive when applied to large-scale simulations. For this reason, a blocked partitioning domain decomposition strategy with the philosophy of parallel computing of DEM-LBM is adopted. Moreover, the combination of DEM-LBM and a large-scale simulation tool, i.e. DEM-DFF, is also proposed in this thesis.

.In DEM, the motion of particles is determined based on the Newton's second law. The force and torque acting on a single particle are generated from the contacts, the body force (for force only), and the fluid flows. For the fluid phase, LBM with LES turbulence model is employed to simulate the fluid flows with high Reynolds number. The fluid-wall and fluid-particle interactions are solved with the aid of the half-spacing bounce-back rule and the IMB scheme, respectively. Verification of the DEM-LBM implementation and examination of its capability in modelling leakage-soil interaction are conducted in a later section, i.e. Section 7.2. In the DEM-LBM verification, comparisons are firstly made between the *FPS-BHAM* results and the *TRUBAL* results

on the DEM calculations. Satisfactory agreement is obtained, indicating effective DEM computations in *FPS-BHAM*. It is followed by simulations of Couette flows, and the results agree well with the analytical solutions. This suggests the verified implementation of LBM with IMB in *FPS-BHAM*. Afterwards, the simulation of an internal fluidisation problem is carried out, and the results are validated against previous experimental findings. Good qualitative agreements are achieved. Furthermore, the fluidising pressures measured from the numerical tests are found to match the predicted ones from the derived equation well. All the above verifications indicate that the DEM-LBM technique has been successfully implemented in *FPS-BHAM*, and it is capable of providing both qualitative description and quantitative analysis of the overall behaviour in the leakage-soil interaction problem.

The proposed blocked partitioning domain decomposition strategy with the parallel computing is illustrated in Chapter 5. The same decomposition scheme is employed for both the fluid and DEM calculations to avoid extra data communication among processors when considering fluid-particle interactions. Due to the significant difference in CPU time consumed between DEM-LBM and DEM-DFF, the numbers of both types of sub-domains are almost identical in each processor, so as to balance the working load. Fluid within each sub-domain is regarded as an isolated system, and solved independently by its governing equation and boundary conditions. Virtual nodes, which overlap the corresponding nodes in the neighbouring sub-domains, are set as the

sub-domain boundaries to guarantee the continuity requirement. Data at virtual nodes could be obtained from calculations in the corresponding nodes from the neighbouring sub-domains, and are interpreted as boundary conditions. MPI is used to facilitate communication when two sub-domains belong to different processors. For the solid phase, once mapped onto different sub-domains according to centre coordinates, particles are stored and calculated only in their local sub-domains. All the information of a particle within an overlapping area, including its contacts, is required to be transferred among all related sub-domains. The performance of such a parallel scheme is evaluated in Section 7.3. From the evaluation, it is found out that the 'pseudo-vector processing capacity' boosts the parallel behaviour of LBM in domain decomposition. A parallel efficiency of 0.72 is finally achieved with 32 processors, which indicates a good parallel behaviour of DEM-LBM with the blocked partitioning domain decomposition.

The excess pressure in DFF is solved by the Laplace's equation, and the fluid force is correspondingly computed. This force is then used in the DEM calculations. For the combination of DFF and LBM, a strategy is developed to facilitate the two-way conversion between macroscopic and microscopic variables used in DFF and LBM. The conversion from density distribution functions in LBM to pressure in DFF can be achieved by a direct summation, while the inversed process is not straightforward as more unknowns exist in LBM. Therefore, the extrapolation scheme and the non-equilibrium bounce-back scheme are separately employed to solve this problem.

When particles are involved in the combined system, the bell-shaped smoothing function is adopted to determine the data on the lattice nodes covered by particles. In addition, for a particle located at the LBM/DFF interface, the fluid force adopts the value which is calculated from the DFF calculations. The validation of the combination strategy is presented in Section 7.4. Simulation results from a one-dimensional pressure-driven flow through a porous medium by DEM-DFF agree well with the analytical solution and the DEM-LBM solutions. It is then followed by a validation of the combination strategy between DFF and LBM by simulating both one- and two-dimensional pressure-driven flows. Numerical results at the steady state demonstrate the non-equilibrium bounce-back scheme provides successful combination between DFF and LBM, while the extrapolation scheme achieves good performance only with a relatively large relaxation time. Finally, a previous pipe leakage test with the DEM-LBM technique is reproduced using the combined system. Evolution behaviours of the excess pressures, the orifice velocity, and the cavity size are obtained consistent with the results from pure DEM-LBM tests. Different cavity shapes are identified, which is attributed to different sensitivities of DEM-DFF and DEM-LBM to the local variety in the particle configurations. In spite of the difference, a 42% save in computational time is achieved in the combined system, which indicates a good computational benefit provided by the proposed combination strategy.

Different behaviours between LBM and DFF in the combination strategy during the

dynamic propagation to the steady state are investigated in Chapter 8. The flow pattern is controlled by a parabolic PDE and exhibits diffusion behaviour in DFF. While in LBM, the flow pattern is governed by a hyperbolic PDE, and exhibits the convection behaviour. Compared with DFF, more calculation cycles are required to reach the steady state in LBM. The standard difference is adopted to analyse the difference between LBM and DFF behaviours at a given time. With a sufficiently long time, it is seen the standard difference undergoes a reduction until it approaches to zero with the steady solution is achieved. Parametric studies indicate that, for a fluid with a higher viscosity, the kinetic energy is dissipated faster, resulting a faster decrease in the standard difference. With a higher lattice speed, more periods of wave oscillation are performed during a given time, and hence more energy is disspated so that the steady solution is quicker to be achieved. From the result, it is also noted that the reduction in the difference between LBM and DFF behaivours is more sensitive to the change in the lattice speed rather than in the kinematic viscosity. On the other hand, the lattice spacing has little influence on the LBM convection behavior. And there is no direct relationship being observed between the relaxation time and the convection behaviour, either. For DFF, in the tests with a smaller cell size, a general larger difference between LBM and DFF behaviours is found to be induced. From the investigations, it is suggested that by increasing the lattice speed, the difference can be controlled in a most effeicient way.

9.2 Future work

The research presented in this thesis is an early attempt to develop the numerical tool *FPS-BHAM* for exploring the large-scale fluid-particle phenomenon with the local behaviours being captured. The future work related to this thesis could include:

a) Dynamic combination system. In this stage, the arrangement of different techniques in the combined system is predetermined by users according to experience of the physical phenomenon or the information of previous DEM-LBM simulations. However, for most fluid-particle systems, the flow behaviour in each sub-domain changes dynamically. This may results in the conservativeness in the arrangement of numerical techniques, and the optimum performance of the combined strategy is difficult to be achieved. In addition, for the simulations with complicated mechanism, cumbersome trail process before the simulation is often required to obtain the overall description of the phenomenon. Such limitations invoke the motivation of developing the dynamic combination system in the future, in which the arrangement of numerical techniques is conducted automatically and dynamically according to the features of flow behaviors in each sub-domain.

b) Explicit determination of Permeability. In this thesis, the simple form DFF is governed by the Laplace's equation, which is based on the assumption of a uniform permeability over the sample. For this reason, only one-way coupling between DEM and

DFF is adopted in this thesis. However, the real soils often exhibit heterogeneous or even anisotropic hydraulic behaviour. Hence, explicit consideration of permeability based on particle configuration is necessary in many geotechnical applications, which may lead to a two-way coupling between DEM and DFF in the future.

c) In the current two-dimensional simulations, fluid paths through the particle assembly are considered by artificially adopting the hydraulic radius. This makes the determination of permeability ambiguous, although it is sometimes necessary to obtain the flow field in soil. Moreover, the current model is difficult to give an explicit description of turbulent flow which involved at the vicinity of the orifice, as it is actually a three-dimensional phenomenon. Therefore, it is expected to extend the simulations by *FPS-BHAM* to three-dimensional ones, by which the physical behaviour can be captured more accurately.

REFERENCES

Alejandro, J.C.C. (2009). Application of the Smoothed Particle Hydrodynamics model SPHysics to free-surface hydrodynamics. PhD thesis, Universidade De Vigo.

Alsaydalani, M.O.A. (2010). Internal fluidisation of granular material. PhD thesis, University of Southampton.

Anderson, T.B. and Jackson, R. (1967). A fluid mechanical description of fluidised beds. I&EC Fundamentals, **6**, 527-39.

Blaise, B. (2012a). OpenMP. https://computing.llnl.gov/tutorials/openMP.

Blaise, B. (2012b). Message Passing Interface. https://computing.llnl.gov/tutorials/mpi.

Bonilla, R.R.O. (2004). Numerical Simulations of Undrained Granular Media. PhD thesis, University of Waterloo.

Boutt, D.F., Cook, B.K., McPherson, B.J.O.L., and Williams, J.R. (2007). Direct simulation of fluid-solid mechanics in porous media using the discrete element and lattice-Boltzmann methods. Journal of Geophysical Research B: Solid Earth, **112**,

B10209.

Campbell, C.S. and Brennen, C.E. (1985). Computer simulations of granular shear flows. Journal of Fluid Mechanics, **151**, 167–188.

Charbeneau, R.J. (2006). Groundwater hydraulics and pollutant transport. Waveland Press.

Chen, L., Luan, H.B., Feng, Y., Song, C., He, Y.L. and Tao, W.Q. (2011). Coupling between finite volume method and lattice Boltzmann method and its application to fluid flow and mass transport in proton exchange membrane fuel cell. International Journal of Heat and Mass Transfer, **55**, 3834-3848.

Chen, S. and Doolen, G.D. (1998). Lattice Boltzmann method for fluid flows. Annual Review of Fluid Mechanics, **30**, 329–364.

Chen, S., Martinez, D. and Mei R. (1996). On boundary conditions in lattice Boltzmann methods. Physics of Fluids, **8**(9), 2527-2527.

Cook, B.K., Noble, D.R., Preece, D.S., and Williams J.R. (2000). Direct simulation of particle-laden fluids. Pacific Rocks 2000, 279-286.

Cook, B.K., Noble, D.R., and Williams, J.R. (2004). A direct simulation method for particle-fluid systems. Engineering Computations, **21**(2-4), 151-168.

Cui, X. (2012). Numerical simulation of internal fluidisation and cavity evolution due to a leaking pipe using the coupled DEM-LBM technique. PhD thesis, University of Birmingham.

Cui, X., Li, J., Chan, A.H. and Chapman D. (2012). A 2D DEM-LBM study on bed behaviour due to a locally injected fluid. Particuology, **10**(2), 242-252.

Cundall P.A. (1971). A computer model for simulating progressive, large-scale movements in blocky rock systems. Proceedings Symposium of international society for rock mechanics, **2**, 132-150.

Cundall, P. A. (1988). Computer simulations of dense sphere assemblies. Micromechanics of granular materials, Satake and Jenkins (eds), Elsevier Science Publishers, Amsterdam, Netherlands, 113-123.

Cundall, P. A. (2001). A discontinuous future for numerical modelling in geomechanics. Geotechnical Engineering, **149**(1), 41 – 47.

Cundall, P. A. and Hart, R. D. (1992). Numerical modeling of discontinua. Engineering Computations, **9**, 101-113.

Cundall, P.A. and Strack, O.D.L. (1978). The discrete element method as a tool for research in granular media. Part I. Report to National Science Foundation, University of Minnesota.

Cundall, P.A. and Strack, O.D.L. (1979). The discrete element method as a tool for research in granular media. Part II. Report to National Science Foundation, University of Minnesota.

Darmana, D., Deen N.G. and Kuipers, J.A.M. (2006). Parallelisation of a Euler–Lagrange model using mixed domain decomposition and a mirror domain technique: Application to dispersed gas–liquid two-phase flow. Journal of Computational Physics, **220**(1), 216-248.

Deen, N.G., Van Sint Annaland, M., Van der Hoef, M.A. and Kuipers, J.A.M. (2007). Review of discrete particle modeling of fluidised beds. Chemical Engineering Science, **62**(1-2), 28-44.

Delenne J.Y., E.l. Youssoufi M.S., Cherblanc F., and Bénet J.C. (2004). Mechanical behaviour and failure of cohesive granular materials. International Journal For Numerical And Analytical Methods In Geomechanics **28**, 1577-1594.

Di Felice, R. (1994). The voidage function for fluid–particle interaction systems. International Journal of Multiphase Flow, **20**, 153–159.

Duran, J. (2000). Sands, powders, and grains: An introduction to the physics of granular materials. Springer.

Eggels, J.G.M. (1996). Direct and large-eddy simulation of turbulent fluid flow using the lattice- Boltzmann scheme. International Journal of Heat and Fluid Flow, **17**, 307–323.

Ergun; S. (1952). Fluid flow through packed columns. Chemical Engineering Progress, **43**(2), 89-94.

Farley, M. (2001). Leakage management and control: A best practice training manual. WHO/SDE/WSH/01.1 © 2001, WHO.

Feng Y.T., Han K. and Owen D.R.J. (2007). Coupled lattice Boltzmann method and discrete element modeling of particle transport in turbulent fluid flows: Computational issues. International Journal for Numerical Methods in Engineering, **72**(9), 1111-1134.

Feng Y.T., Han K., and Owen D.R.J. (2010). Combined three-dimensional lattice Boltzmann method and discrete element method for modelling fluid–particle interactions with experimental assessment. International Journal for Numerical Methods in Engineering, **81**(2), 229–245.

Feng, Y.Q. and Yu, A.B., (2004). Assessment of model formulations in the discrete particle simulation of gas–solid flow. Industrial and Engineering Chemistry Research, **43**, 8378–8390.

Filippova, O. and HÄanel, D. (1998). Grid refinement for lattice-BGK models. Journal of Computational Physics, **147**, 219-228.

Ghaboussi, J., Basole, M. and Ranjithan, S. (1993). Three dimensional discrete element analysis on massively parallel computers. Proceedings of the Second International Conference on Discrete Element Methods.

Ghassemi, A. and Pak, A. (2011). Coupled lattice Boltzmann - Discrete Element method

for numerical modelling of sand production, Particle-Based Methods II - Fundamentals and Applications, 371-382.

Gingold, R.A. and Monaghan, J.J. (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars. Monthly Notices Royal Astronomical Society, **181**, 375–389.

Gong, G.B. (2008). DEM simulations of drained and undrained behaviour. PhD thesis, University of Birmingham.

Hakuno M. and Tarumi Y. (1988). Granular assembly simulation for the seismic liquefaction of sand. Doboku Gakkai Rombun-Hokokushu/Proceedings of the Japan Society of Civil Engineers, **398**/I-10, 129-138.

Han, K., Feng, Y.T. and Owen D.R.J. (2006). Polygon-based contact resolution for superquadrics, International Journal for Numerical Methods in Engineering, **66**, 485-501.

Han, K., Feng, Y.T., and Owen D.R.J. (2007a). Coupled lattice Boltzmann and discrete element modeling of fluid-particle interaction problems. Computers and Structures, **85**(11-14), 1080-1088.

Han, K., Feng, Y.T., and Owen D.R.J. (2007b). Numerical simulations of irregular particle transport in turbulent flows using coupled LBM-DEM. Computer Modeling in Engineering and Sciences, **18**(2), 87-100.

He X., Zou Q., Luo L.S. and Dembo M. (1997). Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model. Journal of Statistical Physics, **87**(1-2), 115-136.

Hoomans, B.P.B., Kuipers, J.A.M., Briels, W.J. and Van Swaaij, W.P.M. (1996). Discrete particle simulation of bubble and slug formation in a two-dimensional gas-fluidised bed: a hard-sphere approach. Chemical Engineering Science, **51**(1), 99–118.

Hou S., Sterling J., Chen S. and Doolen G.D. (1996). A lattice Boltzmann subgrid model for high Reynolds number flows. Fields Institute Communications, **6**, 151-166.

Hu, H.H. (1996). Direct simulation of flows of solid–liquid mixtures. International Journal of Multiphase Flow, **22**, 335–352.

Inamuro, T. and Sturtevant, B. (1990). Numerical study of discrete-velocity gases.

Physics of Fluids, **2**, 2196–2203.

Israelachvili, J.N. (1991). Intermolecular and Surface Forces. Academic Press.

Jensen R.P. and Preece D.S. (2000). Modeling sand production with darcy-flow coupled with discrete elements. Sandia National Laboratories Reports.

Johnson K.L. (1985). Contact Mechanics. Cambridge University Press.

Kafui, K.D., Johnson, S., Thornton, C. and Seville, J.P.K. (2011). Parallelisation of a Lagrangian-Eulerian DEM/CFD code for application to fluidised beds. Powder Technology, **207**(1-3), 270-278.

Kawaguchi, T., Tanaka, T. and Tsuji, Y. (1998). Numerical simulation of two-dimensional fluidised beds using the discrete element method (comparison between the two- and three-dimensional models). Powder Technology, **96**(2), 129–138.

Kumar, V., Grama, A., Gupta, A., and Karypis, G. (1994). Introduction to parallel computing - design and analysis of algorithms. The Benjamin/Cummings Publishing Company.

Kunkel, G., Laven, K. and Mergelas, B. (2008). Field report -- does your city have high-risk pipes? Journal AWWA, **100**(4), 70-74.

Kwon, Y.W. and Hosoglu, S. (2008). Application of lattice Boltzmann method, finite element method, and cellular automata and their coupling to wave propagation problems. Computers & Structures, **86**(7-8), 663-670.

Lallemand, P. and Luo, L.S. (2000). Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. Physical Review E, **61**(6), 6546-6562.

Lavall´ee, P., Boon J.P., and Noullez, A. (1991). Boundaries in lattice gas flows. Physica D: Nonlinear Phenomena, **47**, 233–240.

Leonardi, C. R., Owen, D. R. J. and Feng, Y. T. (2011). Numerical rheometry of bulk materials using a power law fluid and the lattice Boltzmann method, Journal of Non-Newtonian Fluid Mechanics, **166**(12-13), 628-638.

Li, J., Cui, X., Bridgeman, J. and Chan, A.H. (2010). An innovative method coupling Darcy Fluid Flow and Lattice Boltzmann Method. Proceedings of the 18th UK National Conference of the Association for Computational Mechanics in Engineering, 191-194.

Li, S.F., and Liu, W.K. (2007). Meshfree particle methods. Springer.

Li, X.K., Chu, X.H., and Sheng, D.C. (2007). A saturated discrete particle model and characteristic-based SPH method in granular materials. International Journal For Numerical Methods In Engineering, **72**, 858–882.

Li, Y., Yang, G.Q., Zhang, J.P. and Fan, L.S. (2001). Numerical studies of bubble formation dynamics in gas–liquid–solid fluidization at high pressures. Powder Technology, **116**(2–3), 246–260.

Lin, X. and Ng, T.-T. (1997). A three-dimensional discrete element model using arrays of ellipsoids. Geotechnique, **47**(2), 319-329.

Liu, M.B. and Liu, G.R. (2010). Smoothed particle hydrodynamics (SPH): An overview and recent developments. Archives of Computational Methods in Engineering, **17**(1), 25-76.

Lucy, L. (1977). A numerical approach to testing the fission hypothesis. The Astronomical Journal, **82**, 1013-1024.

Lynch, M. and Stimpson, J. (2011). Network Rail battles to cure Croydon lines after mudslide, New civil engineer/18.08.11, 5.

Krafczyk, M. (2001). Gitter-Boltzmann-Methoden: Von der Theorie zur Anwendung. PhD thesis, FakultÄat fÄur Bauingenieur- und Vermessungswesen der Technischen UniversitÄat MÄunchen.

Makar, J.M. (2000). A preliminary analysis of failures in grey cast iron water pipes. Engineering Failure Analysis, **7**(1), 43-53.

Malvern, L. (1969). Introduction to the mechanics of a continuous medium. Prentice-Hall.

Mansouri M., Delenne J.Y., El Youssoufi M.S. and Seridi A. (2009). A 3D DEM-LBM approach for the assessment of the quick condition for sands. Comptes Rendus Mecanique, **337**(9-10), 675-681.

Mansouri M., Delenne J.Y., Seridi A. and El Youssoufi M.S. (2010). Numerical model for the computation of permeability of a cemented granular material. Powder Technology, **208**(2), 532-536.

McDowell, G.R. and Harireche, O. (2002). Discrete element modeling of soil particle fracture. Geotechnique, **52**(2), 131–135.

Mikami, T., Kamiya, H. and Horio, M. (1998). Numerical simulation of cohesive powder behavior in a fluidised bed. Chemical Engineering Science, **53**, 1927–1940.

Mindlin, R.D. (1949). Compliance of elastic bodies in contact. Journal of Applied Mechanics, **16**, 259-268.

Mindlin, R.D. and Deresiewicz, H. (1953). Elastic spheres in contact under varying oblique forces. Journal of Applied Mechanics, **20**, 327-344.

Morris, J. and Johnson, S. (2009). Dynamic simulations of geological materials using combined FEM/DEM/SPH analysis. Geomechanics and Geoengineering, **4**(1), 91-101.

Munjiza A. (2004). The Combined Finite-Discrete Element Method. John Wiley & Sons Ltd.

Munjiza, A. and Andrews, K.R.F. (1998). NBS contact detection algorithm for bodies of similar size. International Journal for Numerical Methods in Engineering, **43**, 131-149.

Nakase, H., Takeda, T. and Oda, M.A. (1999). Simulation study on liquefaction using DEM. Earthquake Geotechnical Engineering, **3**, 637-642.

Ng, T.T. and Dobry, R. (1994). Numerical simulations of monotonic and cyclic loading of granular soil. Journal of Geotechnical & Geoenvironmental Engineering, **120**(2), 388–403.

Noack, C. And Ulanicki, B. (2006). Modelling of soil deffusibility on leakage characteristics of burried pipes. 8th Annual Water Distribution Systems Analysis Symposium.

Noble D. and Torczynski J. (1998). A lattice Boltzmann method for partially saturated cells. International Journal of Modern Physics C, **9**, 1189–1201.

Ohtsuki, S. and Matsuoka, T. (2008). The behavior of sand grains around the perforation channel by coupled LBM and DEM. 42nd U.S. Rock Mechanics - 2nd U.S.-Canada Rock Mechanics Symposium.

Onishi, J., Kawasaki, A., Chen, Y. and Ohashi, H. (2008). Lattice Boltzmann simulation of capillary interactions among colloidal particles. Computers & Mathematics with

Applications, **55**(7), 1541-1553.

Owen, D., Leonardi, C.R. and Feng, Y.T. (2011). An efficient framework for fluid-structure interaction using the lattice Boltzmann method and immersed moving boundaries. International Journal for Numerical Methods in Engineering, **87**(115), 66-95.

O'Connor, R. M., Torczynski, J. R., Preece, D. S., Klosek, J. T., and Williams, J. R. (1997). Discrete element modeling of sand production, International Journal of Rock Mechanics and Mining Sciences, **34**, 3-4, 373.

O'Sullivan, C. (2011). Particle-Based Discrete Element Modeling: Geomechanics Perspective. International Journal of Geomechanics. **11**, 449-465.

Pan C., Luo L.S. and Miller C.T. (2006). An evaluation of lattice Boltzmann schemes for porous medium flow simulation. Computers & Fluids, **35**, 898–909.

Pan, T.W., Joseph, D.D., Bai, R., Glowinski, R. and Sarin, V. (2002). Fluidization of 1204 spheres: simulation and experiment. Journal of Fluid Mechanics, **451**, 169–191.

Patankar, S.V. (1980). Numerical heat transfer and flurd flow. Hem-Isphere.

Preece, D.S., Jensen, R.P., Perkins, E.D. and Williams, J.R. (1999). Sand production modeling using superquadric discrete elements and coupling of fluid flow and particle motion. Proceedings of the 37th U.S. Rock Mechanics Symposiu.

Premnath, K.N., Nave, J.C. and Banerjee, S. (2005). Computation of multiphase flows with Lattice Boltzmann methods. American Society of Mechanical Engineers, Fluid Engineering Division, **261**, 403-420.

Puust, R., Kapelan, Z., Savic, D.A. and Koppel, T. (2010). A review of methods for leakage management in pipe networks. Urban Water Journal, **7**(1), 2010.

Rajani, B. and McDonald, S. (1994). Water main break data on different pipe materials for 1992 and 1993. National Research Council, Ottawa, Ontario.

Reed, A. (2007). Notes for the participants of the Blue Bear Open Service. https://intranet.birmingham.ac.uk/it/teams/infrastructure/fm/bear/bluebear/docs.aspx.

Rogers, B. and Dalrymple, R.A. (2004). SPH modeling of breaking waves. Proceedings of the 29th International Conference on Coastal Engineering, 415-427.

Rogers, C.D.F., Chapman, D.N. and Royal, A.C.D. (2008). Experimental investigation of

the effects of soil properties on leakage: final report. University of Birmingham.

Royal, A.C.D., Rogers, C.D.F., Chapman, D.N., Prescott, A. and Algaard, E. (2008). Experimental investigation of the effects of soil properties on leakage. To be submit.

Satofuka, N. and Nishioka, T. (1999). Parallelisation of lattice Boltzmann method for incompressible flow computations. Computational Mechanics, **23**(2), 164-171.

Schepke, C., Maillard, N. and Navaux, P.O.A. (2009). Parallel Lattice Boltzmann Method with Blocked Partitioning. International Journal of Parallel Programming, **37**(6), 593-611.

Schroder, T. (1995). Discrete particle and continuum modelling of particulate solids. Technical Report SS-95-02.

Shafipour, R. and Soroush, A. (2008). Fluid coupled-DEM modelling of undrained behavior of granular media. Computers and Geotechnics, **35**(5), 673-685.

Shigeto, Y. and Sakai, M. (2011). Parallel computing of discrete element method on multi-core processors. Particuology, **9**(4), 398-405.

Söderlund M., Bots, P.J., Eriksson, P., Nilsson, P. and Hartlen, J. (2007). Evaluating the

domino-effect of failure of critical constructions due to damage of underground water pipelines. Loss Prevention Bulletin, **195**, 22-27.

Succi, S. (2001). The Lattice Boltzmann Equation for Fluid Dynamics and Beyond (Numerical Mathematics and Scientific Computation). Oxford University Press.

Supraksorn, T. (2009). Investigating the effects of soil type on leakage from water pipes. MSc dissertation, University of Birmingham.

Steindorff, K. (2008). Sinking into a hole new level downtown sinkhole costs & concerns. The undergrounder: Ontario sewer and water main construction association, 19-20.

Thornton, C. and Randall, C.W. (1988). Applications of theoretical contact mechanics to solid particle system simulation. Micromechanics of granular materials, Satake and Jenkins (eds), Elsevier Science Publishers, Amsterdam, Netherlands, 133-142.

Thornton C. and Yin K.K. (1991). Impact of elastic spheres with and without adhesion. Powder Technology, **65**, 153-166.

Tsuji, Y., Kawaguchi, T., and Tanaka, T. (1993). Discrete particle simulation of two-dimensional fluidised bed. Powder Technology, **77**(1), 79–87.

Wang, J., Zhang, X., Bengough, A.G. and Crawford, J.W. (2005). Domain-decomposition method for parallel lattice Boltzmann simulation of incompressible flow in porous media. Physical Review E, **72**(1), 016706.

Washington, D.W. and Meegoda, J.N. (2003). Micro-mechanical simulation of geotechnical problems using massively parallel computers. International Journal for Numerical and Analytical Methods in Geomechanics, **27**(14), 1227-1234.

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

Xu, H., Luan, H.B., He, Y.L. and Tao, W.Q. (2012). A lifting relation from macroscopic variables to mesoscopic variables in lattice Boltzmann method: derivation, numerical assessments and coupling computations validation. Computers & Fluids, **54**, 92-104.

Yang F. (2009). A numerical examination of the fluidised behaviour of Geldart's group A type particle beds. PhD thesis, University of Birmingham.

Zhang, J., Fan, L.S., Zhu, C., Pfeffer, R. and Qi, D. (1999). Dynamic behavior of

collision of elastic spheres in viscous fluids. Powder Technology, **106**(1–2), 98–109.

Zhu, H.P., Zhou, Z.Y., Yang, R.Y., and Yu, A.B. (2007). Discrete particle simulation of particulate systems: Theoretical developments. Chemical Engineering Science, **62**(13), 3378–3396.

Zou, Q. and He, X. (1997). On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. Physics of Fluids, **9**, 1591–1598.

APPENDIXES

Appendix 1: Subroutine list of *FPS-BHAM*

Subroutine initialization: calling various initialization subroutine

contains

Subroutine system_initialization: system initialization

includes:

Subroutine input_control_data: reading control data from 'control_data.txt'

Subroutine mp_initialization: MPI initialization for parallel computing

Subroutine DEM_initialization: DEM initialization

includes:

Subroutine input_DEM_data: reading DEM data from 'ball_coordinates.txt', 'wall_coordinates.txt','ball_properties.txt' and 'wall_properties.txt'

Subroutine fluid_initialization: fluid initialization

includes:

Subroutine set_region_fluid_type: set fluid type according to user requirement

Subroutine input_fluid_data: reading fluid data from 'fluid_data.txt'

Subroutine sLBM_initialization

Subroutine sDFF_initialization

Subroutine lCFD_initialization

Subroutine lDFF_initialization

Subroutines related to sCFD, lLBM is to be developed

Subroutine region_map_particle: mapping particles into sub-domains and form contact detection list

contains

Subroutine ball_transport: transporting related ball information between different processors

Subroutine region_map: mapping particles into related sub-domain

Subroutine mp_contact_transfer: transfering related information in contact array between different processsors

Subroutine fluid_calculation: calculating fluid and its related hydrodynamic force

contains

Subroutine fluid_calculation_initialization: setting fluid force applied to particles to zero

Subroutine mp_internal_boundary: applying internal boundary conditions between different sub-domains

includes:

Subroutine mp_lDFF_send

Subroutine mp_lDFF_transfer

Subroutine mp_lDFF_receive

Subroutine mp_sLBM_send

Subroutine mp_sLBM_transfer

Subroutine mp_sLBM_receive

Subroutines related to sCFD, sDFF, lLBM, lCFD is to be developed

Subroutine sLBM_system_boundary: applying system boundary conditions to sLBM sub-domains

Subroutine sDFF_system_boundary: applying system boundary conditions to sDFF sub-domains

Subroutine lDFF_system_boundary: applying system boundary conditions to lDFF sub-domains

Subroutine sLBM_region_calculation: performing computations in sLBM sub-domains

includes:

Subroutine sLBM_map_particle: mapping particles into sLBM framework

has:

Subroutine slbm_imb: immersed moving boundary

Subroutine sLBM_map_wall: mapping walls into sLBM framework

Subroutine sLBM_calculation: sLBM calculations with extrapolation scheme

Subroutine sLBM_calculation2: sLBM calculations with non-equilibrium bounce-back scheme

Subroutine sDFF_region_calculation: performing computations in sDFF sub-domains

includes:

Subroutine sdff_calculation: sDFF calculations

Subroutine lDFF_region_calculation: performing computations in lDFF sub-domains

includes:

Subroutine lDFF_map_particle: mapping particles into lDFF framework

Subroutine lDFF_calculation: lDFF calculations

Subroutine lDFF_hydroforce: computation of fluid force on the lDFF sub-domain

Subroutines related to sCFD, lLBM, lCFD is to be developed

Subroutine mp_transfer_hydro: modifying fluid force applied to the particles belonging to more than one processors

Subroutine DEM_calculation: DEM calculations

contains

Subroutine contact_calculation: detecting contacts and performing contact calculations

includes:

Subroutine ball_ball_contact_calculation

Subroutine wall_ball_contact_calculation

Both subroutines will use subroutine contact_force_calculation (transferred from the *TRUBAL* code)

Subroutine dem_update: updating particles and walls information

Subroutine postprocess: postprocessing

Appendix 2: Main calculation cycle in *FPS-BHAM*

```
CALL INITIALIZATION
CALL REGION_MAP_PARTICLE
DO CYCLENUMB= CYCLE_BEGIN, CYCLE_END
    CALL FLUID_CALCULATION
    DO I= 1, DEM_SUBCYCLE
    CALL DEM_CALCULATION
    CALL REGION_MAP_PARTICLE
    END DO
END DO
CALL POSTPROCESS
```

Appendix 3: Input files in *FPS-BHAM*

control_data.txt

Parameter	Example
cycle number when simulation begin	100001
cycle number when simulation end	105000
region_numb: number of sub-domains	4 4 1
scell_numb: number of small-scale cells in each sub-domain	150 150
lcell_numb: number of large-scale cells in each sub-domain	10 10
nballs: number of balls	10047
nwalls: number of walls	10
nballproperties: number of ball properties	12
nwallproperties: number of wall properties	1
dem_dt_frac: ratio between DEM time step to the critical DEM time step	0.5
global damping factor; damping factors for ball-ball contact and ball-wall contact	0.01 0.016 0.032
time_step: timestep of each cycle (unit: s)	1e-4
domain_size: size of domain (unit: m)	0.6 0.6 1
bodyf: body force/mass (unit: m^2/s)	0.0 -4.25 0.0

fluid_data.txt

Parameter	Example
fluid_density: density of fluid (unit: kg/m^3)	1000
unit_weight: unit weight of fluid (unit: N/m^3)	9.81e3
dynamic_viscosity (unit: kg/m.s)	1e-3
kinematic_viscosity (unit: m²/s)	1e-6
Permeability (unit: m²)	2e-4
initial_rho: initial_density (unit: kg/m^3)	1000
initial_veloc: initial_velocity (unit: m/s)	0.0 0.0
left_boundary_rho:left_boundary_density (unit: kg/m^3)	1000.0
left_boundary_veloc: left_boundary_velocity (unit: m/s)	0.0 0.0
right_boundary_rho: right_boundary_density (unit: kg/m^3)	1000.000
right_boundary_veloc: right_boundary_velocity (unit: m/s)	0.0 0.0
front_boundary_rho: front_boundary_density (unit: kg/m^3)	1000.0
front_boundary_veloc: front_boundary_velocity (unit: m/s)	0.0 0.0
back_boundary_rho: back_boundary_density (unit: kg/m^3)	1000.000
back_boundary_veloc: back_boundary_velocity (unit: m/s)	0.0 0.0
system_left_boundary: system_left_boundary_type *system_right_boundary: system_right_boundary_type* *system_front_boundary: system_front_boundary_type* *system_back_boundary: system_back_boundary_type* *4: velocity boundary 5: pressure boundary* *6: no-gradient outlet 9: periodic*	5 5 6 6

ball_coordinates.txt

Parameter	*x-coordinate (unit: m)*	*y-coordinate (unit: m)*	*type of particle*
Example	1.620E-003	3.366E-002	11

ball_properties.txt

Parameter	Example
radii (unit: m) & density (unit:kg/m^3)	1.625e-3 2765
cohes: surface energy (unit:J/m^2) & frict: friction coefficient	0.0 0.3
Ymd: Young's modolus (unit:Pa), yie: yield strength (unit:Pa) & poiss: Possion ratio	6.895e7 1.9306e9 0.3

wall_coordinates.txt

Parameter	Example
point 1: x-coordinate (unit: m)	0
point 1: y-coordinate (unit: m)	0.032
point 1: x-coordinate (unit: m)	0.298
point 1: y-coordinate (unit: m)	0.032
type of wall	1
function of wall: 1: use for both DEM and fluid 2: only for fluid; 3: only for DEM	1

wall_properties.txt

Parameter	Example
cohes: surface energy (unit:J/m^2) *& frict: friction coefficient*	0.0 0.3
Ymd: Young's modolus (unit:Pa), *yie: yield strength (unit:Pa)* *& poiss: Possion ratio*	6.895e7 1.9306e9 0.3