Weijie Zhang

2015-09-02

http://doi.org/10.20602/00003224
MODELING OF WATER-AIR-SOIL THREE-PHASE MATERIAL USING SMOOTHED PARTICLE HYDRODYNAMICS METHOD AND ITS APPLICATION TO SEEPAGE FAILURE DUE TO HEAVY RAINFALL

SPH 法を用いた土-水-空気の三相地盤のモデル化及び
集中豪雨による浸透破壊への適用

Submitted by
Weijie ZHANG

Supervised by
Professor Kenichi MAEDA

Department of Civil Engineering
Nagoya Institute of Technology, Japan

June 2015
Abstract

Extreme weather events caused by the climate change have triggered many natural disasters worldwide. Among these disasters, rainstorms or heavy rainfalls including the localized torrential downpours and the large-scale typhoon place the greatest threats to the people's lives and properties in the urban area for the triggering of river dike ruptures, floods and flow slides. There are some important factors that should be considered in the analysis of seepage failures due to heavy rainfall, such as the water-air-soil coupling, the unsaturated behavior, the large deformation, the surface infiltration, and so on.

The Smoothed Particle Hydrodynamics (SPH) method, has a unique advantage on the simulation of free surfaces, deformation boundaries and large deformations, therefore it is suitable for the analysis of slope and dike failures under heavy rainfall. However, to date, few studies regarding SPH simulations of seepage failures under heavy rainfall that consider the water-soil-air coupling, large deformation, surface infiltration, and suction reduction have been reported in the literature. Therefore, this work conducted the study on the modeling of water-air-soil three-phase material using the SPH method, the verifying and validating of the proposed SPH method, and the application of it to seepage failure model tests.

In the proposed water-air-soil three-phase SPH model, fluids and solid are simulated on different layers. The three-phase field theory was introduced and related quantities are derived for each phase. Then, governing equations including the continuity equation and the momentum equation are transformed to SPH formulas for the water phase, air phase, and soil phase, respectively. The equation of state is used to calculate the dynamic pressure of the water phase and the air phase. After that, we introduce a novel unsaturated constitutive model to describe the behavior of the soil phase. In this constitutive model, the degree of saturation is selected as one state variable. Based on the concept of sub-loading cam-clay model and skeleton stress, the yield surface can be obtained. Then, combing with the associate flow rule and giving evolutions of the state variable of overconsolidation and the state variable of saturation, the incremental
relationship between stress tensor and strain tensor can be achieved. Besides, the proposed SPH model adopts frictional forces, with due consideration of the effects of the porosity and the coefficient of permeability, as interaction forces of different phases.

The B-spline function is selected as the smoothing kernel function to obtain high accuracy and efficiency. Based on the Linked-list searching algorithm, a highly effective searching algorithm, which can be easily optimized by parallel computing, has been proposed and applied in the SPH programs. After that, the critical section in OpenMP API and the atomicAdd in CUDA solved the incoherent issue of the parallel optimization. Using the C++ programming language, highly parallelized SPH programs are written, called the ZM-OP (OpenMP) SPH program and the ZM-CD (CUDA) SPH program, respectively.

From simulations of a dam break, a triaxial compression test and a water-air coupled case, SPH simulations of the water phase, air phase, soil phase, and furthermore the effect of the suction reduction are verified and validated. The coupling between the water phase and the soil phase was checked through the simulation of a falling-head permeability test and a flowing test of liquefied soil. After that, the proposed SPH method is proved to be a highly effective numerical method by simulating a three-dimension dam break with different number of CPU threads and GPUs.

The application of the proposed SPH method is extended to three seepage failure tests, which are dike failures due to the water level up with two cases: the homogeneous dike failure and the heterogeneous dike failure, dike failures due to heavy rainfall infiltration, and slope failure due to heavy rainfall infiltration with two categories: two-phase SPH simulation and three-phase simulation. The SPH simulations reproduced the surface infiltration of shallow slope failures, the deformation process, the air blow, the suction reduction, and the Rayleigh–Taylor instability, corresponding with model tests.

The proposed SPH method provided new insights and is a novel numerical tool for the analysis of seepage failures due to heavy rainfall.
Acknowledgement

At first, I would like to express my sincere gratitude to Professor Kenichi Maeda at Nagoya Institute of Technology for his continual guidance, encouragement given through my study in master and doctor’s courses, and his critical reading of this thesis. Prof. Maeda frequently cares about my research work and daily life. Professor’s rigorous spirits and creative sights will continue to be beneficial to my work in the future.

I also wish to express my gratitude to Professor Feng Zhang at Nagoya Institute of Technology for his kind advice and valuable suggestions during the course of this study. I would also like to thank his wife, Mrs. Ying Mao. I really appreciate her for the delicious foods and life guidance.

After that, I would like to express my gratitude to Professor Yu Huang at Tongji University. Prof. Yu Huang was my supervisor of master course in Tongji University, and also gave me a lot of suggestions on life and research in my doctor’s course.

Besides, I would like to express my thanks to Prof. Teruo Nakai, Associate Professor Md. Shahin Hossain, Assistant Professor Yukihiro Morikawa at Nagoya Institute of Technology, and Mr. Tomonori Sato at Nagoya Institute of Technology, for their valuable suggestions and help. Meanwhile, I would also give my deep thanks to Assistant Professor Tatsuya Matsuda at Toyohashi University of Technology. He was my tutor in my first year of studying at NITech, and help me adjust to the life and study in Japan. I also wish to give thanks to Associate Professor Shuji Moriguchi at Tohoku University and Assistant Professor Hideto Nonoyama at National Defense Academy of Japan for their help on my study. In addition, I would like to give my thanks to Mr. Akihiko Kondo for his help on my study and life.

This research cannot be finished without the collaboration and support of dike team
in Maeda Lab. Here, I would like to express my gratitude to Mr. Satoshi Shibata, Mr. Kenichiro Ide, Mr. Hiroshi Saito, Ms. Ai Wakasa, Ms. Zhaoqing Li, and Mr. Atsushi Yamaguchi for their excellent experiments. Meanwhile, Maeda Lab provided a friendly environment for studying, thus I also wish to give my thanks to Mr. Naoto Naito, Mr. Yuta Okumura, Mr. Takuya Arai, Mr. Hiroaki Sato, Mr. Yuya Tanaka, and other members that were or are in Maeda Lab.

In addition, I would like to give my thanks to Mr. Yuanfeng Bao, Ms. Xiaohua Bao, and Mr. Yonglin Xiong, for their help on the constitutive model of unsaturated soils. I would also like to give my thanks to Mr. Zili Dai at Tongji University, and Ms. Hualin Cheng at Tongji University, for their sharing on study and life. Besides, I would also thank my friends in Japan, and we have experienced a lot of good times.

I would pay my sincere gratitude to my family, especially to my wife and my parents, for their unconditional love, support and encouragement. I am always feeling lucky and happy once I think of my family.

Finally, I would like to acknowledge Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT) for its financial support throughout my five-year study.

Weijie ZHANG
Nagoya Institute of Technology
June 2015
Table of contents

Abstract ......................................................................................................................... i
Acknowledgement ........................................................................................................ i
Table of contents ......................................................................................................... 1

Chapter 1 Introduction ................................................................................................. 3
  1.1 Research background ............................................................................................ 3
  1.2 Current research .................................................................................................. 5
  1.3 Research purposes .............................................................................................. 7
  1.4 Research scheme and innovations ...................................................................... 8
  1.5 Structure of this thesis ....................................................................................... 9
  References ................................................................................................................ 10

Chapter 2 Principles of the water-air-soil three-phase coupled SPH method .......... 17
  2.1 General ............................................................................................................... 17
  2.2 Basic concepts of the SPH method ..................................................................... 17
  2.3 Assumptions and the three-phase mixture theory .............................................. 20
  2.4 Controlling equations ....................................................................................... 22
  2.5 Equations of state for fluids ............................................................................. 27
  2.6 Unsaturated constitutive model for the soil phase ........................................... 28
  2.7 The strain rate tensor and Jaumann rate tensor of the Cauchy stress ............ 39
  2.8 Interaction forces between different phases ...................................................... 40
  2.9 Summary ........................................................................................................... 41
  References ................................................................................................................ 41

Chapter 3 SPH programs and their parallel optimization ........................................ 45
  3.1 General ............................................................................................................... 45
  3.2 Smoothing kernel function ............................................................................... 45
  3.3 Particle searching algorithm.............................................................................. 47
  3.4 Boundary treatment .......................................................................................... 52
  3.5 Time integration method .................................................................................. 54
  3.6 Parallel computing ............................................................................................ 54
  3.7 The ZM-OP SPH program and ZM-CD SPH program .................................... 59
  3.8 Summary ........................................................................................................... 70
References .............................................................. 70

Chapter 4 Verification and validation of the SPH method .............. 73
  4.1 General ............................................................ 73
  4.2 Dam break .......................................................... 73
  4.3 Tri axial compression test ....................................... 77
  4.4 Air bubble rising and burst from water ......................... 80
  4.5 Falling head permeability test ................................ 82
  4.6 Flowing process of liquefied soils .............................. 84
  4.7 Efficiency of ZM-OP SPH program ............................ 88
  4.8 Efficiency of ZM-CD SPH program ............................ 91
  4.9 Summary .......................................................... 93

References .................................................................. 93

Chapter 5 Application of the proposed SPH method to seepage failures
due to heavy rainfall .................................................. 95
  5.1 General ............................................................... 95
  5.2 The water-soil-coupled SPH simulations of seepage failure due to the water level up .............................................. 95
  5.3 The water-soil-coupled SPH simulation of dike failures due to the rainfall infiltration .................................................. 103
  5.4 The water-soil-coupled SPH simulation of slope failures due to the rainfall infiltration .................................................. 108
  5.5 The water-air-soil-coupled SPH simulation of slope failures due to the rainfall infiltration .................................................. 113
  5.6 Summary ............................................................ 116

References .................................................................. 117

Chapter 6 Conclusions and remarks ...................................... 119
  6.1 Conclusions .......................................................... 119
  6.2 Remarks on the future study ...................................... 120

Appendix ........................................................................
  A-1 Source of ZM-OP SPH program and ZM-CD SPH program .......... i
  A-2 Compiling of the ZM-OP SPH program and the ZM-CD SPH program ........ i
  A-3 Input file ............................................................ ii
  A-4 Running the SPH programs ...................................... vi
  A-5 Output file .......................................................... vi
Chapter 1

Introduction

1.1 Research background

Extreme weather events caused by the climate change have induced many natural disasters worldwide. Among these disasters, rainstorms or heavy rainfalls including the localized torrential downpours and the large-scale typhoon place the greatest threats to the people's lives and properties in the urban area for the triggering of river dike ruptures, floods and flow slides.

The Shin-river dike in the Nagoya City broke at about 3:30 on September 12th, 2000, due to the Tokai Downpour that started from the preceding day, and areas along the river were seriously damaged (The Japanese Geotechnical Society Chubu 2001). Eyewitness testimony describing the breakage and washout of the Shin-river River dike mentioned that “a crack of approximately one meter width obliquely formed across the dike and white bubbly water escaped. After that, the crack slowly expanded for about three hours.” (Chunichi Shinbun 2000).

Differently from the air burst in the Shin-river dike, the air blow shown in Fig. 1.1, which was less intense than the air burst, was witnessed in the dike of the Syonai-river in Nagoya City on September 2011 triggered by the downpour of typhoon No.15. However, failure did not appear in the Syonai-river. Thus, it is needed to clarify the air effect in the seepage process of rainfall and water to assess the stability of dike.

Besides the air effect, piping also plays an important role in the fail process of river dikes. From 13th July to 14th July 2012, a rapid downpour that reached a maximum intensity of 108 mm/hr. induced a rise of the water level in the Yabe-river of Fukuoka Prefecture, Japan. Because of the seepage in the sand layer underlying the dike, it failed
on 14\textsuperscript{th} July (Investigate Committee of Yabe-river Dike 2013). This disaster brought an area of 2,759 ha flooded, causing the death of 30 people and the damage of 1,870 houses.

In addition, the rainfall induced flow slide was also a serious geological disaster. A strong storm attacked the Zhouqu County in Gansu Province, China, and a slope around the urban area failed on 7\textsuperscript{th} August 2010. The flow slide blocked a river and formed two barrier lakes. This disaster caused 1,467 people dead and 298 people lost, and serious damages on buildings and roads as well (Yu et al. 2010).

From above, the seepage failures triggered by heavy rainfall, such as dike failures, flow slides, and so on, should be studied to investigate the failure mechanism, to propose
related countermeasures, and to guide the reconstruction.

1.2 Current research

For the great damage and complexity of seepage failures, it is a hot and challenging topic in the geotechnical engineering. Many researchers tried to interpret the mechanism of seepage failures from different viewpoints, for example, the influencing factors, the constitutive models, the shear strength reduction, the large deformation with free surface, the shallow slope failure, and so on.

For influencing factors, the antecedent rainfall and 24-hour rainfall amount have important impacts on the stability of slope and dike according to the research of Rahardjo et al. (2001). Besides, the formation of the perched water table, the infiltration, and the erosion by the kinetic energy of overland flow are also three important factors in the fail process of slopes due to rainfall. Then, the characteristics of the rainfall (duration, intensity and pattern), the saturated hydraulic conductivity of soil, the slope geometry, the initial conditions and the boundary conditions have all been identified as influential factors as well (Ali et al. 2014; Camici et al. 2015). In addition, Lam et al. (1987), Kodaka & Asaoka (1994), Zhang et al. (2009), Maeda & Sakai (2010), Mori et al. (2011) and Xiong et al. (2014) have presented the air behavior in the seepage process using the three-phase-coupled methods. Even though, from real cases of seepage failures, the effect of the air phase is still unclear, thus, the coupling of water, air, and soil is essential for the simulation of seepage failures.

The accuracy of stability analysis depends on the accuracy of model used. Early seepage model usually adopted the saturated soil model without considering the change of matric suction. It is well known that the seepage can change the geomaterial from unsaturated state to saturated state, and then reduce the matric suction and shear strength (Bui et al. 2008a), thus more and more scholars turned to the unsaturated model to get more precise seepage failure mechanism (Lam et al. 1987; Rahardjo et al. 2001; Chen & Zhang 2006; Bui et al. 2008a; Pastor et al. 2009; Mori et al. 2011; Polemio & Lollino 2011; Regmi et al. 2013; Ali et al. 2014; Rabie 2014; Xiong et al. 2014). Along with the development of geomechanics, some new constitutive models for unsaturated soils have
appeared, and the seepage simulation tool should learn the new research on unsaturated soils.

For seepage failures, the analysis of the entire process from the rainfall impact to the final accumulation, and the coupling between the skeleton and the pore fluids, are important to get clearer mechanism (Pastor et al. 2009). Besides, the phase transition from solid to liquid at the triggering stage and from liquid to solid at the ceasing stage often appeared for the flow-like landslides by rainfall (Pastor et al. 2009). Therefore, the simulation of large deformation is a key factor to get the precise mechanism (Kodaka et al. 2001; Maeda et al. 2006; Maeda & Sakai 2010).

Previous studies have found that the slope failure due to rainfall is commonly the shallow slope failure and the shallow landslide (Collins & Znidarcic 2004; Reubens et al. 2007; Egeli & Pulat 2011; Cascini et al. 2013). The shallow slope failure and the shallow landslide induced by rainfall are mainly subject to a surface infiltration that is parallel to the slope surface and two distinct failure mechanisms: complete liquefaction of failed mass by seepage force and the suction reduction of unsaturated soil (Collins & Znidarcic 2004; Regmi et al. 2013; Ali et al. 2014). Based on this, the surface infiltration, the liquefaction effect, and the suction reduction should be considered in the modeling of seepage failures.

At present, many approaches are used to conduct the analysis of seepage failures. Among these methods, limit equilibrium methods and the finite element method are often adopted to obtain the factor of safety for the assessment (Cascini et al. 2013; Rabie 2014). Although such methods can obtain precise simulations in certain situations, there are still some difficulties that these methods cannot deal with, e.g. the large deformation problem, because excessive mesh distortions require remeshing (Cascini et al. 2013; Ali et al. 2014). To solve these problems, several mesh free methods, such as the Element Free Galerkin Method (EFGM) (Belytschko et al. 1994), the Material Point Method (MPM) (Sulsky et al. 1995), the Computational Fluid Dynamics (CFD) method (Trunk et al. 1986), and the Smoothed Particle Hydrodynamics (SPH) method (Lucy 1977), have been developed and applied in the geotechnical engineering. Among these methods, the CFD method is often used to simulate the large deformation of geomaterial and achieved some insights (Sousa & Voight 1991; Uzuoka et al. 1998; Hadush et al. 2001; Sawada et al. 2004; Moriguchi
et al. 2005).

In the category of CFD methods, the Smoothed Particle Hydrodynamics (SPH) method, has a unique advantage on the simulation of free surfaces, deformation boundaries and large deformations (Liu & Liu 2010). SPH can readily accommodate the large deformation and the entire fail process of geomaterial, therefore it is suitable for the analysis of slope and dike failures under heavy rainfall (Bui et al. 2008b). This method was originally developed in astrophysics to solve the equation of motion for galaxies (Lucy 1977) and in a short time was extended to solid mechanics (Monaghan 1988; Monaghan 1992; Chen et al. 2013). In the geotechnical engineering, applications of SPH to landslides and flowing of liquefied soils have been reported recently (McDougal and Hungr 2004; Pastor et al. 2009; Huang et al. 2011; Huang et al. 2012; Dai et al. 2014; Huang & Dai 2014; Hu et al. 2015).

At present, there have been some studies on geo disasters using the coupled SPH methods. The flowing process of municipal solid waste (MSW) landfill was simulated by the two-phase-coupled SPH method (Huang et al. 2013a). Besides, Huang et al. (2013b) conducted an analysis of large-deformation flows of liquefied soils using the water-soil-coupled SPH method. Nonoyama et al. (2012) and Bui & Fukagawa (2013) employed the single-phase and two-phase SPH method, respectively, to analyze the stability of the slope. In addition, the tsunami disaster was also simulated by the SPH method with three-phase coupling (Imase et al. 2013). For the seepage failure, Maeda and Sakai (2010) introduced air dynamics into the SPH framework to investigate air effects. Zhang & Maeda (2014) and Zhang & Maeda (2015) have showed preliminary results of a slope failure test that considered the effect of air and SPH simulations using the two-phase-coupled SPH model. However, to date, few studies regarding SPH simulations of seepage failures due to heavy rainfall that consider the water-soil-air coupling, the large deformation, the surface infiltration, and the suction reduction have been presented in the literature.

### 1.3 Research purposes

This research aims at the three-phase coupling, surface infiltration, suction reduction, and large deformation, which occur in the process of seepage failures due to heavy rainfall,
and employs the SPH method to achieve the following purposes,

(1) Building the SPH model of water-air-soil three-phase material considering the three-phase coupling.

(2) Introducing the constitutive model for unsaturated soils into the SPH framework.

(3) Implementation of the highly effective SPH programs using the parallel computing.

(4) Verification and validation of the proposed SPH method.

(5) Extending the application of the proposed SPH method to the real cases of seepage failures.

### 1.4 Research scheme and innovations

<table>
<thead>
<tr>
<th>STUDY OF BASIC PRINCIPLES</th>
<th>THE SPH PROGRAM</th>
<th>VERIFICATION AND VALIDATION</th>
<th>THE APPLICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>The concepts of SPH method</td>
<td>Smoothing kernel function</td>
<td>Dam break</td>
<td>Seepage failures due to piping</td>
</tr>
<tr>
<td>Mixture theory</td>
<td>Particle searching algorithm</td>
<td>Tri axial shear test</td>
<td>● Homogeneous</td>
</tr>
<tr>
<td>Controlling equations</td>
<td>Boundary treatment</td>
<td>Air-bubble rising and burst</td>
<td>● Heterogeneous</td>
</tr>
<tr>
<td>Modeling of fluid phase</td>
<td>Integration method</td>
<td>Falling-head permeability test</td>
<td>Dike failure due to heavy rainfall</td>
</tr>
<tr>
<td>Modeling of unsaturated soils</td>
<td>Time step</td>
<td>Flow analysis of liquefied soils</td>
<td>● Two phases</td>
</tr>
<tr>
<td>Interaction forces</td>
<td>Parallel optimization</td>
<td>The efficiency (3D dam break)</td>
<td>● Three phases</td>
</tr>
</tbody>
</table>

Fig. 1.3 Flow chart of this research.

Fig. 1.3 shows the research scheme. We can see that this research has four parts as follows,

(1) Studying on the basic principles of the water-air-soil three-phase coupled SPH model.

(2) Studying of the particle searching algorithm, the kernel function or smoothing function, the boundary treatment, the numerical integration method, and the
parallel optimization (OpenMP and CUDA).
(3) Studying on the verification and validation of the proposed SPH method.
(4) Studying on the application of the proposed SPH method to seepage failure test due to heavy rainfall.

The innovations of this research can be summarized as follows,
(1) The newly developed Smoothed Particle Hydrodynamics method is proposed to simulate the seepage failure.
(2) The coupling of water, air, and soil is considered in the framework of the SPH method.
(3) A novel constitutive model for unsaturated soils is firstly used in the SPH method describing the behavior of the solid phase and the suction reduction in seepage failures.
(4) An improved particle searching algorithm is proposed based on the Linked-list algorithm.
(5) Highly parallelized SPH programs, the ZM-OP (OpenMP) SPH and the ZM-CD (CUDA) SPH, are written with the OpenMP API and CUDA language, respectively. Both SPH programs achieve a high performance.
(6) The application of the proposed SPH method is extended to seepage failure tests.

1.5 Structure of this thesis

The structure of the current thesis is as follows,

Chapter 2: Principles of the water-air-soil three-phase coupled SPH method

The basic concepts of the SPH method are presented firstly. Then, some assumptions and the three-phase mixture theory are described. After that, controlling equations for different phases, as well as calculations of the fluid pressure and the soil stress, are derived by the particle approximation. At last, interaction forces between different phases are presented.

Chapter 3: The SPH program and its parallel optimization

The smoothing kernel function and particle searching algorithm are summarized,
Chapter 1 Introduction

compared, and selected to obtain the high efficiency and accuracy simultaneously. Then, the boundary treatment method and the integration method are described. At last, the parallel optimization of the ZM-OP (OpenMP) SPH program and the ZM-CD (CUDA) SPH program are introduced.

Chapter 4: Verification and validation of the SPH method

In this chapter, we show the SPH simulations of a dam break, a tri axial compression test, the rising and burst of air bubble, a falling-head permeability test, a flowing test of liquefied soils, and a 3D dam break, to verify and validate the application and efficiency of the proposed SPH method.

Chapter 5: Application of the proposed SPH method to seepage failure tests due to heavy rainfall

Three seepage failure tests are simulated by the proposed SPH method. At first, the seepage failure due to piping is simulated with two cases: the homogeneous dike and the heterogeneous dike. Then, a dike failure test due to heavy rainfall is simulated with the two-phase coupled SPH method. At last, a slope failure by heavy rainfall is simulated with two cases: the two-phase coupled simulation and the three-phase coupled simulation. Some mechanisms of seepage failures are presented in this chapter.

Chapter 6: Conclusions and remarks

This chapter presents the conclusions of this thesis and several remarks on the future study.

Appendix

The source, compiling, and usage of the SPH programs are introduced in this part.

References

heritage. Journal of Disaster Mitigation of Cultural Heritage and Historic Cities, 2, 111-118.


during high intensity and short duration rainfall. Scientia Iranica, 18(6), 1179–1187.


Chapter 1 Introduction


Chapter 1 Introduction
Chapter 2

Principles of the water-air-soil three-phase coupled SPH method

2.1 General

This chapter is mainly about the basic concepts of water-air-soil three-phase coupled SPH model. The basic concepts of the SPH method are presented at first. Then, some assumptions and the three-phase mixture theory are described. After that, controlling equations for different phases are derived by the particle approximation. For the fluid phase (water and air), the equation of state is used for the calculation of dynamic pressure. Meanwhile, the novel constitutive model for unsaturated soils is introduced to describe the mechanical behavior of soil phase. At last, the Jaumann stress rate and interaction forces between different phases are used to obtain the objective stress and the coupling of three phases.

2.2 Basic concepts of the SPH method

SPH is a truly mesh-free particle method based on a pure Lagrange description, which was first developed to solve astrophysical problems in the three-dimensional open space, particularly polytopes (Lucy 1977; Gingold & Monaghan 1977). Later, Monaghan & Lattanzio (1985), Liu & Liu (2003) and Liu & Liu (2010) summarized basic concepts of discretization for governing equations, including the continuity equation, the momentum equation, and the energy equation.

The core of this method is fully implied in the three words Smoothed Particle
Hydrodynamics (Huang et al. 2014). “Smoothed” represents the smoothed approximation nature of using weighted averages of neighboring particles for stability. “Particle” indicates that the method is based on mesh-free particle method and the computing domain is treated using a discrete particle instead of continuous entities. “Hydrodynamics” points to the fact that the SPH method was first applied to hydrodynamics problems and employs the Navier-Stokes equations in fluid dynamics as controlling equations.

The basic concept of SPH is that a continuous entity is represented by a set of arbitrarily distributed particles. The moving particles possess material properties and various physical quantitates. By providing accurate and stable numerical solutions for hydrodynamic equations, and tracking movements of each particle, this method can describe the mechanical behavior of an entire system. Therefore, the key point of SPH is how to solve the Partial Differential Equations (PDEs) using a series of arbitrarily distributed particles carrying field variables, such as the mass, density, energy, and stress tensors. In an actual situation, it is usually difficult to obtain an analytical solution of these PDEs, which gives rise to the need for numerical methods of them. The first step is to discretize the problem domain of the PDEs. Then, there is a need to approximate the variable function and its derivative for the arbitrary particles. Finally, approximate functions are applied to the PDEs to obtain a series of discretization Ordinary Differential Equations (ODEs), which are only related to time.

The basic concepts of the SPH method can be summarized as follows (Huang et al. 2014):

1. The problem domain of SPH model is replaced by a series of arbitrarily distributed particles. There is no connectivity between these particles, which reflects the mesh-free nature of this method. The major concern of this method is how to ensure the stability of numerical solutions, especially in applying the arbitrarily distributed particles to solve problems with derivative boundary conditions.

2. One of the most important steps is to represent a function in continuous form using an interpolation function. This step is usually called the kernel approximation. The integral has a smoothing effect, similar to weak form equations. In reality, the kernel approximation stabilizes numerical calculation of
the SPH.

3. Another important step is that the value of a function at computing particle \( i \) is approximated using averages of function values at all neighboring particles within the horizon of particle \( i \). This step is termed the particle approximation. The role of this approximation is to generate banded or sparse discretized system matrices, which are extremely important for calculation efficiency.

4. Using an explicit integration algorithm to solve differential equations can achieve fast time stepping. The time history of all field variables for all the particles can also be obtained. An appropriate method to determine the time step must be selected in the SPH method.

The smooth approximation of a field function is shown as follow,

\[
\langle f(x) \rangle = \int_D f(x') W(x - x', h) dx' = \int_D f(x') W(r, h) dx' \quad (2.1)
\]

where \( f(x) \) is the field function, \( x \) is the position vector of a computing particle, \( x' \) is the position vector of the supporting particle, \( h \) is the smoothing length and \( W \) is the weight that is also called the smoothing kernel function.

The smooth approximation of the spatial derivative is obtained replacing the \( f(x) \) in Eq. (2.1) with \( \nabla f(x) \).

\[
\langle \nabla f(x) \rangle = \int_D \nabla f(x') W(x - x', h) dx' = \int_D \nabla f(x') W(r, h) dx' \quad (2.2)
\]

Using the integration by parts, we can obtain the following equation,

\[
\langle \nabla f(x) \rangle = \int_S f(x') W(x - x', h) ndS - \int_D f(x') \nabla W(x - x', h) dx' \quad (2.3)
\]

where \( S \) is the boundary surface of the integral volume and \( n \) is the unit vector normal to the surface \( S \). Because of the compact support condition of \( W \), the surface integral is zero in the influence domain \( D \). Thus, Eq. (2.3) can be rewritten as,

\[
\langle \nabla f(x) \rangle = -\int_D f(x') \nabla W(x - x', h) dx' \quad (2.4)
\]

The particle approximation shown in Fig. 2.1 can be used to obtain the discretized forms of Eq. (2.1) and Eq. (2.4).
Fig. 2.1 The particle approximation in the SPH method.

The particle approximation of a field function is presented in the following equation,

$$\langle f(x) \rangle = \sum_{j=1}^{N} m_j \frac{f_j(x')}{\rho_j} W(x - x', h) \quad (2.5)$$

where, $N$ is the number of supporting particles, $f_j(x')$ is the function value at the supporting particle $j$, $m_j$ is the mass of supporting particle $j$, and $\rho_j$ is the density of the supporting particle $j$.

The particle approximation of Eq. (2.4) is as follow,

$$\langle \nabla f(x) \rangle = -\sum_{j=1}^{N} m_j \frac{f_j(x')}{\rho_j} \nabla W(x - x', h) \quad (2.6)$$

The partial differential form of the field function can be written as:

$$\frac{\partial \langle f(x) \rangle}{\partial x_i} = \sum_{j=1}^{N} \frac{m_j}{\rho_j} f_j(x_j) \frac{\partial W(x - x_j, h)}{\partial x_i} \quad (2.7)$$

2.3 Assumptions and the three-phase mixture theory

Before the derivation of the three-phase mixture theory, following assumptions should be listed out,

1. The strain is small;
2. The distribution change of porosity in space and time is very small comparing to other variables;
3. The soil and water particles are incompressible.
In the proposed SPH method, the fluids and the solid are simulated on different layers. The fluid layer includes the water phase and the air phase, while the solid layer is for the soil phase. The particle in the SPH method is a set of real material elements (water, air, and soil) with meaning of mathematic. From this point of view, the particle on one layer can overlap another particle on the other layer in the SPH method.

![Fluid layer and solid layer in the proposed SPH model.](image)

Uzuoka et al (2007, 2008 and 2010) has proposed a soil-water-air three-phase field theory, which works effectively for slope stability problems in unsaturated soils. Such type of soil-water-air three-phase field theory can also be found in the works of Li et al (2004), Borja (2005), Mori et al. (2011), Xiong et al. (2014) and Zhang et al. (2014). Because of its good performance, we adopted this field theory in the framework of SPH method to model the water-air-soil three-phase material.

According to the field theory, the density of the water phase \( \rho^w \), the air phase \( \rho^a \), the soil phase \( \rho^s \), and the mixture \( \rho \) can be written as follows,

\[
\rho^w = nS_s \rho^w
\]

\[
\rho^a = n_s (1 - S_s) \rho^a
\]

\[
\rho^s = (1 - n_s) \rho^s
\]

\[
\rho = \rho^w + \rho^a + \rho^s = (1 - n_s) \rho^s + n_s \left[ S_s \rho^w + (1 - S_s) \rho^a \right]
\]
2.3.1 Navier-Stokes equations

In the SPH method, Navier–Stokes (N–S) equations are often employed as the controlling equations (Liu & Liu 2003). It includes the equation of continuity derived from the conversation of mass, the equation of motion derived from the conversation of momentum, and the equation of energy derived from the conversation of energy. These conversations are also applicable for geomaterials (Li et al 2004; Borja 2005; Mori et al. 2011; Xiong et al. 2014; Zhang et al. 2014), thus, N–S equations can be the controlling equations in the proposed SPH method. Because simulation cases in this thesis are assumed to be isothermal, the energy equation is not considered.
\[
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho \mathbf{u}^m)}{\partial x^m} = 0 \tag{2.17}
\]

where, \( t \) is the time, \( \mathbf{u}_i \) is the velocity also indicated by \( \mathbf{v}_i \) and \( m \) indicates the direction of coordination. Therefore, we can get the change rate of density as,

\[
\frac{d \rho}{dt} = -\nabla (\rho \mathbf{v}) \tag{2.18}
\]

Using Eq. (2.6), the particle approximation of the continuity equation can be obtained as,

\[
\frac{d \bar{\rho}_i}{dt} = \sum_{j=1}^{N_i} m_j (v_i^m - v_j^m) \frac{\partial W_{ij}}{\partial x^m} \tag{2.19}
\]

where, \( i \) indicates the computing particle, \( j \) indicates the supporting particle, \( N \) is the number of supporting particles, \( W_{ij} \) is the smoothing kernel function of particle \( i \) evaluated at particle \( j \), \( x \) is the position vector, and \( \mathbf{v} \) is the velocity vector. Thus, we can get the equations of continuity for each phase in Eqs. (2.20-2.22).

\[
\frac{d \bar{\rho}_i^w}{dt} = \sum_{j=1}^{N_i} m_j (v_i^m - v_j^m) \frac{\partial W_{ij}}{\partial x^m} \tag{2.20}
\]

\[
\frac{d \bar{\rho}_i^a}{dt} = \sum_{j=1}^{N_i} m_j (v_i^m - v_j^m) \frac{\partial W_{ij}}{\partial x^m} \tag{2.21}
\]

\[
\frac{d \bar{\rho}_i^s}{dt} = \sum_{j=1}^{N_i} m_j (v_i^m - v_j^m) \frac{\partial W_{ij}}{\partial x^m} \tag{2.22}
\]

When Eqs. (2.20-2.22) are used, since there will be fewer supporting particles in the vicinity of a free interface, the density tends to be underestimated in that region. To achieve a better summarization of density, the next equation can be used, in which the smoothing functions are summed for normalization (Randles & Libersky 1996).

\[
\rho_i = \frac{\sum_{j=1}^{N_i} m_j W_{ij}}{\sum_{j=1}^{N} \left( \frac{m_j}{\rho_j} \right) W_{ij}} \tag{2.23}
\]

However, some improvements to this normalization are needed in order to calculate
the density in the vicinity of interface between phases in multi-phase conditions. In other words, in the vicinity of interface between phases with greatly differing state equations and densities, even Eq. (2.23) will result in large errors in the calculations of density. For example, considering the situation of air bubbles in water, the state equations for air and water are quite different, and water is roughly one thousand times denser than air. This causes an underestimate in the water density near bubble surfaces and an overestimate of the air density on the air side of the interface. The calculation then fails because the pressure is estimated at physically impossible high or low values due to these misestimates of density. This cannot be avoided by using either of Eq. (2.19) and Eq. (2.23).

In this study, this problem is solved by summating all particles of each given phase when using Eq. (2.23). Following equations show the present approach for a three-phase mixture combining water, air and soil represented by respective particles.

\[
\rho_{i\text{water}} = \frac{\sum_{j=1}^{N_{\text{water}}} m_j W_{ij} \rho_j}{\sum_{j=1}^{N_{\text{water}}} m_j W_{ij}} \quad (2.24)
\]

\[
\rho_{i\text{air}} = \frac{\sum_{j=1}^{N_{\text{air}}} m_j W_{ij} \rho_j}{\sum_{j=1}^{N_{\text{air}}} m_j W_{ij}} \quad (2.25)
\]

\[
\rho_{i\text{soil}} = \frac{\sum_{j=1}^{N_{\text{soil}}} m_j W_{ij} \rho_j}{\sum_{j=1}^{N_{\text{soil}}} m_j W_{ij}} \quad (2.26)
\]

The above approach allows the densities of different materials to be calculated with high precision everywhere, including the vicinity of interfaces.

2.3.3 The equation of momentum

Generally, the equation of momentum is given by,
\[ \rho \frac{dv_i}{dt} = \frac{\partial \sigma_{ij}}{\partial x_j} + F_i \]  \hspace{1cm} (2.27)

where, \( \sigma_{ij} \) is the total stress and \( F_i \) is the external force including the gravity and interaction force.

Using the particle approximation shown in Eq. (2.7), the SPH formulation of the equation of momentum can be derived as,

\[ \frac{dv_{im}}{dt} = \sum_{j=1}^{N} m_j \left( \frac{\sigma_{ij}^{mn}}{\rho_j^2} + \frac{\sigma_{ij}^{mn}}{\rho_j^2} \right) \frac{\partial W_{ij}}{\partial x_i^n} + \frac{F_{im}}{\rho_i} \]  \hspace{1cm} (2.28)

Here, \( \rho \) is the density calculated from Eqs. (2.20-2.22), \( i \) indicates the computing particle, \( j \) indicates the supporting particle, and \( m \) and \( n \) indicates the direction of coordination.

For fluids, the equation of momentum is,

\[ \frac{dv_{fm}}{dt} = \sum_{j=1}^{N} m_j \left( \frac{\sigma_{ij}^{mn}}{\rho_j^2} + \frac{\sigma_{ij}^{mn}}{\rho_j^2} \right) \frac{\partial W_{ij}}{\partial x_i^n} + \frac{F_{im}}{\rho_i} \]  \hspace{1cm} (2.29)

For the water phase, \( F_i \) is defined as,

\[ F_{im} = b_{im} - R_{im} \]  \hspace{1cm} (2.30)

where, \( b_{im} \) is the body force and \( R_{im} \) is the interaction force between the water phase and the soil phase.

For the air phase, it is,

\[ F_{im} = b_{im} - Q_{im} \]  \hspace{1cm} (2.31)

where, \( b_{im} \) is the body force and \( Q_{im} \) is the interaction force between the air phase and the soil phase.

For fluids, the total stress tensor \( \sigma^{mn} \) can be obtain from following equations considering the effect of viscosity, which are derived from the momentum equation of Liu & Liu (2003),

\[ \sigma^{wun} = n_p S \left[ -(p_d^w + p_v^w) \delta^{mn} - \frac{2}{3} \mu \dot{e}^{ii} \delta^{mn} + 2 \mu \dot{e}^{mn} \right] \]  \hspace{1cm} (2.32)

\[ \sigma^{mun} = n_p (1 - S) \left[ -(p_d^w + p_v^w) \delta^{mn} - \frac{2}{3} \mu \dot{e}^{ii} \delta^{mn} + 2 \mu \dot{e}^{mn} \right] \]  \hspace{1cm} (2.33)
where, \( n_p \) is the porosity, \( \sigma^{nmn} \) and \( \dot{\varepsilon}^{nmn} \) is the stress tensor of the water phase and the air phase, and \( \dot{\varepsilon}^{nmn} \) is the strain rate. The pressure of fluids is obtained from the equation of state for fluids. In Eq. (2.29), particle \( i \) and particle \( j \) can be different phases, which means that if particle \( i \) is a water particle, particle \( j \) can be a water particle or air particle, and if particle \( i \) is an air particle, particle \( j \) can also be a water particle or air particle. This is the interaction between the water phase and the air phase.

For the soil phase, the equation of momentum is,

\[
\frac{dv_i^m}{dt} = \sum_{j=1}^{N} m_j \left( \frac{\sigma_i^{mn}}{\rho_i} + \frac{\sigma_j^{mn}}{\rho_j} - \delta^{mn} \Pi_{ij} + f_{ij}^{mn} \right) \frac{\partial W_{ij}}{\partial x_i^m} + \frac{F_i^m}{\rho_i} \tag{2.34}
\]

where, \( \sigma_i^{mn} \) is the total stress tensor shown in Eq. (2.15). \( F_i^m \) is defined in the following equation for the soil phase,

\[
F_i^m = b_i^m + R_i^m + Q_i^m \tag{2.35}
\]

Here, artificial viscosity \( \Pi_{ij} \) is introduced into the equation of momentum to avoid the numerical oscillation and resist penetration between particles (Monaghan 1988). Artificial viscosity can be obtained from Eq. (2.36) and Eq. (2.37), where \( m \) indicate the direction of coordination, \( c \) is the velocity of sound, \( \rho \) is the density, \( h \) is the smoothing length, parameter \( a \) is set to 0.001 and parameter \( b \) is set to 0.0.

\[
\Pi_{ij} = \begin{cases} 
\frac{-ac_{ij}v_{ij}^m + b\phi_{ij}^2}{\rho_{ij}} & v_{ij}^m x_{ij}^m < 0 \\
0 & v_{ij}^m x_{ij}^m \geq 0 
\end{cases} \tag{2.36}
\]

\[
\phi_{ij} = \frac{h_{ij} (v_{ij} \cdot x_{ij})}{|x_{ij}^2| + 0.01h_{ij}^2} \\
c_{ij} = 0.5(c_i + c_j) \\
\overline{\rho_{ij}} = 0.5(\rho_i + \rho_j) \\
h_{ij} = 0.5(h_i + h_j) \tag{2.37}
\]

The tensile instability will appear when the SPH method simulates the material with strength, and the artificial stress \( f_{ij}^{mn} \) can be introduced to deal with this problem (Grey et al. 2001). The artificial stress can be obtained from Eqs. (2.38-2.42), where \( m \) and \( n \)
indicate the direction of coordination, $R_{ij}$ is the distance between particle $i$ and particle $j$, $\varepsilon$ is a controlling parameter of 0.3 (Grey et al. 2001), and $\Delta p$ is the initial spacing of particles.

$$\theta_i = \begin{cases} \frac{0.5 \arctan \left( \frac{2\sigma_{yy}^{\text{ij}}}{\sigma_{xx}^{\text{ij}} - \sigma_{yy}^{\text{ij}}} \right)}{\pi} & \sigma_{xx}^{\text{ij}} \neq \sigma_{yy}^{\text{ij}} \\
\frac{\pi}{4} & \sigma_{xx}^{\text{ij}} = \sigma_{yy}^{\text{ij}} \end{cases} \text{ for } t = i, j$$ (2.38)

$$\sigma_{xx}^{\text{ij}} = c^2 \sigma_{xx}^{\text{ij}} + 2cs \sigma_{yy}^{\text{ij}} + s^2 \sigma_{yy}^{\text{ij}} \quad c = \cos \theta_i$$

$$\sigma_{yy}^{\text{ij}} = s^2 \sigma_{xx}^{\text{ij}} + 2cs \sigma_{yy}^{\text{ij}} + c^2 \sigma_{yy}^{\text{ij}} \quad s = \sin \theta_i$$ (2.39)

$$R_{ii}^{xx} = \begin{cases} \frac{-\varepsilon \sigma_{xx}^{\text{ij}}}{\rho_i^2} & \sigma_{xx}^{\text{ij}} > 0 \\
0 & \text{else} \end{cases} \quad R_{ii}^{yy} = \begin{cases} \frac{-\varepsilon \sigma_{yy}^{\text{ij}}}{\rho_i^2} & \sigma_{yy}^{\text{ij}} > 0 \\
0 & \text{else} \end{cases}$$ (2.40)

$$r^{xx} = c^2 R_{ii}^{xx} + s^2 R_{ii}^{yy}$$

$$r^{yy} = s^2 R_{ii}^{xx} + c^2 R_{ii}^{yy}$$

$$r^{xy} = cs(R_{ii}^{xx} - R_{ii}^{yy})$$ (2.41)

$$f = \frac{W(R_{ij})}{W(\Delta p)}, \quad r_{\text{ij}}^{\text{mn}} = r_{i}^{\text{mn}} + r_{j}^{\text{mn}}$$ (2.42)

### 2.5 Equations of state for fluids

In the proposed SPH method, the water phase is regarded as a quasi-incompressible fluid, while the air phase is a compressible fluid.

For water phase, pressure is calculated from the density, and the following state equation is employed to handle quasi-incompressible fluids (Batchelor 1967).

$$p_d^w = p^w - p_0^w = p_0^w \left( \frac{\rho}{\rho_0} \right)^{\gamma_{\text{liq}}} - 1.0$$ (2.43)

where, the initial pressure of water, the pressure at an arbitrary time, and the pressure deviation are denoted by $p_0$, $p$, and $p_d$, respectively.

Here, $\gamma_{\text{liq}}$ is the parameter controlling the compressibility of liquid and set equal to 7.0 for the water phase (Monaghan 1994). Lower this value is, higher the compressibility.
of water is. This value was selected because the speed of compression waves in water is
defined by the bulk modulus and density. In Eq. (2.43), changes in density have a large
effect on changes in pressure; even small changes in density suffice to bring about large
changes in pressure. Actual calculations using this value showed changes of density of
less than 0.1%, a good approximation of incompressibility.

If the bulk modulus is denoted by \( B \), the equation of state for the air phase (Batchelor
1967) is given by,

\[
p_a^a = p^a - p_0^a = B \frac{D - \rho_0}{\rho_0}
\]  

(2.44)

where \( B \) is determined by,

\[
B = \gamma_{\text{air}} p_0^a
\]  

(2.45)

The ratio \( \gamma_{\text{air}} \) of specific heat at constant pressure to specific heat at constant volume
is 1.403 for the air phase.

The normalized density in Section 2.4 and The Courant-Friedrichs-Levy (CFL)
condition constrained time step in Chapter 3 eliminate the fluctuation of dynamic pressure.

2.6 Unsaturated constitutive model for the soil phase

Any constitutive model appropriate for the problem can be employed in the
framework of the SPH method. But, in order to simulate the surface infiltration and the
suction reduction, a novel constitutive model for unsaturated soils proposed by Zhang and
Ikariya (2011) is employed in this SPH method for its good performance (Xiong et al.
2014; Zhang et al. 2014).

First, the effective stress is defined by the concept of skeleton stress as shown in
following relation,

\[
\sigma_{\text{mn}}^e = \sigma_{\text{mn}}^f - U \delta_{\text{mn}}
\]  

(2.46)

\[
U = S_r p^v + (1 - S_r) p_a
\]  

(2.47)

where \( U \) is the mean pore pressure, \( \sigma_{\text{mn}}^f \) is the skeleton stress tensor, \( S_r \) is the degree of
saturation, $\sigma'_{mn}$ is the total stress tensor, $p^w$ is the water pressure, and $p^a$ is the air pressure. Eq. (2.46) means that the skeleton stress tensor is the difference of total stress tensor with mean pore pressure. Eq. (2.46) can also be rewritten as,

$$\sigma'_{mn} = \sigma_{mn} - p^w \delta_{mn} + S_r (p^v - p^w) \delta_{mn} = \sigma^\text{net}_{mn} + S_r \delta_{mn} \quad (2.48)$$

$$\sigma^\text{net}_{mn} = \sigma'_{mn} - p^w \delta_{mn} \quad s = p^a - p^w \quad (2.49)$$

Where $\sigma^\text{net}_{mn}$ is the net stress tensor and $s$ is the suction. Eq. (2.48) is just the definition of the effective stress by Bishop if taking the value $\chi$ in Bishop’s definition as $S_r$. The physical explanation for Eqs. (2.46) and (2.49), however, is different. For simplicity, throughout the context, the skeleton stress tensor $\sigma''$ will be abbreviated as $\sigma$ without specification in the following context. In addition, $m$ and $n$ in this section are indices of the stress tensor.

2.5.1 The elastic model

From elastic mechanics, the stress-strain relationship of elastic deformation can be represented as,

$$\sigma_{mn} = E_{ijkl} \varepsilon^e_{ij} \quad (2.50)$$

where $E_{ijkl}$ is a fourth-order elastic coefficient tensor, and $\varepsilon^e_{ij}$ is the elastic strain. The incremental form of the elastic stress–strain relationship can be expressed as,

$$d\sigma_{mn} = E_{ijkl} d\varepsilon^e_{ij} \quad (2.51)$$

Given the Young’s modulus $E$ and Poisson’s ratio $\nu$, the elastic coefficient matrix is:

$$[E] = \begin{bmatrix}
\lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\
\lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\
\lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\
0 & 0 & 0 & \mu & 0 & 0 \\
0 & 0 & 0 & 0 & \mu & 0 \\
0 & 0 & 0 & 0 & 0 & \mu
\end{bmatrix} \quad (2.52)$$
\[
\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} 
\]  \hspace{1cm} (2. 53)
\[
\mu = \frac{E}{2(1+\nu)} 
\]  \hspace{1cm} (2. 54)

Eq. (2.51) can be transformed to the following formation combining with Eq. (2.52),

\[
\begin{bmatrix}
\frac{d\sigma_{xx}}{d\varepsilon_{xx}} \\
\frac{d\sigma_{xy}}{d\varepsilon_{xy}} \\
\frac{d\sigma_{xz}}{d\varepsilon_{xz}} \\
\frac{d\sigma_{yx}}{d\varepsilon_{yx}} \\
\end{bmatrix} =
\begin{bmatrix}
\lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\
\lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\
\lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \mu & 0 \\
0 & 0 & 0 & 0 & 0 & \mu \\
\end{bmatrix}
\begin{bmatrix}
\frac{d\varepsilon_{xx}}{d\varepsilon_{xy}} \\
\frac{d\varepsilon_{xy}}{d\varepsilon_{yz}} \\
\frac{d\varepsilon_{xz}}{d\varepsilon_{yz}} \\
\frac{d\gamma_{xy}}{d\gamma_{yz}} \\
\end{bmatrix}
\]  \hspace{1cm} (2. 55)

\[
\frac{d\gamma_{mn}}{d\varepsilon_{mn}} = 2\frac{d\varepsilon_{mn}}{d\varepsilon_{mn}} 
\]  \hspace{1cm} (2. 56)

2.5.2 Unsaturated constitutive model

A quantitative relation for void ratio-logarithmic mean skeleton stress, \(e\)-\(\ln p\) relation, is established using the degree of saturation as a state variable. Here, it is assumed that the normally consolidated line in unsaturated state (N.C.L.S.) is parallel to the normally consolidated line in saturated state (N.C.L.) but in a higher position than N.C.L., as shown in Fig. 2.3, which means that under the same mean skeleton stress, the unsaturated soil can keep higher void ratio than that of saturated soil.

![Image of e-lnp relation considering moving up of N.C.L. and C.S.L. due to instauration (based on Zhang & Ikariya 2011).](image-url)
The \textit{N.C.L.S.} and \textit{C.S.L.S.} are given by following relations,

\begin{align*}
\text{N.C.L.S.: } e &= N(S_r) - \lambda \ln \frac{p}{p_r} \quad (\eta = \frac{q}{p} = 0) \quad (2.57) \\
\text{C.S.L.S.: } e &= \Gamma(S_r) - \lambda \ln \frac{p}{p_r} \quad (\eta = \frac{q}{p} = M) \quad (2.58)
\end{align*}

where, \(N(S_r)\) and \(\Gamma(S_r)\) are the void ratios at \textit{N.C.L.S.} and \textit{C.S.L.S.} under a reference mean skeleton stress \(p_r\) (usually \(p_r = 98\text{kPa}\)) and certain degree of saturation. \(p = \sigma_{ii}/3\) and 
\[q = \sqrt{3(\sigma_{ij} - p_{ij})(\sigma_{ij} - p_{ij})/2}\] are the mean skeleton stress and the second invariant of deviatory skeleton stress tensor. \(M\) is the stress ratio at critical state, and has the same value for saturated and unsaturated states. Therefore, similar to the derivation of Cam-Clay model for saturated soils, the void ratio \(e\) subjected to shearing is assumed to be,

\[e = \chi(\eta, S_r) - \lambda \ln \frac{p}{p_r} \quad (2.59)\]

where \(\chi(\eta, S_r)\) is a function of shear stress ratio \(\eta\) and the degree of saturation \(S_r\), and can be expressed with simple functions as,

(1) For Cam-clay type (Roscoe et al. 1963):
\[e = N(S_r) - \frac{N(S_r) - \Gamma(S_r)}{M} \eta - \lambda \ln \frac{p}{p_r} \quad (2.60)\]

(2) For Modified Cam-clay type (Schofield and Wroth 1968):
\[e = N(S_r) - \frac{N(S_r) - \Gamma(S_r)}{\ln 2} \ln \frac{M^2 + \eta^2}{M^2} - \lambda \ln \frac{p}{p_r} \quad (2.61)\]

Under an saturated isotropic normally consolidated state, that is, \(s = 0\), \(p = p_0\), \(\eta = 0\), \(S_r = 1\), \(N = N(S_r = 1)\), and \(e\) takes a value of \(e_0\), Eq. (2.61) can be expressed as,

\[e_0 = N - \lambda \ln \frac{p_0}{p_r} \quad (2.62)\]

Thus, we can get the void change based on the Modified Cam-clay type as,
\[-\Delta e = e_0 - e = N - N(S_r) + \frac{N(S_r) - \Gamma(S_r)}{\ln 2} \ln \frac{M^2 + \eta^2}{M^2} + \lambda \ln \frac{p}{p_0} \quad (2.63)\]

where \(\lambda\) is the compression index. Similar to the original Cam-Clay model, the elastic change of void ratio of unsaturated soil can be calculated with the swelling index \(\kappa\) as
Chapter 2 Principles of the water-air-soil three-phase coupled SPH method

\[-\Delta e^e = \kappa \ln \frac{p}{p_0}\]  

(2. 64)

It can be seen that unlike most constitutive models for unsaturated soils, both the compression index and swelling index are independent from the suction or the degree of saturation.

Then, the elastic volumetric strain can then be calculated as,

\[\varepsilon^e_i = \frac{-\Delta e^e}{1+e_0} = \frac{\kappa}{1+e_0} \ln \frac{p}{p_0}\]  

(2. 65)

The differential equation is,

\[d\varepsilon^e = \frac{\kappa}{1+e_0} \frac{dp}{p}\]  

(2. 66)

From Eq. (2.63) and Eq. (2.64), the plastic part of the change of void ratio can be obtained as,

\[-\Delta e^p = N - N(S_e) + \frac{N(S_e) - \Gamma(S_e)}{\ln 2} \ln \frac{M^2 + \eta^2}{M^2} + (\lambda - \kappa) \ln \frac{p}{p_0}\]  

(2. 67)

Therefore, the volumetric strain can also be divided into elastic and plastic parts, and the plastic part can be expressed as,

\[\varepsilon^p_i = \frac{-\Delta e^p}{1+e_0} = \frac{N - N(S_e)}{1+e_0} + \frac{N(S_e) - \Gamma(S_e)}{1+e_0} \ln 2 \ln \frac{M^2 + \eta^2}{M^2} + (\lambda - \kappa) \ln \frac{p}{p_0}\]  

(2. 68)

Eq. (2.68) is only suitable for the normally consolidated soil. For over consolidated soil, the concept of sub loading surface proposed by Hashiguchi and Ueno (1977) can easily be applied to unsaturated soil as,

\[\varepsilon^p_i = \frac{-\Delta e^p}{1+e_0} = \frac{N - N(S_e)}{1+e_0} + \frac{N(S_e) - \Gamma(S_e)}{(1+e_0)\ln 2} \ln \frac{M^2 + \eta^2}{M^2} + (\lambda - \kappa) \ln \frac{p^*}{p_0}\]  

(2. 69)

where \((p^*, q^*)\) represents a normally consolidated stress state through which a normal yielding surface passes, as shown in Fig. 2.4.
According to the similarity between the normal yielding surface and sub-loading yielding surface, the following relations can easily be obtained.

\[ \eta = \eta^* = \frac{q^*}{p^*} = \frac{q}{p} = \frac{p_{Ne}}{p_{N1}} \]  
\[ \frac{p^*}{p_0} = \frac{p}{p_0} \]  
\[ \frac{p^*}{p_{Ne}} = \frac{p}{p_{N1}} \]

Substituting Eq. (2.70) and Eq. (2.71) into Eq. (2.69), we can get the following equation,

\[ e_v = \frac{N - N(S_e)}{1 + e_0} \left( \frac{N(S_e) - \Gamma(S_e)}{1 + e_0} \right) \ln \frac{M^2 + \eta^2}{M^2} + \frac{(\lambda - \kappa)}{1 + e_0} \ln \frac{p}{p_{Ne}} \]  
\[ = \frac{N - N(S_e)}{1 + e_0} \left( \frac{N(S_e) - \Gamma(S_e)}{1 + e_0} \right) \ln \frac{M^2 + \eta^2}{M^2} + C_p \frac{p}{p_0} + \frac{\rho_e}{1 + e_0} \]

where,

\[ C_p = \frac{\lambda - \kappa}{1 + e_0}, \quad \rho_e = (\lambda - \kappa) \ln \frac{p_{Ne}}{p_{N1}} \]

\( \rho_e \) represents a void ratio difference between normally consolidated state and over-consolidated state under the same mean skeleton stress.

As shown in Fig. 2.3, a new state variable \( \rho_s \), which represents a void ratio difference between \( N.C.L. \) and \( N.C.L.S. \) under the same mean skeleton stress, can be introduced as,

\[ \rho_s = N(S_e) - N \]  

(based on Zhang & Ikariya 2011).
Therefore, Eq. (2.72) can be written as,
\[ \varepsilon^p = C_p \ln \frac{p}{p_0} + \frac{N(S_r) - \Gamma(S_r)}{(1+e_0) \ln 2} \ln \frac{M^2 + \eta^2}{M^2} - \frac{\rho_s}{1+e_0} + \frac{\rho_e}{1+e_0} \]  
(2.75)

Here, the yield surface can be described as,
\[ f = \ln \frac{p}{p_0} + \frac{N(S_r) - \Gamma(S_r)}{C_p (1+e_0) \ln 2} \ln \frac{M^2 + \eta^2}{M^2} - \frac{\rho_s}{1+e_0} C_p + \frac{\rho_e}{1+e_0} \frac{1}{C_p} - \varepsilon^p \frac{1}{C_p} = 0 \]  
(2.76)

By the definition of the critical state and some algebraic calculations, it is easy to obtain a useful relation as,
\[ d\varepsilon^p = \Lambda \frac{\partial f}{\partial p} \bigg|_{\eta=M} = 0 \quad \Rightarrow \quad N(S_r) - \Gamma(S_r) = (\lambda - \kappa) \ln 2 \]  
(2.77)

Thus, the yield function can be rewritten as,
\[ f = \ln \frac{p}{p_0} + \ln \frac{M^2 + \eta^2}{M^2} - \frac{\rho_s}{1+e_0} C_p + \frac{\rho_e}{1+e_0} \frac{1}{C_p} - \varepsilon^p \frac{1}{C_p} = 0 \]  
(2.78)

where,
\[ f_\sigma = \ln \frac{p}{p_0} + \ln \frac{M^2 + \eta^2}{M^2} \]  
(2.79)

From consistency equation \( df = 0 \), it is known that,
\[ df = \frac{\partial f}{\partial \sigma_{mn}} d\sigma_{mn} - d(\frac{\rho_s}{1+e_0} C_p) + d(\frac{\rho_e}{1+e_0} \frac{1}{C_p}) - d\varepsilon^p \frac{1}{C_p} = 0 \]  
(2.80)

here, it is necessary to give evolution equations for the state variables \( \rho_e \) of overconsolidation and \( \rho_s \) of saturation, and the flow rule for plastic strain tensor in the following ways,

(1) Associate flow rule: 
\[ d\varepsilon^p_{mn} = \Lambda \frac{\partial f}{\partial \sigma_{mn}} \]  
(2.81)

(2) 
\[ d(\frac{\rho_e}{1+e_0}) = -\Lambda \frac{\rho_s}{p}, \quad \rho = a\rho_e + b\rho_s \]  
(2.82)
Chapter 2 Principles of the water-air-soil three-phase coupled SPH method

\[
N(S_r) = N + \frac{N_r - N}{S_r^f - S_r^v} (S_r^f - S_r^v) \quad N_r = N(S_r^f)
\]

(3)

\[
\rho_s = N(S_r) - N = Q(S_r^f - S_r^v) \quad Q = \frac{N_r - N}{S_r^f - S_r^v}
\]

\[
d\rho_s = -QdS_r
\]

where, \(a\), \(b\) and \(\beta\) are the parameters controlling the evolution speed of \(\rho_e\), \(N_r\) is the void ratio of soil with residual saturation, \(S_r^s\) is the saturated saturation, and \(S_r^r\) is the residual saturation.

The strain increment can be divided into elastic and plastic parts as,

\[
d\varepsilon_{mn} = d\varepsilon_{mn}^e + d\varepsilon_{mn}^p
\]

(2.84)

Using Eq. (2.51), incremental stress tensor can be expressed as,

\[
d\sigma_{mn} = E_{mnkl} (d\varepsilon_{kl} - d\varepsilon_{kl}^p) = E_{mnkl} d\varepsilon_{kl} - E_{mnkl} \Lambda \frac{\partial f}{\partial \sigma_{kl}}
\]

(2.85)

Substituting this equation into Eq. (2.80), the following equation can be obtained,

\[
\frac{\partial f}{\partial \sigma_{mn}} E_{mnkl} d\varepsilon_{kl} - \frac{\partial f}{\partial \sigma_{mn}} E_{mnkl} \Lambda \frac{\partial f}{\partial \sigma_{kl}} - \Lambda \rho^p \frac{1}{p} \frac{Q}{C_p} + \frac{Q}{1 + e_0} dS_r \frac{1}{C_p} \left[ \frac{1}{C_p} \frac{\partial f}{\partial \sigma_{mm}} \right] = 0
\]

(2.86)

From above equation, we can get,

\[
\Lambda = \frac{h_p}{C_p} + \frac{1}{C_p \frac{\partial f}{\partial \sigma_{mn}} E_{mnkl} \frac{\partial f}{\partial \sigma_{kl}}} \frac{Q}{C_p}
\]

(2.87)

where, \(h_p = \frac{\partial f}{\partial \sigma_{mm}} + \rho^p \frac{Q}{p}
\)

(2.88)

Therefore it is easy to define the loading criteria as,

\[
\begin{align*}
\Lambda &> 0 \quad \text{loading} \\
\Lambda &= 0 \quad \text{neutral} \\
\Lambda &< 0 \quad \text{unloading}
\end{align*}
\]

(2.89)

Substituting Eq. (2.87) into Eq. (2.81),
\[ dE^p_{op} = \frac{\frac{\partial f}{\partial \sigma_{mn}} E_{mnkl} \, d\varepsilon_{kl} + \frac{1}{C_s} \frac{Q}{1+e_o} \, dS_r}{h_p} \] (2.90)

Meanwhile,

\[ d\sigma_{mn} = E_{mnkl} (d\varepsilon_{kl} - dE^p_{kl}) \]
\[ = E_{mnkl} d\varepsilon_{kl} - E_{mnqr} E_{opkl} \frac{\partial f}{\partial \sigma_{op}} \frac{\partial f}{\partial \sigma_{qr}} \frac{1}{D} d\varepsilon_{kl} - \frac{1}{C_p} \frac{Q}{1+e_o} \, dS_r \, \frac{1}{D} E_{mnqr} \frac{\partial f}{\partial \sigma_{qr}} \] (2.91)

\[ = (E_{mnkl} - E_{mnkl}^p) d\varepsilon_{kl} - A E_{mnkl} \frac{\partial f}{\partial \sigma_{kl}} \]

where,

\[ A = \frac{1}{C_s} \frac{Q}{1+e_o} \, dS_r \, \frac{1}{D}, \quad D = h_p + \frac{\partial f}{\partial \sigma_{op}} E_{opkl} \frac{\partial f}{\partial \sigma_{kl}} \] (2.92)

\[ E_{mnkl}^p = \frac{E_{mnqr} E_{opkl} \frac{\partial f}{\partial \sigma_{op}} \frac{\partial f}{\partial \sigma_{qr}}}{D} \] (2.93)

### 2.5.3 Definition of the moisture characteristic curve

In order to determine the degree of saturation from the suction, it is necessary to give a precise description of the moisture characteristics, considering the moisture hysteresis. Therefore, a suitable moisture characteristics curve for suction-saturation relation should include skeleton curves and scanning curves so that at any moisture state \((S_r, s)\), it is possible to obtain an incremental relation between the suction and the degree of saturation as,

\[ dS_r = k^{-1} \, ds \] (2.94)

where, \(k\) is the tangential stiffness of the suction-saturation relation.
Chapter 2 Principles of the water-air-soil three-phase coupled SPH method

Fig. 2.5 Image of moisture characteristic curve of unsaturated soil (based on Zhang & Ikariya 2011).

The skeleton curves for the moisture characteristics with tangential functions are given in three different ways according to the moisture state as,

(1) Primary drying curve from slurry:

\[ S_r = S_{r0} - \frac{2}{\pi} (S_{r0}' - S_r') \tan^{-1}\left(\frac{(\varepsilon_{r0} - 1)}{e^{\varepsilon_{r0}}}ight) \]  \hspace{1cm} (2.95)

(2) Secondary drying curve experienced drying-wetting process:

\[ S_r = S_{r1} - \frac{2}{\pi} (S_{r1}' - S_r') \tan^{-1}\left(\frac{(\varepsilon_{r1} - 1)}{e^{\varepsilon_{r1}}}ight) \]  \hspace{1cm} (2.96)

(3) Wetting curve:

\[ S_r = S_{r2} - \frac{2}{\pi} (S_{r2}' - S_r') \tan^{-1}\left(\frac{(\varepsilon_{r2} - 1)}{e^{\varepsilon_{r2}}}ight) \]  \hspace{1cm} (2.97)

where, \( S_d \) is a parameter corresponding to drying A.E.V. and \( S_w \) is a parameter corresponding to W.E.V., as shown in Fig. 2.5. \( c_1 \) and \( c_2 \) are scaling factors that controlling the shape of moisture curves. \( S_{r0} \) is the degree of saturation of a slurry under fully
saturated condition and is equal to 1.0.

As to the scanning curve in the process of drying-wetting process between the skeleton curves, the incremental relation between suction and saturation is expressed as,

\[ k_{s1}^{-1} = k_{s0}^{-1} + k_{s1}^{-1} \]  

(2.98)

\[ k_{s0} \] is the gradient of suction-saturation relation under the condition that inner variable \( r \) equals to 0. \( k_{s1} \) is expressed as:

\[ k_{s1} = k_{s0}^r (1 + c_3 \frac{1-r}{r}) \]  

(2.99)

where, \( c_3 \) is a scaling factor which controlling the curvature of the scanning curve. \( k_{s0}^r \) is the gradient of the corresponding skeleton curve on which the moisture state \((S_r, s)\) is locating under the condition that \( r \) equals to 1, as shown in Fig. 2.5. The inner variable \( r \) is defined as,

\[ r = \begin{cases} \frac{\delta_s}{\delta_s} & ds > 0 \\ \frac{\delta_r}{\delta_s} & ds \leq 0 \end{cases} \]  

(2.100)

Eq. (2.98) means that the stiffness of \( k_s \) consists of two parts, \( k_{s0} \) and \( k_{s1} \) in a way that its value looks like the value of a spring consisted from two series springs. It is easily understood from Eq. (2.98) and Eq. (2.99) that if \( r=0 \), \( k_{s1} \) will be infinite and \( k_s = k_{s0} \). If \( r =1 \) and \( k_{s0} \gg k_{s1} \), \( k_s \) will equal to \( k_{s1} \), which coincides with the gradient of the skeleton curve. This explanation can also be easily understood by means of the illustration shown in Fig. 2.5.

Eight parameters are involved in this moisture characteristic curve, among which three parameters \( c_1, c_2 \) and \( c_3 \) are determined with curve fitting method while other five parameters, \( k_{s0}, S_r^s, S_r^r, S_d \) and \( S_w \) have definite physical meaning and can be determined by the test of moisture characteristics easily.

Another method to calculate the saturation is using the particle approximation of SPH method. It is assumed that the water particle has a saturation of \( S_r^s \), and the air particle has a saturation of 0.0. Then, the degree of saturation can be obtained from the following equation,
where \( S^w \) is the water saturation. It is a simple method, not relying on the SWCC.

### 2.7 The strain rate tensor and Jaumann rate tensor of the Cauchy stress

According to the first assumption in Section 2.3, the strain rate used for the calculation of stress tensor (water, air, and soil) is determined by the following equation,

\[
\dot{\varepsilon}^{mn} = \frac{1}{2} \left( \frac{\partial v^m}{\partial x^n} + \frac{\partial v^n}{\partial x^m} \right)
\]

(2.102)

where \( m \) and \( n \) indicates the direction of coordination.

The particle approximation of Eq. (2.102) for each phase can be rewritten as,

\[
\dot{\varepsilon}^{mn}_i = \frac{1}{2} \sum_{j=1}^{N} \frac{m_j}{\rho_j} \frac{1}{W_{ij}} \left[ (v^m_i - v^m_j) \frac{\partial W_{ij}}{\partial x^n} + (v^n_i - v^n_j) \frac{\partial W_{ij}}{\partial x^m} \right]
\]

(2.103)

where, \( i \) indicates a computing particle, \( j \) indicates the supporting particle, \( N \) is the number of supporting particles, \( W_{ij} \) is the smoothing kernel function of particle \( i \) evaluated at particle \( j \), \( x \) is the position vector, and \( v \) is the velocity.

Many constitutive equations are designed in the form of a relation between a stress-rate and a strain-rate (or the rate of deformation tensor). The mechanical response of a material should not depend on the reference frame. In other words, material constitutive equations should be frame indifferent (objective). If the stress and strain measures are material quantities then objectivity is automatically satisfied. However, if the quantities are spatial, then the objectivity of the stress-rate is not guaranteed even if the strain-rate is objective. Thus, the Jaumann rate tensor of Cauchy stress can be employed to obtain the objective stress.

The Jaumann stress rate tensor is defined as,

\[
\mathbf{\sigma}^v = \mathbf{\dot{\sigma}} + \mathbf{\omega} \cdot \mathbf{\sigma} - \mathbf{\sigma} \cdot \mathbf{\omega}
\]

(2.104)

where \( \mathbf{\sigma}^v \) is the Jaumann stress rate tensor that is calculated from the constitutive model, \( \mathbf{\dot{\sigma}} \) is the stress rate tensor for time integration, and \( \mathbf{\omega} \) is the spin tensor.
Therefore, the stress rate tensor for time integration is as follow,

\[ \dot{\sigma} = \sigma - \sigma \cdot \omega + \omega \cdot \sigma \]  

(2.105)

The spin tensor can be obtained from the following equation,

\[ \omega_{i}^{m} = \frac{1}{2} \sum_{j \neq i}^{N} \frac{m_{j}}{\rho_{j}} \left[ (v_{i}^{m} - v_{j}^{m}) \frac{\partial W_{ij}}{\partial x_{n}} - (v_{i}^{n} - v_{j}^{n}) \frac{\partial W_{ij}}{\partial x_{m}} \right] \]  

(2.106)

**2.8 Interaction forces between different phases**

For this research, the water phase, the air phase, and the soil phase were handled on different layers (see Fig. 2.2). These layers were then combined to obtain the three-phase system.

When combining the solid layer and the fluid layer, this SPH method adopts the physical forces described below, with due consideration of the effects of the porosity \( n_{p} \) and the coefficient of permeability \( k \) for water or air. The frictional body forces resulting from velocity differences between two adjacent phases are employed as the above physical forces. The mixture theory (Biot 1941; Prevost 1980) is used here, writing the force by the soil phase on the water phase, and the force by the soil phase on the air phase, respectively, \( R \) and \( Q \),

\[ R_{m} = n^w \left( \frac{\rho^w g}{k^w} (v^{wm} - v^{m}) \right) \]  

(2.107)

\[ Q_{m} = n^a \left( \frac{\rho^a g}{k^a} (v^{am} - v^{m}) \right) \]  

(2.108)

where, \( R_{m} \) is the interaction force by the soil phase on water phase at coordination \( m \), \( Q_{m} \) is the interaction force by the soil phase on air phase at coordination \( m \). \( k^w \) and \( k^a \) are permeability coefficients of water and air, respectively. \( v^{wm}, v^{am}, \) and \( v^{sm} \) are the velocity of water phase, air phase and soil phase at coordination \( m \), respectively. \( n^w \) and \( n^a \) can be obtained from following equations (Mori et al. 2011),

\[ n^w = n_{p} S_{t} \]  

(2.109)

\[ n^a = n_{s} (1 - S_{t}) \]  

(2.110)
where \( n_p \) is the porosity.

Because there are commonly many supporting particles for a computing particle, it is needed to summate the interaction force by the particle approximation, thus the SPH formation of interaction forces can be presented as follows,

\[
\text{water} : R^\alpha_i = \sum_{j=1}^{N} m_j \frac{R^\alpha_{ij}}{\rho_j} W_{ij} \quad (2.111)
\]

\[
\text{air} : Q^\alpha_i = \sum_{j=1}^{N} m_j \frac{Q^\alpha_{ij}}{\rho_j} W_{ij} \quad (2.112)
\]

\[
\text{soil} : R^\alpha_i + Q^\alpha_i = \sum_{j=1}^{N} m_j \frac{R^\alpha_{ij}}{\rho_j} W_{ij} + \sum_{j=1}^{N} m_j \frac{Q^\alpha_{ij}}{\rho_j} W_{ij} \quad (2.113)
\]

### 2.9 Summary

In the proposed water-air-soil three-phase SPH model, fluids and solid are simulated on different layers. The three-phase field theory was introduced and related quantities were derived. Then, governing equations including the continuity equation and the momentum equation were transformed to the SPH formulas for water, air, and soil, respectively. The equation of state was used to calculate the dynamic pressure of the water phase and the air phase. After that, we introduced the unsaturated constitutive model to describe the behavior of soil phase. Meanwhile, the proposed model used the frictional forces, with due consideration of the effects of the porosity and the coefficient of permeability, as interaction forces between different phases.

### References


Chapter 2 Principles of the water-air-soil three-phase coupled SPH method
Chapter 3

SPH programs and their parallel optimization

3.1 General

In this chapter, the smoothing kernel function and particle searching algorithm are summarized, compared, and selected to obtain the high efficiency and accuracy simultaneously. Then, the boundary treatment method and the integration method are described. At last, the parallel optimization of the ZM-OP (OpenMP) SPH program and the ZM-CD (CUDA) SPH program are introduced.

3.2 Smoothing kernel function

As mentioned in the previous chapter, the smoothing kernel function is very important in establishing governing equations in the SPH method. The order of the smoothing kernel function determines the accuracy of particle approximation, but high order means high time consumption (Liu & Liu 2003).

The kernel function must satisfy several conditions (Liu & Liu 2010) as,

(1) The smoothing function must be normalized over its supporting domain (Unity condition). This normalization property ensures that the integral of the smoothing function over the supporting domain to be unity.

\[ \int_{\Omega} W(x-x', h)dx' = 1 \]  \hspace{1cm} (3.1)

(2) The smoothing function should be compactly supported (Compact support). The dimension of the compact support is defined by the smoothing length \( h \) and a scaling factor \( \kappa \), where \( h \) is the smoothing length, and \( \kappa \) determines the spread of the specified smoothing function.
Chapter 3 SPH program and their parallel optimization

\[ W(x - x', h) = 0, \text{for} |x - x'| > \kappa h \quad (3.2) \]

(3) \( W(x - x') \geq 0 \) for any point at \( x' \) within the support domain of the particle at point \( x \) (Positivity). This property states that the smoothing function should be non-negative in the support domain.

(4) The smoothing function should be an even function (Symmetric), and the value should be monotonically decreasing with the increase of the distance away from the particle (Decay).

(5) The smoothing function should satisfy the Dirac delta function condition as the smoothing length approaches zero (Delta function property).

\[ \lim_{h \to 0} W(x - x', h) = \delta(x - x') \quad (3.3) \]

(6) The smoothing function should be smooth enough to ensure the accuracy of particle approximation (Smoothness).

Until now, many smoothing kernel functions have been proposed and applied in various fields, such as the bell-shaped function, the Gaussian Kernel function, the cubic-spline function, the B-spline function, and so on (Liu & Liu 2010).

Among these functions, the B-spline function based on the cubic-spline function was widely used, for its good accuracy and efficiency (Monaghan & Lattanzio 1985). Therefore, we employ the B-spline function as the smoothing function in this research.
B-spline function and its 1st order differential form are defined as,

\[
W(s, h) = \alpha_d \begin{cases} 
\frac{2}{3} - s^2 + \frac{1}{2} s^3 & 0 \leq s \leq 1 \\
\frac{1}{6} (2 - s)^3 & 1 \leq s \leq 2 \\
0 & s \geq 2 
\end{cases}
\]

(3.4)

\[
\frac{\partial W(s, h)}{\partial x} = \alpha_d \begin{cases} 
\left(-2s + \frac{3}{2} s^2\right) \frac{(x_i - x_j)}{hr} & (0 \leq s < 1) \\
\frac{1}{2} (2 - s)^2 \frac{(x_i - x_j)}{hr} & (1 \leq s < 2) \\
0 & (2 \leq s) 
\end{cases}
\]

(3.5)

where, \( s = |x - x_j|/h \). For two dimensions, \( \alpha_d \) is \( 15/(7\pi h^2) \) and for three dimensions, \( \alpha_d \) is \( 3/(2\pi h^3) \). Fig. 3.1 shows the B-spline smoothing function and its 1st order differential form for two dimensions.

### 3.3 Particle searching algorithm

In the SPH method, physical quantities of a computing particle are calculated by summating physical quantities of neighboring particles in the supporting domain. The particle position in the SPH method is updated at each time step, thus, particle searching is performed at each time step and the efficiency of SPH is largely determined by the particle searching algorithm.

Three kinds of searching algorithms are often used in the SPH method: all-pair searching algorithm, linked-list searching algorithm and tree searching algorithm (Liu & Liu 2003).
Chapter 3 SPH program and their parallel optimization

All-pair searching algorithm is a directly searching algorithm. Given a computing particle, this method calculates distances between it and all other particles (see Fig. 3.2). If the distance of one particle to the computing particle is less than or equal to double smoothing length, it is regarded as the supporting particle. The time complexity of this algorithm is $O(N^2)$, where $N$ is the total number of particles.

Linked-list searching algorithm places grids or cells on the problem domain, as shown in Fig. 3.3. The size of grid is same as the size of the supporting domain. Only particles in adjacent grids of a computing particle are selected as candidates for supporting particles (Fig. 3.4). The time complexity of this method is $O(N)$, therefore, this algorithm is highly efficient dealing with problems with the constant smoothing length. But, for cases with a variable smoothing length, the efficiency will be greatly decreased because of the remesh.

Fig. 3.2 All-pair searching algorithm.

Fig. 3.3 Grids of Linked-list searching algorithm (left: 2D; right: 3D).
Chapter 3 SPH program and their parallel optimization

Tree searching algorithm is based on the linked-list searching algorithm and has high efficiency dealing with problems with variable smoothing length. The time complexity of this algorithm is \( O(N \log N) \).

In this research, the linked-list (verlet-list) algorithm is employed for the particle searching, and improvement has been carried out to obtain a better efficiency.

**Fig. 3. 4 Linked-list searching algorithm.**

Several variables are used in the improved linked-list algorithm. They are: celllink

**Fig. 3. 5 Variables for grids.**

Several variables are used in the improved linked-list algorithm. They are: celllink
storing the neighboring grids No. or ID for each grid, cellinfo storing the particle ID in the grid, and cellno indicating the cell ID where a particle belongs. The structures of celllink and cellinfo are shown in Fig. 3.5, where c indicates a certain grid, n is element ID, and the first element of cellinfo is the total number of particles in this grid. In addition, variables, influ and wij shown in Fig. 3.6 and Fig. 3.7, represents supporting particles and values of the smoothing kernel function, respectively. Same as the cellinfo, the first element of influ is the total number of supporting particles.

The neighboring grids of an inner grid can be determined by the following equation:

\[
\text{celllink}(i, n) = (\text{dim} - 1)(\text{dim} - 2)(l - 2)nXm \cdot nYm / 2 + (\text{dim} - 1)(m - 2)nXm + (n - 2) + i
\]

where \(\text{dim}\) is the dimensions of problem, \(nXm\) is the number of grids at \(x\) coordinate, \(nYm\) is the number of grids at \(y\) coordinate, and \(nZm\) is the number of grids at \(z\) coordinate. \(l, m,\) and \(n\) are the loop variables equal to 1, 2 and 3. Eq. (3.6) indicates that the grid inside a problem domain has three neighboring grids for one dimension, nine neighboring grids for two dimensions, and 27 neighboring grids for three dimensions, including itself. But, boundary grids have less neighboring grids than inner grids.

![Diagram of celllink and cellinfo structures](image)

Fig. 3.6 Variable for supporting particles.
The improved searching algorithm can be summarized as:

1. At the beginning of a simulation, the problem domain is determined and the size of grid is set to double smoothing length. Then, $nXm$, $nYm$, and $nZm$ are calculated. After that, $celllink$ is initialized and calculated according to Eq. (3.6).

2. For every step, $cellinfo$ and $cellno$ are initialized. Then, calculate the grid ID that each particle belongs to, put the particle into the grid, and write the total number of particles in grid into the first element of $cellinfo$.

3. While searching supporting particles, get the grid ID of the computing particle from $cellno$, and obtain the neighboring grid IDs from $celllink$. Then, the particles in these grids are regarded as candidates. Calculate distance of the computing particles to each candidate. If a distance is less than or equal to double smoothing length, the number of supporting particles ($n$) is plus one ($n+1$) and related particle ID is written into the position $n+1$ of $influ$. Meanwhile, calculate values of smoothing kernel function and write them into the position $n+1$ of $wij$.

In other SPH subroutines, supporting particles and values of smoothing function can be directly used. Fig. 3.8 is the flow chart of searching algorithm. It can be found that Step (1) are only executed at the beginning of simulation, while Step (2) and Step (3) are performed at every time step, therefore, the improved linked-list algorithm can be an effective particle searching algorithm.
3.4 Boundary treatment

To realize a better physical meaning of the boundary, the phase boundary is reproduced by creating an array of virtual boundary particles. Here, we adopt the boundary treatment method proposed by Takeda et al. (1994) and Morris et al. (1997). It can be assumed that the boundary particle has a virtual velocity $v_B$, and the effect of $v_B$ on a moving particle $a$ is calculated according to the distance between two particles (Fig. 3.9).
Fig. 3.9 Boundary treatment method.

The boundary effect can be calculated from following equations,

\[ \mathbf{v}_{ab}\mathbf{x}_{aB} = (v_{ax} - v_{Bx})(x_a - x_B) + (v_{ay} - v_{By})(y_a - y_B) \]
\[ + (z_a - z_B)(z_a - z_B) \]  
\[ (3.7) \]

\[ \beta = \min \left( \beta_{\text{max}}, 1 + \frac{d_B}{d_a} \right) \]  
\[ (3.8) \]

\[ v_{cx} = \frac{(1 - \beta) \cdot \mathbf{v}_{ab}\mathbf{x}_{aB}(x_a - x_B)}{r_{ab}^2} \]  
\[ (3.9) \]

\[ v_{cy} = \frac{(1 - \beta) \cdot \mathbf{v}_{ab}\mathbf{x}_{aB}(y_a - y_B)}{r_{ab}^2} \]  
\[ (3.10) \]

\[ v_{cz} = \frac{(1 - \beta) \cdot \mathbf{v}_{ab}\mathbf{x}_{aB}(z_a - z_B)}{r_{ab}^2} \]  
\[ (3.11) \]

where, \( v_c \) is the velocity effect of boundary particle, \( r_{ab} \) is the distance between the moving particle and the boundary particle, \( \beta_{\text{max}} \) is equal to 2.0, \( \mathbf{v}_{ab} \) is the relative velocity vector, and \( \mathbf{x}_{aB} \) is the relative position vector.

This method can allow the adjustment of different parameters, and seems promising for the modeling of various boundary conditions.
3.5 Time integration method

The second order Runge-Kutta method is employed as the time integration method, which is presented as,

\[ v(t + \Delta t) = v(t) + \frac{a(t) + a(t + \Delta t)}{2} \Delta t \]  
\[ u(t + \Delta t) = u(t) + \frac{v(t) + v(t + \Delta t)}{2} \Delta t \]  
\[ t = t + \Delta t \]  

The time step \( \Delta t \) is constrained by the Courant-Friedrichs-Levy (CFL) condition,

\[ \Delta t_1 \leq 0.25 \min(h_i / c_i) \]  
\[ \Delta t_2 \leq 0.25 \min(\sqrt{h_i / |a_i|}) \]  
\[ \Delta t = \min(\Delta t_1, \Delta t_2) \]  

where \( c \) is the velocity of sound, \( h \) is the smoothing length and \( a_i \) is the acceleration vector.

3.6 Parallel computing

Traditionally, a computer software was written for the serial computation. To solve a problem, an algorithm is constructed and implemented as a serial stream of instructions. These instructions are executed on a central processing unit (CPU) on one computer. Only one instruction may execute at a time, and after that instruction is finished, the next is executed. The frequency of central processing unit greatly influences the time consumption of compute-bound programs, and the consuming time of a program is equal to the number of instructions multiplied by the average time per instruction. Maintaining everything else constant, increasing the clock frequency decreases the average time it takes to execute an instruction (Hennessy & Patterson 2011). But larger frequency of CPU means larger power consumption, thus the frequency cannot be increased unlimitedly so that efficiency of the serial computing is limited by the serial computing capacity of a CPU.

Flynn’s taxonomy is a classification of computer architectures, proposed by Flynn in
1972 (Flynn 1972). There are four classifications in the Flynn's taxonomy based on the number of concurrent instruction and data streams in the architecture as shown in Fig. 3.10.

(1) Single Instruction, Single Data stream (SISD): a sequential computer that exploits no parallelism in either the instruction or the data stream. It means that the controlling unit only fetches one instruction and the processing unit only operates single data stream at a time. Early PCs, such as the IBM PC and 8-bit PC, belong to this classification.

(2) Single Instruction, Multiple Data streams (SIMD): the computer operates single instruction stream, and with many same processing elements, it processes multi data streams. The examples are single-core CPUs (with MMX, Streaming SIMD Extensions (SSE), SSE2, SSE3, SSSE3, SSE4, and AVX), array processors and General-Purpose computing on Graphics Processing Units (GPGPUs).

(3) Multiple Instructions, Single Data stream (MISD): the computer of this classification adopts multiple instruction streams to process single data stream. This classification is not common in practice except the flight-controlling computers of the Space Shuttle.

(4) Multiple Instructions, Multiple Data streams (MIMD): the computer that can deal with multiple instructions and multiple data streams at the same time. Examples of MIMD systems are all SMP CPUs, such as, multi-core CPUs and Intel Xeon Phi.

The potential speedup of a program from parallel computing is given by the Amdahl's law (Amdahl 1967). An algorithm is usually consisting of several parallelizable parts and several non-parallelizable parts. If the consuming time of an algorithm is one
on sequential platform and the fraction of non-parallelizable parts is $\alpha$, the consuming time of this algorithm on a parallel platform can be obtained by the following equation,

$$t = \alpha + \frac{1-\alpha}{N}$$  

(3.18)

where $t$ is the consuming time of an algorithm on a parallel platform and $N$ is the number of processing units.

The speedup can be defined as,

$$s = \frac{1}{\alpha + \frac{1-\alpha}{N}}$$  

(3.19)

where $s$ is the speedup. Therefore, the limitation of speedup is,

$$\lim_{N \to \infty} \lim_{\alpha \to \infty} \frac{1}{\alpha + \frac{1-\alpha}{N}} = \frac{1}{\alpha}$$  

(3.20)

Above equation reveals that the speed of a parallel program is mainly restricted by the non-parallelizable parts, also called the Cannikin Law. Thus, more parallelizable parts in an algorithm, more effective can it be from the parallel computing.

The memory in the parallel system is either shared memory or distributed memory. The shared memory refers the memory that different processors or threads can access in a parallel system. The distributed memory refers to a multiple-processor computer system where each processor has its private memory, such as clusters and super computers. The CPU-to-memory or GPU-to-memory connection and the cache coherence are still main issues in the parallel programming. When one cache that is used by one processor is updated with new information, the change needs to be reflected to other processors, otherwise the different processors will be working with the incoherent data, and the computing will output wrong results (Gerber 2012).

In order to explain the issue of cache coherence, we can take the following codes for example. The codes are written using C language and FORTRAN language, respectively, with the Open Multi-Processing (OpenMP) parallelization.
### C language

```c
#include <stdio.h>

int main(){
    int a[1000],i,sum;
    sum=0;
    #pragma omp parallel for
    for(i=0;i<1000;i++){
        a[i]=i+1;
        sum+=a[i];
    }
    printf("sum=%d\n",sum);
    return 0;
}
```

### FORTRAN language

```fortran
program testc
    implicit none
    integer a(1000),i,add

    add=0
    !$omp parallel do
    do i=1,1000
        a(i)=i
        add=add+a(i)
    enddo
    write(*,1) add
1 format("sum=",i8)
    stop
end
```

<table>
<thead>
<tr>
<th>Core 0</th>
<th>Core 1</th>
<th>Core 2</th>
<th>Core 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>sum0+a[i]</td>
<td>sum0+a[i+1]</td>
<td>sum0+a[i+2]</td>
<td>sum0+a[i+3]</td>
</tr>
</tbody>
</table>

**Fig. 3.11 Issue of incoherent cache.**

Above codes are used to assign values from 1 to 1000 to `a[1000]` and calculate the sum of array `a[1000]`. The correct result is 500, 500. But, incoherent cache occurs in above
codes, and results are random, as shown in Fig. 3.11.

To ensure the cache coherence, several methods can be used, such as, the critical section, the lock operation, and the atomic operation. Besides, some functions provided by the API and programming languages also have the function of ensuring cache coherence. The above example can be revised by the reduction directive of OpenMP API to obtain the correct result.

<table>
<thead>
<tr>
<th>C language</th>
<th>FORTRAN language</th>
</tr>
</thead>
</table>
| #include <stdio.h>  
#include <omp.h>  
int main(){  
    int a[1000],i,sum;  
    sum=0;  
    #pragma omp parallel for reduction(+:sum)  
    for(i=0;i<1000;i++){  
        a[i]=i+1;  
        sum+=a[i];  
    }  
    printf("sum=%d\n",sum);  
    return 0;  
} | program testc  
implicit none  
integer a(1000),i,add  
add=0  
!$omp parallel do reduction(+:add)  
do i=1,1000  
a(i)=i  
add=add+a(i)  
enddo  
write(*,1) add  
1 format("sum=",i8)  
stop  
end |

Differently from the parallel computing on CPU, for example, OpenMP and Message Passing Interface (MPI), GPGPU uses the graphic processing unit (GPU) to conduct the general-purpose computation that is traditionally performed on CPU. The modern GPUs have powerful capability of parallel computing and programmable pipelines, thus they can deal with non-graphic data. As a result, the performance of GPGPU surpasses the performance of traditional CPU especially for the case of SIMD
that is of more data computing than data scheduling and transmission (NVIDIA 2015).

### 3.7 The ZM-OP SPH program and ZM-CD SPH program

In order to improve the efficiency of the proposed SPH method, the OpenMP API and the CUDA language are used to parallelize the SPH program for CPU and GPGPU, respectively.

#### 3.7.1 The ZM-OP SPH program

OpenMP, maintained by the OpenMP Architecture Review Board, is the programming API that supports multi-thread platform with shared memory architecture in C, C++ and FORTRAN (Barney 2014). It is a cross-platform API, thus, the OpenMP parallelized program can easily be ported to other platforms with no revision or little revision.

OpenMP is the implementation of multithreading using the runtime environment to allocate a master thread forking a specified number of slave threads that can run concurrently on different processors (Amritkar et al. 2012) (see Fig. 3.12). The section of code that is meant to run in parallel is marked accordingly with a preprocessor directive that will fork slave threads before the section is executed. Therefore, it is easy to study and to realize the coarse-grained parallel computing using OpenMP.

The clause of `#pragma omp parallel` is used to fork additional threads to carry out the work enclosed in the construct. The original thread will be denoted as master thread with thread ID 0.
The work-sharing constructs can be used to specify how to assign independent work to one or all of the threads. Related clauses are as follows,

1. *omp for* or *omp do*: used to split up loop iterations among the threads, also called loop constructs.
2. *sections*: assigning consecutive but independent code blocks to different threads.
3. *single*: specifying a code block that is executed by only one thread, and a barrier is implied in the end.
4. *master*: similar to single, but the code block will be executed by the master thread only and no barrier is implied in the end.

OpenMP is based on the shared-memory architecture, thus the data sharing is very important and OpenMP has defined many data-sharing clauses.

1. *shared*: the data within a parallel region is shared, which means visible and accessible by all threads simultaneously. By default, all variables in the work sharing region are shared except the loop iteration counter.
2. *private*: the data within a parallel region is private to each thread, which means each thread will have a local copy and use it as a temporary variable. A private variable is not initialized and the value is not maintained for use outside the parallel region. By default, the loop iteration counters in the OpenMP loop constructs are private.
3. *default*: allows the programmer to state that the default data scoping within a
parallel region will be either shared, or none for C/C++, or shared, firstprivate, private, or none for Fortran. None option forces the programmer to declare each variable in the parallel region using the data sharing attribute clauses.

(4) firstprivate: like private except initialized to original value.
(5) lastprivate: like private except original value is updated after construct.
(6) reduction: a safe way of joining work from all threads after construct.

The thread synchronization is provided by following clauses,

(1) critical: the enclosed code block will be executed by only one thread at a time, and not simultaneously executed by multiple threads. It is often used to protect shared data from race conditions.
(2) atomic: the memory update (write, or read-modify-write) in the next instruction will be performed atomically. It does not make the entire statement atomic; only the memory update is atomic. A compiler might use special hardware instructions for better performance than using critical.
(3) ordered: the structured block is executed in the order in which iterations would be executed in a sequential loop.
(4) barrier: each thread waits until all of other threads of a team have reached this point. A work-sharing construct has an implicit barrier synchronization at the end.
(5) nowait: specifies that threads completing assigned work can proceed without waiting for all threads in the team to finish. In the absence of this clause, threads encounter a barrier synchronization at the end of the work-sharing construct.

Besides, OpenMP has also provided some useful functions as,

(1) int omp_get_thread_num(): returns the thread ID executing within its thread team.
(2) int omp_get_num_threads(): returns the number of threads in the parallel region.
(3) void omp_set_num_threads(int): sets the number of threads in subsequent parallel regions, unless overridden by a num_threads clause.

and so on.

With OpenMP API, a highly parallelized SPH program, called the ZM-OP SPH, was written using the C++ programming language for its object-oriented nature and good
support of pointer. Among SPH subroutines, the particle searching algorithm is the most difficult part to be parallelized, especially the subroutine of putting particles into grids, for this subroutine involves the reading and writing of the first element of \textit{cellinfo} simultaneously by multi-threads, and improper handling will cause the problem of incoherent cache. To avoid this problem, the \textit{critical} section is used to assure the cache coherence as following,

\begin{verbatim}
{
    /*parallel section*/
    ...
    /*parallel codes*/
    /*critical section*/
    
    #pragma omp critical
    {
        cellinfo[p][0]++;
        ...
    }
    /*other serial codes*/

}
\end{verbatim}

Other SPH subroutines can be easily parallelized by the following code,

\begin{verbatim}
#pragma omp parallel for private(ntd,id,i,...)
for(ntd=0;ntd<cn;ntd++){
    id=ntd;
    for (i=id;i<ctrpro.ntotal;i+=cn){
        ...
    }
}
\end{verbatim}

where, \textit{cn} is the number of threads in parallel regions that can be obtained from the function \textit{omp_get_num_threads()} or assigned by the function \textit{omp_set_num_thread()}, and \textit{ctrpro.ntotal} is the total number of particles.

The flow chart of the ZM-OP SPH program is shown in Fig. 3.13. Meanwhile, the compiling and usage of this program will be presented in Appendix, and the efficiency will be discussed in the next chapter.
Chapter 3 SPH program and their parallel optimization

Fig. 3.13 Flow chart of ZM-OP SPH program.
3.7.2 The ZM-CD SPH program

CUDA (Compute Unified Device Architecture) is a parallel computing platform and programming model created by NVIDIA Corporation. The CUDA computing is implemented on the graphics processing units (GPUs) that NVIDIA produces, supporting C, C++ and FORTRAN, and is the dominant proprietary framework for GPGPU programming (NVIDIA 2015). Driven by the insatiable market demand for real-time and high-definition 3D graphics, the programmable GPU has evolved into a highly parallel, multithreaded, many core processor with tremendous computational power and very high memory bandwidth comparing to CPU (Harada et al. 2007; Satish et al. 2009; Goswami et al. 2010; Hérault et al. 2010).

CUDA provides three key abstractions for programmers, a hierarchy of thread groups, shared memories, and a barrier synchronization (NVIDIA 2015). These abstractions provide a fine-grained data parallelism and thread parallelism, nested within the coarse-grained data parallelism and task parallelism. Programmers can partition the problem into coarse sub-problems independently in parallel by blocks of threads (Fig.
3.14) and each sub-problem into finer pieces cooperatively in parallel based on CUDA abstractions.

Streaming Multiprocessors (SMs) are main execute units in GPU. The multi-threaded program is partitioned into blocks of threads and different SM executes different blocks (Fig. 3.14). Therefore, a GPU of more SMs will automatically execute the same program in less time than that of fewer SMs. The program can be run on any number of multiprocessors for the scalable programming model (NVIDIA 2015) in CUDA, where only the runtime needs to know the number of physical multiprocessors.

There are three types of function declaration in CUDA: __host__, __global__, and __device__.

(1) __host__: as the implicit declaration in CUDA, declares the function that executes on CPU calling other __host__ and __global__ functions.

(2) __global__: declares the function that executes on GPU that can be called by the __host__ function and is able to call the __device__ function.

(3) __device__: declares the function that executes on GPU, which can only be called by the __global__ function.

and so on.

Fig. 3. 15 The grid in CUDA.
Differently from OpenMP, threads in CUDA are organized in block and each thread can be identified by its index \((\text{threadIdx})\). For a one-dimensional block, the thread ID is same to the index \((x)\); for a two-dimensional block, the ID of a thread with index \((x, y)\) is \((x+yDx)\); for a three-dimensional block, the ID of a thread with index \((x, y, z)\) is \((x+yDx+zDxDy)\). \(Dx, Dy\) and \(Dz\) are dimensions of blocks \((\text{blockDim})\). Similarly to thread, blocks can be assembled in Grid and identified by \(\text{blockIdx}\). Fig. 3.15 shows the thread, block and grid in CUDA.

In the CUDA programming model, host function and device function maintain their own separate memory spaces in DRAM, referred to host memory and device memory. Therefore, a CUDA program must manage global, local, constant, shared, and texture memory spaces for kernels, including the device memory allocation, device memory free, and data transfer between host and device. Some memory operations are defined in CUDA as,

1. \(\text{cudaMalloc ( void** devPtr, size_t size )}\): allocate memory on device.
2. \(\text{cudaFree ( void** devPtr )}\): free memory on device.
3. \(\text{cudaMemcpy ( void* dst, const void* src, size_t count, cudaMemcpyKind kind )}\): copy data between host and device.

\(\text{Kind}\) determines the type of data transfer and can be \(\text{cudaMemcpyHostToHost}\), \(\text{cudaMemcpyHostToDevice}\), \(\text{cudaMemcpyDeviceToHost}\), \(\text{cudaMemcpyDeviceToDevice}\), and \(\text{cudaMemcpyDefault}\).

and so on.

Based on above, a typical CUDA program has the structure or flow chart presented in Fig. 3.16. The dimensions of grid and block, which are mainly determined by the number of particles, are preferably equal to the value of \(2^n\).
Chapter 3 SPH program and their parallel optimization

For CUDA program, the data transfer between host and device is always a large time consumption because of the insufficient of PCI-E bandwidth. Therefore, a well-parallelized CUDA program should place almost all subroutines on GPU (Harada et al. 2007; Goswami et al. 2010; Hérault et al. 2010).

Same as the OpenMP SPH program, particle searching algorithm is also a hard work. There are several methods that can implement the searching algorithm on GPU, such as the Z-indexing (Goswami et al. 2010), Radix sort (Satish et al. 2009), and atomic operations (Green 2008).

We adopted the atomic operations for the parallel optimization of particle searching algorithm. Atomic operations allow distinct concurrent threads to update the same memory area without conflict. For the code in Section 3.6.1, it can be written as,

```
__global__ void cellno(...){ /*CUDA kernel function*/
...
```

Fig. 3. 16 Structures and flow chart of CUDA program.
where atomicAdd reads a word at some address in global or shared memory, adds a number to it, and writes the result back to the same address.

The other SPH subroutines can be easily parallelized by the following code,

```c
__global__ void cellno(...){ /*CUDA kernel function*/
    unsigned int id=blockIdx.x*blockDim.x+threadIdx.x;
    int ntotal;
    ...
    ntotal=gParacr->ntotal;
    ...
    for (i=id;i<ntotal;i+=gridDim.x*blockDim.x){
        ...
        /*other codes*/
    }
}
```

ZM-CD SPH program, which can almost fully run on GPU, has been written by C/C++ and CUDA programming languages. The flow chart is shown in Fig. 3.17. Meanwhile, the compiling and usage of ZM-CD SPH program will be presented in Appendix, and the efficiency will be discussed in the next chapter.
Chapter 3 SPH program and their parallel optimization

Fig. 3.17 Flow chart of ZM-CD SPH program.
3.8 Summary

The B-spline function was selected as smoothing kernel function to obtain high accuracy and efficiency. Based on the Linked-list searching method, a highly effective searching method was proposed and applied in the proposed SPH programs. Meanwhile, critical section in OpenMP API and atomicAdd in CUDA solved the incoherent issue of parallel optimization. After that, using the C++ programming language, highly parallelized SPH programs were written, called ZM-OP (OpenMP) SPH program and ZM-CD (CUDA) SPH program.

References


Chapter 3 SPH program and their parallel optimization
Chapter 4

Verification and validation of the SPH method

4.1 General

This chapter shows the SPH simulations of a dam break, a tri axial compression test, the rising and burst of air bubble, a falling-head permeability test, a flowing test of liquefied soils, and a 3D dam break, to verify and validate the application and efficiency of the proposed SPH method.

4.2 Dam break

Dam break is a typical problem of the free surface in fluid dynamics, therefore, this research simulated the two dimensional and three dimensional dam breaks to verify the modeling of the water phase.

4.2.1 Two dimensional dam break

The size of the 2D dam break case was 1m×0.5m, and the water column was in the left side of the model box with a dimension of 0.17m×0.35m (Fig. 4.1). In the discrete model, there were totally 1, 232 particles, and 548 particles were boundary particles. Initial particle spacing was 1cm. At the initial stage, water abuts the left boundary, while the right side of the water column is unconstrained. Then, water particles simply flowed along the fixed boundary at the base of the model box under the action of gravity. The parameters in the simulation are shown in Table 4.1. Three cases were simulated with different coefficients of XSPH as follow,

\[
v_i = v_i' - \varepsilon \sum_{j=1}^{N} m_j \left( v_i' - v_j' \right) W_{ij} \quad (4.1)
\]
where, $\varepsilon$ is a constant between 0 and 1. Larger of $\varepsilon$ is, greater effect of this correction for the velocity to match the average velocity of surrounding particles is.

Table 4.1 SPH parameters of the 2D dam-break simulation.

<table>
<thead>
<tr>
<th>Density $\rho$</th>
<th>Viscosity</th>
<th>Time step</th>
<th>Total steps</th>
<th>Case 1</th>
<th>Case 2</th>
<th>Case 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000 kg/m$^3$</td>
<td>0.001 Pa·s</td>
<td>0.001 s</td>
<td>500</td>
<td>$\varepsilon=0.1$</td>
<td>$\varepsilon=0.3$</td>
<td>$\varepsilon=0.5$</td>
</tr>
</tbody>
</table>

![Fig. 4.1 Dimension of 2D dam break.](image)

Flowing configurations of the 2D dam break is shown in Fig. 4.2, where the contour is the water pressure (unit: Pa).

Martin and Moyce (1952) developed a comparative scaling law for tests and numerical simulations that can verify the accuracy of the proposed SPH method. The notations used in this paper are $T=\frac{t(2g/L)^{0.5}}{}$ and $Z=X/L$, where $X$, $L$, and $t$ are the distance of the surge front from the axis, the characteristic dimension of the column base, and the time, respectively. The SPH results were compared with the results calculated by above equations using data available in literature (Martin & Moyce 1952), as indicated in Fig. 4.3. The SPH results agree with the scaling law, thereby demonstrating the accuracy of the SPH simulation for the water phase. In addition, the XSPH correction of 0.3 is the most near to the benchmark, corresponding with the conclusion of Liu and Liu (2003).
Fig. 4.2 Flowing configurations of 2D dam break (Case 2).

Fig. 4.3 Comparison of SPH simulation and analytical solutions for 2D dam break.

4.2.2 Three dimensional dam break

A three-dimension dam break case was built of 1m (length) × 0.1m (width) × 0.4m
(height), and the water column was in the left side of the model box with a dimension of 0.1m×0.1m×0.2m. There were totally 18,089 particles, and 8 particles were used to define border planes. The initial spacing of particles is 0.005m. At the initial stage, water abuts the left boundary, while the right side of the water column is unconstrained. Then, the water simply flowed along the fixed boundary at the base of the model under the action of gravity (z direction). Only the XSPH of 0.3 was simulated here, while other parameters were the same as the 2D dam break. Flowing configurations of the 3D dam break is shown in Fig. 4.4.

![Flowing configurations of 3D dam break.](image)

The normalized time and flowing distance of SPH simulation are shown in Fig. 4.5 with the benchmark, from which the SPH result agrees with the scaling law.
In order to verify the application of the proposed SPH method in the solid phase (soil), the discrete model of a tri axial compression test (Fig. 4.6) was built and simulated using the unsaturated constitutive model. Here, the sample size was 20 cm × 10 cm, and the initial particle spacing was 1 cm. The bottom was a fixed boundary and the confining stress, which was constant in the process of simulation, was exerted around the soil sample. At the top of the soil sample, loading surface was controlled by a constant strain rate. SPH parameters are shown in Table 4.2 and constitutive model parameters are presented in Table 4.3 (based on Zhang and Ikariya 2011).
Chapter 4 Verification and validation of the SPH method

Fig. 4. 6 The triaxial compression test.

Table 4.2 SPH parameters of the triaxial compression test.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density</td>
<td>1,409 kg/m³</td>
</tr>
<tr>
<td>Strain</td>
<td>0.000001/step</td>
</tr>
<tr>
<td>Total steps</td>
<td>6,000</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.01 m</td>
</tr>
<tr>
<td>Confining stress</td>
<td>50 kPa</td>
</tr>
</tbody>
</table>

Saturation (case 1): 0.680
Saturation (case 2): 0.641
Suction (case 1): 200 kPa
Suction (case 2): 400 kPa

Table 4.3 Constitutive parameters of the triaxial compression test.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compression index ( \lambda )</td>
<td>0.050</td>
</tr>
<tr>
<td>Swelling index ( \kappa )</td>
<td>0.010</td>
</tr>
<tr>
<td>Critical state parameter ( M )</td>
<td>1.0</td>
</tr>
<tr>
<td>Void ratio ( N ) ( (p' = 98) kPa on N.C.L.)</td>
<td>1.14</td>
</tr>
<tr>
<td>Poisson’s ratio ( \nu )</td>
<td>0.30</td>
</tr>
<tr>
<td>Parameter of overconsolidation ( a )</td>
<td>5.00</td>
</tr>
<tr>
<td>Parameter of suction ( b )</td>
<td>1.0</td>
</tr>
<tr>
<td>Parameter of overconsolidation ( \beta )</td>
<td>1.0</td>
</tr>
<tr>
<td>Void ratio ( N_r ) ( (p' = 98) kPa on N.C.L.S.)</td>
<td>1.28</td>
</tr>
</tbody>
</table>
Fig. 4.7 shows the simulated stress, strain and dilatancy relations, with the model test data from Cui & Delage (1996), and Fig. 4.8 indicates the skeleton stress path from the SPH simulation. From Fig. 4.7, at the initial stage of axial strain development, the deviator stress and volumetric strain corresponded well with the test results. But after that, the deviator stress and volumetric strain deviated from the test result, for the simplification of constitutive model and the particle nature of SPH method. Even so, the soil of a smaller saturation kept larger deviator stress than that of a larger saturation at the same axial strain. Meanwhile, the volumetric strain of a smaller saturation was smaller than that of a larger saturation at the same axial strain. Therefore, development trends of deviator stress and volumetric strain corresponded with model test results, and the SPH simulation reflected the different strain-stress relationship and volumetric strain development due to different suctions. Additionally, the strength reduction by the suction
reduction was found in the SPH simulation; thus, the proposed SPH model can consider the effect of suction reduction in the process of seepage failures due to heavy rainfall.

![Simulated skeleton stress path.](image)

**Fig. 4.** Simulated skeleton stress path.

### 4.4 Air bubble rising and burst from water

A water-air coupled case (Fig. 4.9) was built to simulate the rise and burst of air bubble from water. There were totally 1,685 particles in this case. Among these particles, 1,621 water particles and 60 air particles were placed in the lower part of the model, while the upper part was set to vacuum with a pressure of 100kPa (1 atm). The water area was only filled with water particles and the air area was only filled with air particles. At the initial stage, water particles and air particles, resisting by the fixed boundary, were fixed, and moved freely after the beginning of simulation.
Chapter 4 Verification and validation of the SPH method

Fig. 4.9 Water-air coupled case.

Fig. 4.10 Rising and burst of air bubble from water.
For the SPH simulation, the initial density of air $\rho_0$ was set to 1.207 kg/m$^3$ and viscosity $\mu$ was set to 1.810x10$^{-5}$ Pa·s. For the balance of water and air, the dynamic pressure of air bubble at initial stage must equal to the water pressure induced by self-weight and the air density should be set to the value calculated from the equation of state. In addition, for the air phase, the dynamic pressure was used to calculate the acceleration.

Fig. 4.10 shows the simulated process of air rising and burst from water. Air bubble rises with the passage of time due to the effect of buoyancy. The outline of air bubble changes as it rises, and the surface of water also lifts above the bubble. As it rises, the bubble breaks into a nearly straight row of rising bubble. The validation of the motion of air phase is very complex issue in computational fluid dynamics and geomechanics, and preliminary results were obtained from the SPH simulation, based on which the behavior of the air phase can be checked. The motion of the air bubble in water is mainly dominated by the buoyancy and air pressure, for which the air bubble will rise and burst from the water. Thus, simulated air rise and burst correspond with the derived phenomenon of the buoyancy and air pressure. Related result has been published in the literature (Zhang & Maeda 2015).

### 4.5 Falling head permeability test

![Fig. 4.11 Falling head permeability test.](image)

The SPH modeling of the interaction force between the water phase and the soil
phase was verified by a falling head permeability test shown in Fig. 4.11. At the initial stage, the water level in the left side was 0.15 m higher than the right side. The soil sample was 0.1 m in length and 0.05 m in width. Only water particles could move under gravity in the simulation, while soil particles and boundary particles were fixed. In this simulation, there were 66 soil particles, 336 water particles and 404 boundary particles. Table 4.4 was the parameters for simulation.

### Table 4.4 SPH parameters of the permeability test.

<table>
<thead>
<tr>
<th>Density of water (kg/m³)</th>
<th>Density of soil particles (kg/m³)</th>
<th>Porosity</th>
<th>Permeability (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,000</td>
<td>2,650</td>
<td>0.45</td>
<td></td>
</tr>
</tbody>
</table>

According to the Darcy’s law, the water level difference at time \( t \) can be calculated by the following equation,

\[
H = \frac{H_0}{e^{\frac{kt_0}{L}}} - H
\]  

(4.2)

In above equation, \( t_0 \) was the initial time, \( H_0 \) was the initial water level difference, and \( L \) was the length of soil sample. For this case, \( t_0 \) was 0s, \( H_0 \) was 0.15m, and \( L \) was 0.1m.

![Fig. 4.12 Falling head permeability test.](image)

The calculated water level differences by Eq. (4.2) are shown in Fig. 4.12. Meanwhile, water level differences from SPH simulations are also shown in Fig. 4.12 for
a comparison. It can be seen that the SPH simulated results correspond well with theory solutions, which indicates that the proposed SPH method can make a good simulation of the interaction force between the water phase and the soil phase. Related result has been published in the literature (Huang et al. 2013).

4.6 Flowing process of liquefied soils

Using an experimental apparatus, a physical model test of flowing liquefied sand has been designed and conducted, from which the pore water pressure and the flowing distance of liquefied soil were obtained. Using the same test conditions, we have also conducted a soil-water-coupled SPH simulation to capture the dynamic behavior of flowing soils after liquefaction and validate the application of proposed SPH method on water-soil-coupled case. Related result was published in literatures (Huang et al. 2013; Huang et al. 2014).

![Fig. 4.13 Model test box.](image)

The experimental device is divided into five parts: a motor that includes a reduction gear, a slider-crank, a shaking table, a model box and a base (Huang et al. 2012). The model box, containing saturated sand, has outside dimensions of 98 × 35 × 34 cm and is placed on the vibrating base. The model box is constructed of transparent organic glass with grids etched onto it to facilitate the observation of flow configurations. Pressure and velocity can be measured with probes installed in the model box and a high-speed camera system (Huang et al. 2011), respectively. In Fig. 4.13, two pore pressure meters are shown that are installed in the soil sample to measure pore pressure while the apparatus undergoes vibration. Using the high-speed camera system, configurations of the material during the flowing process can be recorded. The left side of model box was separated
from the right side by a non-leaking baffle. Then, water was injected into the left side of the model box with the height of about one third of the box’s height. Subsequently, the sandy soil was spread by the hopper and seen to be fully saturated. The soil sample covers an area of $32 \times 15$ cm. Vibrations of 1 Hz were imposed on the shaking table to liquefy the sand sample. When the excess pore pressure reached a maximum, as shown in Fig. 4.14, the baffle was removed and the liquefied sand flowed under gravity. The high-speed camera record indicates that the flow lasted for six seconds.

![Time series of excessive pore pressure.](image)

**Fig. 4.14 Time series of excessive pore pressure.**

A discrete model of the test was built and simulated by the proposed SPH method. The discrete model consisted of 528 water particles, 528 soil particles and 564 boundary particles. Water particles and soil particles could occupy the same positions, and flowed under gravity. The parameters used in the simulation are shown in Table 4.5. Note that the parameters in Table 4.5 were obtained from the laboratory test results for the soil sample.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density of water (kg/m³)</td>
<td>1,000</td>
</tr>
<tr>
<td>Density of soil particles (kg/m³)</td>
<td>2,681</td>
</tr>
<tr>
<td>Young’s modulus E (initial)</td>
<td>10 MPa</td>
</tr>
<tr>
<td>Poisson ratio ν</td>
<td>0.3</td>
</tr>
<tr>
<td>Porosity</td>
<td>0.60</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.01 m</td>
</tr>
<tr>
<td>Unit time step</td>
<td>0.00003 s</td>
</tr>
<tr>
<td>Total steps</td>
<td>200,000</td>
</tr>
</tbody>
</table>
Elastic model was used in this simulation. The viscosity of liquefied soil decreases with increasing shear strain. Besides, in the flowing process of liquefied soil, the shear module will also decrease as shear strain develops (Hadush et al. 2001). Therefore, the
elastic module will be decayed in the SPH simulation. The simulated configurations of flowing liquefied sand are shown in Fig.4.15 and compared with the test configurations. The simulated results are observed to have good agreement with the model test.

Figs. 4.16 and 4.17 show the SPH simulated velocity vectors for soil and water, respectively. From the velocity vectors, the flowing velocity is significant for the first three seconds after the removal of baffle. After three seconds, the liquefied sand at the free surface boundary continued to flow, but because of the dissipation of excessive pore pressure, the shear strength increased and the shear strain rate decreased. The flow ceased at six seconds after the removal of baffle. The overall flowing process, as determined by the SPH analysis, is characterized by water velocities that are larger than those of the soil. As a result, the flow distance of water was greater than that of soil, and water seeped out of soil. In the model test, water also seeped from soil, and flowed out in front at the late flowing stage. This phenomenon could explain the decrease in the velocity vectors: water seeps from the soil, and then the excessive pore pressure is dissipated.

![Fig. 4. 16 SPH simulated velocity vectors for soil.](image)

(1) 1s  (2) 2s  (3) 3s  (4) 4s  (5) 5s  (6) 6s
Chapter 4 Verification and validation of the SPH method

4.7 Efficiency of ZM-OP SPH program

Fig. 4. 17 SPH simulated velocity vectors for water.

Fig. 4. 18 3D dam break model for the validation of efficiency.
To check the efficiency of the parallel optimization for the proposed SPH program, a three-dimension dam break model has been built and simulated. From the simulation of different threads with the ZM-OP SPH program, the different consuming time was obtained and compared.

The length of the 3D dam break model was 0.4 m, the width was 0.1 m, and the height was 0.4 m. Water particles were aligned to one boundary plane and had a dimension of 0.1 m × 0.1 m × 0.2 m (see Fig. 4.18). In the simulation, water particles could move under gravity, while the boundary was fixed. There were 18, 089 particles in the discrete model, 18, 081 of which were water particles, and eight particles were used to define border planes. For the simulations, CPU was an Intel Core i7 860 with four cores and eight threads (Hyper-Threading Technology [HT], the trademark of Intel Corporation), and the frequency ranged from 2.8 GHz to 3.5 GHz. The specification of CPU is shown in Table 4.6. Here, six cases with different numbers of threads (shown in Table 4.7) were simulated.

![Table 4.6 Specification of the CPU.](image)

<table>
<thead>
<tr>
<th>CPU name</th>
<th>Threads</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache</td>
<td>8MB</td>
<td>Processor Base Frequency</td>
</tr>
<tr>
<td>Instruction Set</td>
<td>64bit</td>
<td>Max Turbo Frequency</td>
</tr>
<tr>
<td>Cores</td>
<td>4</td>
<td>OS</td>
</tr>
</tbody>
</table>

![Table 4.7 Number of threads used in the simulations of the 3D dam break.](image)

<table>
<thead>
<tr>
<th>Case</th>
<th>Case 2</th>
<th>Case 3</th>
<th>Case 4</th>
<th>Case 5</th>
<th>Case 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threads (Cores)</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

Different cases obtained the same flowing configurations, which are presented in Fig. 4.19, where the single time step is 0.0001s. Meanwhile, Fig. 4.20 shows the consuming time of different threads for 3, 000 steps. It is easily seen that the consumed time decreases as the threads increase. However, the speedup over four threads is not obvious because over four threads are using logical threads rather than physical threads. In spite of this, considering the four physical cores, the consumed time with eight threads
was one third of that with a single thread, which proves that the ZM-OP SPH program is a highly effective simulation tool on CPU.

Table 4.8 Number of threads used in the simulations of the 3D dam break.

<table>
<thead>
<tr>
<th>Non-parallelizable parts $\alpha$</th>
<th>2 threads</th>
<th>3 threads</th>
<th>4 threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.136</td>
<td>0.1176</td>
<td>0.1171</td>
<td></td>
</tr>
</tbody>
</table>

Using Eq. (3.19) in Chapter 3, the fraction $\alpha$ of non-parallelizable parts can be obtained as shown in Table 4.8, and the fraction of non-parallelizable parts of the proposed SPH method is around 0.117, therefore, the maximum speedup is about 8.55.

Fig. 4. 19 SPH simulated flowing configurations of 3D dam break.
4.8 Efficiency of ZM-CD SPH program

With the same 3D dam break in Fig. 4.18, we conducted a simulation using the ZM-CD SPH program with different GPUs. Five different GPUs were used to obtain different speedups, and specifications of them are presented in Table 4.9. Other parameters are same as the simulation of the ZM-OP SPH program.

The compute capability of a device is represented by a version number, also sometimes called its "SM version". This version number identifies the features supported by the GPU hardware and is used by applications at runtime to determine which hardware features and/or instructions are available on the present GPU (NVIDIA 2015). The compute capability version comprises a major $x$ and a minor version number $y$ ($x.y$). A CUDA program can be only executed on the GPU of higher or equal compute capability than itself.

The simulation of the 3D dam break on GPUs was conducted on the Windows 7 Professional Edition, and generated the same flowing configurations as presented in Fig. 4.19. Fig. 4.20 shows the consuming time on different GPUs for 3,000 steps. As we can see, ZM-CD SPH program has higher efficiency than the serial SPH program except GeForce 8400GS. In addition, the efficiency of ZM-CD SPH program on GTX TITAN is
about 8.5 times as the serial simulation. This value is close to the maximum speedup 8.55, thus we can see the great power of parallel computing on GPGPU, and furthermore the proposed SPH programs can take advantages of the advanced parallel computing both on the CPU and GPGPU.

Table 4.9 GPUs used in ZM-CD SPH simulations of 3D dam break.

<table>
<thead>
<tr>
<th>Date</th>
<th>Code Name</th>
<th>Compute Capability</th>
<th>Streaming Multiprocessor</th>
<th>Processing Power GFLOPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geforce8400GS</td>
<td>2009/4</td>
<td>GT218</td>
<td>1.1</td>
<td>1^1</td>
</tr>
<tr>
<td>GTS 250</td>
<td>2009/3</td>
<td>G92-428-B1</td>
<td>1.1</td>
<td>16^1</td>
</tr>
<tr>
<td>GTX 470</td>
<td>2010/3</td>
<td>GF100</td>
<td>2.0</td>
<td>14^2</td>
</tr>
<tr>
<td>GTX TITAN</td>
<td>2013/2</td>
<td>GK110-400-A1</td>
<td>3.5</td>
<td>14^3</td>
</tr>
<tr>
<td>Tesla K20</td>
<td>2013/2</td>
<td>GK110</td>
<td>3.5</td>
<td>15^5</td>
</tr>
</tbody>
</table>

1 Each Streaming Multiprocessor(SM) in the chip of G80/GT200 architecture contains 8 SPs and 2 SFUs.
2 Each Streaming Multiprocessor(SM) in the GPU of GF100 architecture contains 32 SPs and 4 SFUs.
3 Calculation formula of GFLOPS is different.
4 Single precision.

![Fig. 4.21 Consumed time of different GPUs.](image-url)
4.9 Summary

From the simulations of a dam break, a tri axial compression test and a water-air coupled case, the proposed SPH method can make a good simulation of the water phase, the air phase, the soil phase, and furthermore the effect of the suction reduction to the soil strength. Through the simulations of a falling head permeability test and a flowing test of liquefied soil, the coupling of water and soil was verified and validated. After that, the proposed SPH method was proved to be highly effective numerical method by a three-dimension dam break.

References

Chapter 4 Verification and validation of the SPH method


Chapter 5

Application of the proposed SPH method to seepage failures due to heavy rainfall

5.1 General

In this chapter, three seepage failure tests are simulated by the proposed SPH method. At first, the seepage failure due to piping is simulated with two cases: the homogeneous dike and the heterogeneous dike. Then, a dike failure test due to heavy rainfall is simulated with the two-phase coupled SPH method. At last, a slope failure by heavy rainfall is simulated with two cases: the two-phase coupled simulation and the three-phase coupled simulation. Some mechanisms of seepage failures are presented in this chapter.

5.2 The water-soil-coupled SPH simulations of seepage failure due to the water level up

The water level up, triggered by the heavy rainfall, is one of major contributors to the seepage failure of the dike and levee. Besides, from the case of Yabe-river in Chapter 1, the piping mechanism is still not very clear. Therefore, in order to obtain the different seepage behavior of different materials, failures due to water level up were simulated with two cases: one was the homogeneous dike, and another was the heterogeneous dike of different permeability.

5.2.1 Seepage failure within the homogeneous dike
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5.1 Dimension of the homogeneous dike.

Based on one model test system, a model test of the homogeneous dike failure due to water level up has been conducted. The model test system can be divided into four parts: the model box, the rainfall and water inject device, the measuring device, and the image acquisition device. The bottom of the model box in this test was undrained bottom. A basic sand layer, fully saturated, was setup and the dike model was placed on the saturated sand layer.

The dimension of the homogeneous dike is shown in Fig 5.1. Masa sand was used as the material of the dike, as shown in Fig. 5.2, which had a density of 2, 644 kg/m$^3$ and a permeability in the saturated state of $2.3\times10^{-3}$ m/s, according to the experiment results. The particles diameter mainly varied from 0.07 mm to 2 mm. The water level up was exerted in the right side of model box (see Fig. 5.3). During the model test and the SPH simulation, the water level at the right side (upstream) was increased to 300 mm during the initial 30 min. After that, water level remained constant for 60 min. At the final stage, water level was raised till the failure of the dike body.

Fig. 5.2 Dike model in the test.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

As with the model test, a discrete model, presented in Fig. 5.4, was built for the two-phase-coupled SPH simulation. The discrete model consisted of 11,338 particles in total, among which 2,133 particles were boundary particles, 4,885 particles were soil particles, 2,211 particles were water particles in base layer, and 2,109 particles were particles for water level up. The parameters used in this simulation is shown in Table 5.1 and the dike body had a homogeneous permeability that can be determined by the following equation,

\[
k = \begin{cases} 
5.0 \times 10^{-5} & S_r \leq 0.2 \\
3.4615 \times 10^{-3} S_r - 6.4231 \times 10^{-4} & 0.2 < S_r \leq 0.85 \\
2.3 \times 10^{-3} & 0.85 < S_r 
\end{cases} 
\]  

(5.1)

At first, soil particles and water particles had no velocity but could move under gravity in the simulation, while the boundary particles were fixed in the whole analysis process. At the first stage of the SPH simulation, the initial stress was generated by the elastic model. To simplify the simulation, the state variable \( \rho_e \) was zero at the initial stage (no overconsolidation). Then, the artificial rainfall started, and when the seepage line reached, the behavior of soil was described by the unsaturated model.
Table 5.1 Parameters of the homogeneous dike simulation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density of water $\rho_w$</td>
<td>1,000 kg/m³</td>
</tr>
<tr>
<td>Density of soil particle $\rho_s$</td>
<td>2,644 kg/m³</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.01 m</td>
</tr>
<tr>
<td>Saturated void ratio $N_s$</td>
<td>1.14</td>
</tr>
<tr>
<td>Unsaturated void ratio $N_r$</td>
<td>1.28</td>
</tr>
<tr>
<td>Saturated saturation $S_r$</td>
<td>0.85</td>
</tr>
<tr>
<td>Residual saturation $S_r'$</td>
<td>0.20</td>
</tr>
<tr>
<td>Porosity</td>
<td>0.533</td>
</tr>
<tr>
<td>Poisson ratio $\nu$</td>
<td>0.25</td>
</tr>
<tr>
<td>Evolution parameter $a$, $b$, $\beta$</td>
<td>0.05, 0.05, 1.0</td>
</tr>
</tbody>
</table>

Fig. 5.5 shows the seepage process from the SPH simulation. In order to check the result, seepages from the model test are also presented in Fig. 5.5. The seepage line developed from right to left with the time same as the model test. Meanwhile, the seepage line at 30 min is close to the model test, but seepage lines at 90 min and 120 min are behind the model test.

Fig. 5.5 Seepage process of SPH simulation and model test.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5.6 Maximum shear strains of SPH simulation.

Distributions of maximum shear strain from the SPH simulation are presented in Fig. 5.6. The maximum shear strain increased with time. Meanwhile, due to the seepage of the rising water level in the right side, the maximum distribution mainly occurs at the left toe of the dike. At 90 min and 120 min, the maximum shear strain formed a slip surface in the left side, indicating that the left part is prone to fail under the seepage of rising water. Besides, under the action of the pore water pressure and the seepage force, the erosion and large deformation of soil occurs in the SPH simulation, and mainly distributes at the left toe of the dike, proving that the proposed SPH method can simulate the erosion and large deformation of soil in the process of seepage failures.

5.2.2 Seepage failure within the heterogeneous dike

In order to check the effect of different permeability on the seepage failure, a heterogeneous dike was built and simulated with two cases: case 1 and case 2 with two
layers of different permeability. The dimension and configuration of soil layers are shown in Fig. 5.7.

![Fig. 5.7 Dimension of the heterogeneous dike.](image)

Table 5.2 Permeability of the heterogeneous dike simulation.

<table>
<thead>
<tr>
<th></th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dike</td>
<td>2.0×10⁻⁶ m/s</td>
<td>5.0×10⁻⁸ m/s</td>
</tr>
<tr>
<td>Base layer</td>
<td>2.0×10⁻⁴ m/s</td>
<td>5.0×10⁻³ m/s</td>
</tr>
</tbody>
</table>

The main difference between case 1 and case 2 is the permeability difference between the dike body and the base layer. The base layer of case 1 was 100 times as the dike body, and case 2 was 10,000 times, as shown in Table 5.2. Differently from the previous SPH simulation, the permeability maintained constant in the whole process of simulation. Two cases used the same material, and its constitutive parameters as well as the SPH parameters are listed in Table 5.3.

Table 5.3 Parameters of the heterogeneous dike simulation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density of water ρ_w</td>
<td>1,000 kg/m³</td>
</tr>
<tr>
<td>Compression index λ</td>
<td>0.04</td>
</tr>
<tr>
<td>Density of soil particle ρ_s</td>
<td>2,644 kg/m³</td>
</tr>
<tr>
<td>Swelling index κ</td>
<td>0.008</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.005 m</td>
</tr>
<tr>
<td>Critical state parameter M</td>
<td>1.00</td>
</tr>
<tr>
<td>Saturated void ratio N</td>
<td>0.97</td>
</tr>
<tr>
<td>Saturated saturation S_r</td>
<td>0.99</td>
</tr>
<tr>
<td>Unsaturated void ratio N_r</td>
<td>1.00</td>
</tr>
<tr>
<td>Residual saturation S_r</td>
<td>0.01</td>
</tr>
<tr>
<td>Porosity</td>
<td>0.492</td>
</tr>
<tr>
<td>Poison ratio ν</td>
<td>0.30</td>
</tr>
<tr>
<td>Evolution parameter</td>
<td>a b β</td>
</tr>
<tr>
<td>b</td>
<td>0.05 0.05 1.0</td>
</tr>
</tbody>
</table>
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

A discrete model was built for the SPH simulations, and is presented in Fig. 5.8. There were totally 13,875 particles in the discrete model, among which 3,028 particles were boundary particles, 5,849 particles were soil particles, 3,417 particles were water particles in base layer, and 1,581 particles were used for the water level up in left of the dike body. At first, soil particles, and water particles had no velocity but could move under gravity in the simulation. The boundary particles were fixed in the whole analysis process. At the first stage of the SPH simulation, the initial stress was generated by the elastic model. To simplify the simulation, the state variable $\rho_e$ was zero at the initial stage (no overconsolidation). After that, the rainfall started, and when the seepage line reached, the behavior of soil was described by the unsaturated soil model.

Figs. 5.9 and 5.10 indicate seepage processes of Case 1 and Case 2 from SPH simulations, respectively. It can be found that both the seepage lines of Case 1 and Case 2 develop with the time. The seepage state of Case 2 is a little later than Case 1 at 30 min. However, the seepage state of Case 2 at 90 min is a little earlier than Case 1. Seepage lines for both cases are around the base layer and the lower part of dike.

Fig. 5. 8 Discrete model for SPH simulations of the heterogeneous dike.

Fig. 5. 9 Seepage process of Case 1 from the SPH simulation.

(1) 30min

(2) 90min
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5.10 Seepage process of Case 2 from the SPH simulation.

Figs. 5.11 and 5.12 present distributions of the maximum shear strain for two cases from SPH simulations. Compared to the previously homogeneous dike, the maximum shear strain of the heterogeneous dike are very large, and dikes encounter very large deformation in the seepage progress. Case 1 has a base layer of smaller permeability, thus, a larger interaction forces occurs and the maximum shear strains are larger than Case 2. Meanwhile, the erosion occurs in both cases, and there is more water-eroded soil in Case 1 than that in Case 2 due to the larger interaction force. But, for Case 2, the upper part of the dike encounters a larger deformation and loses its configuration. Therefore, the base layer of high permeability under a dike is a negative factor for the stability due to water level up.

The proposed SPH method presents the large deformation, the erosion, the related water-eroded soil, and different seepages of different permeability, which is evidenced in real seepage failures of dike due to water level up, thus it can be a useful simulation tool for such failures.
5.3 The water-soil-coupled SPH simulation of dike failures due to the rainfall infiltration

A model test of the dike failure due to the rainfall infiltration was conducted on the same test device as previous section, and two different rainfall intensities were considered:
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

the low intensity (30 mm/hr.) and the high intensity (90 mm/hr.). The dimension of the dike model is described in Fig. 5.13. Masa sand was used as the material of the dike, which had a density of 2,644 kg/m$^3$, and the permeability in the saturated state of $2.3 \times 10^{-3}$ m/s, according to the experiment results. The particles diameter mainly varied from 0.07 mm to 2 mm.

![Dimension of the dike model for the failure due to rainfall infiltration.](image)

Based on the test conditions, a discrete SPH model shown in Fig. 5.14 was built for SPH simulations. There were totally 9,818 particles in the discrete model, among which 4 particles were used to define border lines, 4,885 particles were soil particles, 2,211 particles were water particles, and 2,718 particles were rain particles. As with the test, the base layer was saturated, and unsaturated dike body was placed on saturated base layer. At first, soil particles, water particles, and rain particles had no velocity but could move under gravity in the simulation, while the boundary particles were fixed in the whole analysis process. The SPH parameters are shown in Table 5.4, and the constitutive model parameters are presented in Table 5.5. In addition, the relationship of saturation and permeability in Eq. (5.1) was also used to describe the permeability change in the seepage process.

At the first stage of the SPH simulation, the initial stress was generated by the elastic model. To simplify the simulation, the state variable $\rho_e$ was zero at the initial stage (not considering the overconsolidation). Then, the artificial rainfall started, and when the soil became unsaturated, its behavior was described by the unsaturated model.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5.14 Discrete model for the dike failure due to rainfall infiltration.

Table 5.4 SPH parameters of the dike failure model test without the air effect.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density of water</td>
<td>1,000 kg/m³</td>
</tr>
<tr>
<td>Saturated saturation $S_u$</td>
<td>0.85</td>
</tr>
<tr>
<td>Density of soil</td>
<td>2,644 kg/m³</td>
</tr>
<tr>
<td>Residual saturation $S_r$</td>
<td>0.20</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.01 m</td>
</tr>
</tbody>
</table>

Table 5.5 Constitutive parameters of the dike failure model test without the air effect.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compression index $\lambda$</td>
<td>0.012</td>
</tr>
<tr>
<td>Swelling index $\kappa$</td>
<td>0.008</td>
</tr>
<tr>
<td>Critical state parameter $M$</td>
<td>1.10</td>
</tr>
<tr>
<td>Void ratio $N \ (p' = 98$ kPa on $N.C.L.)$</td>
<td>1.14</td>
</tr>
<tr>
<td>Poisson’s ratio $\nu$</td>
<td>0.25</td>
</tr>
<tr>
<td>Parameter of overconsolidation $a$</td>
<td>0.05</td>
</tr>
<tr>
<td>Parameter of suction $b$</td>
<td>0.05</td>
</tr>
<tr>
<td>Parameter of overconsolidation $\beta$</td>
<td>1.0</td>
</tr>
<tr>
<td>Void ratio $N_r \ (p' = 98$ kPa on $N.C.L.S.)$</td>
<td>1.28</td>
</tr>
</tbody>
</table>

Figs. 5.15 and 5.16 show the infiltration states for the low intensity and the high intensity, respectively. The test results were also presented to check the SPH simulations. It can be observed that the infiltrated areas from SPH simulations correspond well with the model test. The saturated zones develops from the surface of the dike, and the seepage line is parallel to the surface; this is called the surface infiltration, and is the typical characteristic of the shallow slope failure, which usually occurs in the fail process and the transit process from unsaturated state to saturated state of a slope under rainfall (Collins...
& Znidarcic 2004). As we can observe, the SPH proposed model yielded a good simulation of the surface infiltration.

![Fig. 5. 15 Infiltration process from the SPH simulation for the low intensity.](image1)

![Fig. 5. 16 Infiltration process from the SPH simulation for the high intensity.](image2)

Distributions of maximum shear strain are presented in Figs. 5.17 and 5.18 for the low intensity and the high intensity, respectively. The higher rainfall induces a larger value of maximum shear strain, and the maximum shear strain increases with the seepage time. It is also found that two types of shear strain are generated in the simulation. One was the shear strain zone, which appears on the surface of the dike and is caused by the pore water pressure and the suction reduction; the other one is the shear strain zone inside the dike, which is caused by the seepage force. The shear strain zone on the surface
indicated that the surface easily fails in the infiltration process.

Fig. 5. 17 Distribution of maximum shear strain from the SPH simulation for the low intensity.

Fig. 5. 18 Distribution of maximum shear strain from the SPH simulation for the high intensity.

The surface infiltration induced concentrated the pore water pressure around the dike surface. Then, the matric suction as well as the effective stress dropped according to the principle of effective stress. Therefore, the soil strength around the dike surface was greatly influenced by the suction reduction. The simulation of the dike failure could reproduce the surface deformation of shallow slope failure (Fig. 5.19) induced by the suction reduction.
5.4 The water-soil-coupled SPH simulation of slope failures due to the rainfall infiltration

A model test of the slope failure due to artificial rainfall, presented in Fig. 5.20, was conducted using the same test device as the dike failure test. There were two different bottoms in this model test: the left side was the undrained bottom, and the right side was the drained bottom. The drained bottom could let the air and water through, but the undrained bottom could not. The rainfall was set to four different intensities: 30 mm/hr., 60 mm/hr., 90 mm/hr., and 120 mm/hr.

A discrete model (see Fig. 5.21) was built according to the model test. There were totally 7,012 particles in the discrete model, among which 8 particles were used to define border lines, 4,892 particles were soil particles, and 2,112 particles were rain particles. As with the model test, two bottoms were considered: the left side was the undrained bottom, and the right side was the drained bottom. The SPH parameters used are shown in Table 5.6, and constitutive model parameters are presented in Table 5.7. Eq. (5.1) was
also used to describe the permeability change in the rainfall infiltration process.

![Fig. 5.21 The discrete model of slope and its initial configuration.]

**Table 5.6**: SPH parameters of the slope failure model test without the air effect.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density of water</td>
<td>1,000 kg/m$^3$</td>
</tr>
<tr>
<td>Saturated saturation $S_r$</td>
<td>0.85</td>
</tr>
<tr>
<td>Density of soil</td>
<td>2,644 kg/m$^3$</td>
</tr>
<tr>
<td>Residual saturation $S'_r$</td>
<td>0.20</td>
</tr>
<tr>
<td>Particle spacing</td>
<td>0.01 m</td>
</tr>
</tbody>
</table>

**Table 5.7**: Constitutive parameters of the slope failure model test without the air effect.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compression index $\lambda$</td>
<td>0.05</td>
</tr>
<tr>
<td>Swelling index $\kappa$</td>
<td>0.005</td>
</tr>
<tr>
<td>Critical state parameter $M$</td>
<td>1.10</td>
</tr>
<tr>
<td>Void ratio $N (p'\approx 98$ kPa on $N.C.L.)$</td>
<td>1.2</td>
</tr>
<tr>
<td>Poisson’s ratio $\nu$</td>
<td>0.25</td>
</tr>
<tr>
<td>Parameter of overconsolidation $a$</td>
<td>0.05</td>
</tr>
<tr>
<td>Parameter of suction $b$</td>
<td>0.05</td>
</tr>
<tr>
<td>Parameter of overconsolidation $\beta$</td>
<td>1.0</td>
</tr>
<tr>
<td>Void ratio $N_r (p'\approx 98$ kPa on $N.C.L.S.)$</td>
<td>1.28</td>
</tr>
</tbody>
</table>

At the first stage of the SPH simulation, the initial stress was generated by the elastic model. To simplify the simulation, the state variable $\rho_e$ was set to zero at the initial stage.
(not considering the overconsolidation). Subsequently, the artificial rainfall started, and when the soil became unsaturated, its behavior was described by the unsaturated constitutive model.

Fig. 5.22 shows the seepage state simulated at 120 min. It can be found that the infiltrated area of the high intensity is larger than that of the low intensity. For the low intensity, 30 mm/hr. and 60 mm/hr., there are still trapped dry areas inside the dike at 120 min from the model test and SPH simulations. But, for the high intensity, 90 mm/hr. and 120 mm/hr., slopes are almost fully saturated from the SPH simulations. Therefore, SPH simulations reflected the effect of rainfall intensity on the seepage process.

For the high intensity, reaching the stable volumetric water (the horizontal part of time history of the volumetric water content) content requires a shorter time. This phenomenon can be observed from the time histories of the volumetric water content from the model test and SPH simulations, which are shown in Fig. 5.23 and Fig. 5.24, respectively. The volumetric water content has a sharp increase, and then achieves a stable
volumetric water content both in the model test and SPH simulations. Table 5.8 is the time of reaching the stable volumetric water content for the sensor C5, from which the SPH results correspond well with the model test.

Fig. 5.23 Time histories of the volume water content under different rainfall intensities (the drained condition and top layer of slope).

Fig. 5.24 Time histories of the volumetric water content from the SPH simulations for the sensor C5.
Fig. 5.25 indicates the distribution of the maximum shear strain at 120 min obtained from SPH simulations, where a large value means large deformation of the soil phase. The maximum distribution mainly appears around the slope surface due to the increasing of the pore water pressure and the reduction of soil strength. For the low intensity, the shear strain zone does not form a slip surface. But for the high intensity, a slip surface is formed in the simulation. So the rainfall of a high intensity has great influence on the stability of the slope and dike.

Table 5.8 Time of reaching stable volume water content for Sensor C5 (unit: min)

<table>
<thead>
<tr>
<th></th>
<th>30 mm/hr</th>
<th>60 mm/hr</th>
<th>90 mm/hr</th>
<th>120 mm/hr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model test</td>
<td>72</td>
<td>65</td>
<td>45</td>
<td>18</td>
</tr>
<tr>
<td>SPH simulation</td>
<td>70</td>
<td>67</td>
<td>48</td>
<td>32</td>
</tr>
</tbody>
</table>

The infiltration process, volumetric water content, and maximum shear strain were obtained from SPH simulations, corresponding with the model test. Related results has been published in literatures (Zhang & Maeda 2014; Zhang & Maeda 2015).
5.5 The water-air-soil-coupled SPH simulation of slope failures due to the rainfall infiltration

The air phase was taken into account in this simulation to reveal the air behavior in the process of slope failures due to the rainfall infiltration.

The previous slope model was filled with air particles that had the same locations as soil particles. The air phase had an initial pressure of 100 kPa, and its initial density was 1.207 kg/m³. The area beside the particles (water, soil, and air) was set to a vacuum but had a pressure of 100 kPa. Therefore, the dynamic pressure was used for the calculation of acceleration, and equal suction (set to 1 kPa) at the initial stage. The other parameters were the same as parameters in Table 5.6 and Table 5.7. Eq. (5.1) was used to describe the permeability change in the infiltration process.

The infiltration of the SPH simulation with the air effect is presented in Fig. 5.26, for 30 mm/hr. (red particles are unsaturated particles), and Fig. 5.27 shows the time histories of the volumetric water content between the two-phase and three-phase SPH simulations for the sensors G5. It can be easily seen that the air phase decelerates the infiltration of rainfall in the slope and takes a longer time to achieve the stable volumetric water content than does the case without the air effect.

The SPH simulated air bubble rising is shown in Fig. 5.28 and the air blow from the model test is presented in Fig. 5.29. In Fig. 5.28, red particles are rain particles, green particles are air particles and blue particles are water particles. The air bubble rises with the time, and its outline can be observed from the SPH simulation. Meanwhile, air particles are mainly concentrated near the surface of slope where the air blow occurs. In addition, the interface between the water phase and the air phase is a fluctuant curve instead of a straight line, which is called the Rayleigh-Taylor instability in fluid dynamics (see Fig. 30). This is the instability of an interface between two fluids of different densities when one fluid is pushing another fluid (Sharp 1984), and would induce the irregular change of saturation around the interface. Submergence, which would cause extra positive dilatancy and collapse to the soil (Zhang & Ikariya 2011), can be induced in the irregular change process of saturation and affect the stability of slope due to heavy rainfall.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5. 26 Infiltration process from the SPH simulation with the air effect.

Fig. 5. 27 Time histories of volumetric water content between two-phase and three-phase SPH simulations for sensor G5.

Fig. 5. 28 Air blow and Rayleigh-Taylor instability from the SPH simulation.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

Fig. 5.29 Air blow from the model test.

Fig. 5.30 Rayleigh-Taylor instability.

Figs. 5.31 and 5.32 show time histories of the pore water pressure change and the pore air pressure change for the sensor G5, respectively. Changes of the water pressure and the air pressure both started from 40 min. The starting time of the volumetric water content differs from that of the water pressure and air pressure, because of the non-linear SWCC. Time history of the pore water pressure fluctuates before reaching the stable state, while the pore air pressure slightly increases after reaching the minimum value. At the final stage, the water pressure is almost same as the air pressure indicating that the soil around G5 was saturated.

As we can see, this SPH simulation presented the preliminary air behavior, such as the formation of air bubbles, and the irregular interface between the water phase and the air phase, therefore the proposed SPH method can provide new insights on the seepage failure of three-phase materials. Related results was published in literature (Zhang & Maeda 2015).
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

5.6 Summary

The application of proposed SPH method was extended to three seepage failure tests, dike failure for piping, dike failure by heavy rainfall, and slope failure by heavy rainfall. The SPH simulations reproduced the surface infiltration of the shallow slope failure, the deformation process, the air blow, the suction reduction, and the Rayleigh–Taylor instability, corresponding well with model tests.
Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall

References


Chapter 5 Application of the proposed SPH method to seepage failures due to heavy rainfall
Chapter 6

Conclusions and remarks

6.1 Conclusions

For seepage failures due to heavy rainfall, this work conducted the study on the modeling of the water-air-soil three-phase material using the SPH method, the verifying and validating of the proposed SPH method, and the application to seepage failure model tests. Some conclusions can be obtained as,

1. From the summary of the current research, there are some important factors that should be considered in the analysis of seepage failures, such as the water-air-soil coupling, the unsaturated behavior, the large deformation, the surface infiltration, and so on.

2. The Smoothed Particle Hydrodynamics (SPH) method, has a unique advantage on the simulation of free surfaces, deformation boundaries and large deformations, and therefore is suitable for the analysis of seepage failures due to heavy rainfall. However, to date, few studies regarding SPH simulations of seepage failure under heavy rainfall that consider the water-soil-air coupling, the large deformation, the surface infiltration, and the suction reduction have been presented in the literature.

3. In the proposed water-air-soil three-phase SPH model, fluids and solid are simulated on different layers. The three-phase field theory was introduced and related quantities were derived. Then, governing equations including the continuity equation and the momentum equation were transformed to the SPH formulas for water, air, and soil, respectively. The equation of state was used to calculate the dynamic pressure of the water phase and the air phase. After that,
we introduced the unsaturated constitutive model to describe the behavior of soil phase. Meanwhile, the proposed model used the frictional forces, with due consideration of the effects of the porosity and the coefficient of permeability, as interaction forces between different phases.

(4) The B-spline function was selected as smoothing kernel function to obtain high accuracy and efficiency. Based on the Linked-list searching method, a highly effective searching method was proposed and applied in the proposed SPH programs. Meanwhile, critical section in OpenMP API and atomicAdd in CUDA solved the incoherent issue of parallel optimization. After that, using the C++ programming language, highly parallelized SPH programs were written, called ZM-OP (OpenMP) SPH program and ZM-CD (CUDA) SPH program.

(5) From the simulations of a dam break, a triaxial compression test and a water-air coupled case, the proposed SPH method can make a good simulation of the water phase, the air phase, the soil phase, and furthermore the effect of the suction reduction to the soil strength. Through the simulations of a falling head permeability test and a flowing test of liquefied soil, the coupling of water and soil was verified and validated. After that, the proposed SPH method was proved to be highly effective numerical method by a three-dimension dam break.

(6) The application of proposed SPH method was extended to three seepage failure tests, dike failure for piping, dike failure by heavy rainfall, and slope failure by heavy rainfall. The SPH simulations reproduced the surface infiltration of the shallow slope failure, the deformation process, the air blow, the suction reduction, and the Rayleigh–Taylor instability, corresponding well with model tests.

(7) Although the results from SPH simulations are preliminary, the proposed SPH method provided new insights and is a novel numerical tool for the analysis of seepage failures due to heavy rainfall.

6.2 Remarks on the future study

(1) In this thesis, only seepage failure tests were simulated by the proposed SPH method. Thus, in the future, the real case of seepage failures due to heavy rainfall
can be simulated for the disaster assessment.

(2) The application of the proposed SPH method can be extended to other geologic hazards, such as landslides, the large deformation of liquefied soils, MSW landfill slides, and so on.

(3) As the development of geomechanics, more advanced soil model can be introduced into the SPH method.

(4) It is also a hot topic to combine SPH method with the traditional numerical methods. For example, SPH method and DEM method together simulate the seepage in the material of different sizes; the run-out of flow-like slides and its impact on the structure can be analyzed by the SPH and FEM; another combination of SPH and FEM is in the level of algorithm, such as the smoothed finite element method, the Edge-based smoothed finite element method, the Face-based smoothed finite element method, and so on.
Chapter 6 Conclusions and remarks
Appendix

A-1 Source of ZM-OP SPH program and ZM-CD SPH program

If you would like to get the source of the ZM-OP SPH program and the ZM-CD SPH program, please contact the following email:

zhangzmnn2014@outlook.com.

A-2 Compiling of the ZM-OP SPH program and the ZM-CD SPH program

- The compiling of the ZM-OP SPH program
  The compiler must support OpenMP. The following compilers are recommended:
  Visual Studio 2010 Professional or superior
  GCC (GNU Compiler Collection)

  In the case of Visual Studio 2010 Professional or superior, the method of enabling OpenMP support is presented in,

  In the case of GCC, the compiling flag –fopenmp is needed.

- The compiling of the ZM-CD SPH program
  Following software is required before compiling.
  Visual Studio 2010 Professional or superior
A-3 Input file

Input file type:
0—two input files (DATA.dat and PARTI.dat);
1—single input file (input.dat).

DATA.dat

<table>
<thead>
<tr>
<th>ndim</th>
<th>bdtype</th>
</tr>
</thead>
<tbody>
<tr>
<td>fop</td>
<td>inip</td>
</tr>
<tr>
<td>sltype</td>
<td>dttype</td>
</tr>
<tr>
<td>ntotal</td>
<td>rtype</td>
</tr>
<tr>
<td>intertype</td>
<td>interc</td>
</tr>
<tr>
<td>per[0]</td>
<td>per[1]</td>
</tr>
</tbody>
</table>

Line 1:
ndim: dimensions of problem (2 or 3).
bdtype: boundary type.
   1—a simple retaining boundary; 2—Morris and Takeda boundary.

Line 2:
fop: output of particle information to file every fop steps.
inip: steps for the generating of initial stress with elastic model.
loop: total steps of simulation.

Line 3:
sltype— the type of smooth length.
   1— constant and 2— variable smooth length.
dttype— the type of time step interval.
   1— constant and 2— variable time step interval.
dr: m, initial spacing of particles.
dt: s, initial time step.

Line 4:
ntotal: total number of particles.
rtype: rainfall type.
   1— 30mm/hr; 2— 60mm/hr; 3— 90mm/hr; 4— 120mm/hr; 5— water level up
   without retaining; 6— dike low rainfall intensity; 7— dike high rainfall
   intensity; 8— water level up with retaining.
nrain: number of rainfall particles.
nhors: number of rainfall particles at first stage.
   For rain type 8: it depends on the simulation case;
   For other rain types: it equals to the nrain.
nhore: number of rainfall particles at second stage
   For rain type 8: it depends on the simulation case;
   For other rain types: it equals to 0.

Line 5:
Intertype: the type of interaction force.
   1— Biot, 1941 and Prevost, 1979; 2— Biot, 1956.
interc: controlling the magnitude of interaction force.
   intertype=2 interc has no meaning.

Line 6:
per[0]: m/s, the permeability at saturated saturation.
per[1]: m/s, the permeability at residual saturation.
Permttype: permeability calculation type.
1—linear relationship and 2—three layers with different permeability.

Line 7:
modeltype[1]: constitutive model type for matype 1.
modeltype[2]: constitutive model type for matype 2.
modeltype[3]: constitutive model type for matype 3.
   1—elastic model; 2—drucker-prager model; 3—unsaturated subloading cam-clay model and 4—elastic solid model (high Young’s modulus).

Line 8:
apa: Pa, the initial pressure of air phase.

Line 9: controlling parameters for boundary effect.
bdef[0]: used in subroutine noperboundarypost; bdef[1] and bdef[2]: used in subroutine tmboundary.

Part 2

<table>
<thead>
<tr>
<th>e0</th>
<th>poi</th>
<th>eps0</th>
<th>zmf</th>
<th>zramda</th>
<th>zkapa</th>
<th>ann</th>
<th>bnn</th>
<th>beta</th>
<th>sme0</th>
</tr>
</thead>
<tbody>
<tr>
<td>sitas</td>
<td>sitar</td>
<td>epsr</td>
<td>suct0</td>
<td>sd</td>
<td>sw</td>
<td>kse</td>
<td>c1</td>
<td>c2</td>
<td>c3</td>
</tr>
</tbody>
</table>

Line 10:
e0: Pa, the elastic module only for the element test; otherwise it is 0.0.
poi: ν, Poisson’s ratio.
eps0: the void ratio at $p_0=98\text{kPa}$ (saturated).
zmf: $R_0=\sigma_1/\sigma_3$, Stress ratio at critical state.
zramda: $\lambda$, compression index.
zkapa: $\kappa$, swelling index.
ann: $a$, controlling the evolution of overconsolidation.
bnn: $b$, controlling the evolution of overconsolidation.
beta: $\beta$, controlling the evolution of overconsolidation.
sme0: Pa, confining pressure only for the element test and tri axial test.

Line 11:
sitas: \( S^r \), saturated saturation.
sitar: \( S^r \), residual saturation.
epsr: \( N_r \), the void ratio at \( p_0=98\text{kPa} \) (residual saturation).
suct0: \( \text{pa} \), initial suction only for the element test and tri axial test.

Line 12:
- sd: drying AEV parameter.
- sw: wetting AEV parameter.
- kse: slope of scanning curve.
- c1: scaling factor of drying process.
- c2: scaling factor of wetting process.
- c3: scaling factor of scanning curve.

PARTI.dat

2D:

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>vx</th>
<th>vy</th>
<th>type</th>
<th>matype</th>
</tr>
</thead>
</table>

Column 1: \( m, x \).
Column 2: \( m, y \).
Column 3: \( \text{m/s}, vx \). Initial velocity.
Column 4: \( \text{m/s}, vy \). Initial velocity.
Column 5: type. Major material type.

- 0—boundary particles; 1—water particles; 2—soil particles; 3—air particles;
- 7—rain particles

Column 6: matype. Minor material type only for soil (major type 2) to simulate layers of different permeability.

3D:

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
<th>vx</th>
<th>vy</th>
<th>vz</th>
<th>type</th>
<th>matype</th>
</tr>
</thead>
</table>

Column 1: \( m, x \).
Column 2: \( m, y \).
Column 3: \( m, z \).
Column 4: \( \text{m/s}, vx \). Initial velocity.
Column 5: \( \text{m/s}, vy \). Initial velocity.
Column 6: \( \text{m/s}, vz \). Initial velocity.
Appendix

Column 7: type. Major material type.
0—boundary particles; 1—water particles; 2—soil particles; 3—air particles;
7—rain particles

Column 8: matype. Minor material type only for soil (major type 2) to simulate layers of different permeability.

➢ input.dat

It is only the combination of the DATA.dat and PARTI.dat.

A-4 Running the SPH programs

The simulation type:
0—start new calculation;
1—continue old calculation.

Input file type:
0—two input files (DATA.dat and PARTI.dat);
1—single input file (input.dat).

The number of CPU cores used (only for the ZM-OP SPH):
Depending on the computer configuration and the simulation case.

A-5 Output file

➢ inp file

inp file can be viewed in the Micro AVS.

For air (air******.inp) and water (water******.inp), quantities are:

<table>
<thead>
<tr>
<th>vx</th>
<th>Velocity at x m/s.</th>
<th>sigyy</th>
<th>$\sigma_{yy}$ Pa.</th>
</tr>
</thead>
<tbody>
<tr>
<td>vy</td>
<td>Velocity at y m/s.</td>
<td>sigzz</td>
<td>$\sigma_{zz}$ Pa.</td>
</tr>
<tr>
<td>vz</td>
<td>Velocity at z m/s.</td>
<td>ccx</td>
<td>Color function at x of 1st order.</td>
</tr>
<tr>
<td>pre</td>
<td>Pressure Pa.</td>
<td>ccy</td>
<td>Color function at y of 1st order.</td>
</tr>
<tr>
<td>sigxx</td>
<td>$\sigma_{xx}$ Pa.</td>
<td>cs</td>
<td>Color function.</td>
</tr>
</tbody>
</table>
For soil (soil******.inp), quantities are:

<table>
<thead>
<tr>
<th>vx</th>
<th>Velocity at x m/s.</th>
<th>syy</th>
<th>σ_{yy} Pa.</th>
</tr>
</thead>
<tbody>
<tr>
<td>vy</td>
<td>Velocity at y m/s.</td>
<td>szz</td>
<td>σ_{zz} Pa.</td>
</tr>
<tr>
<td>vz</td>
<td>Velocity at z m/s.</td>
<td>sxy</td>
<td>σ_{xy} Pa.</td>
</tr>
<tr>
<td>suc</td>
<td>Suction Pa</td>
<td>mtxy</td>
<td>Maximum shear stress Pa.</td>
</tr>
<tr>
<td>exx</td>
<td>γ_{xx}</td>
<td>szx</td>
<td>σ_{zx} Pa.</td>
</tr>
<tr>
<td>eyy</td>
<td>γ_{yy}</td>
<td>veps</td>
<td>Volumetric strain</td>
</tr>
<tr>
<td>ezz</td>
<td>γ_{zz}</td>
<td>vsig</td>
<td>Mean stress Pa.</td>
</tr>
<tr>
<td>exy</td>
<td>γ_{xy}</td>
<td>q</td>
<td>Deviator stress Pa.</td>
</tr>
<tr>
<td>pre</td>
<td>Water pressure for soil Pa.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prew</td>
<td>Air pressure for soil Pa.</td>
<td>satu</td>
<td>Saturation.</td>
</tr>
<tr>
<td>sxx</td>
<td>σ_{xx} Pa.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For total (step******.inp), quantities are:

<table>
<thead>
<tr>
<th>vx</th>
<th>Velocity at x m/s.</th>
<th>syy</th>
<th>σ_{yy} Pa.</th>
</tr>
</thead>
<tbody>
<tr>
<td>vy</td>
<td>Velocity at y m/s.</td>
<td>szz</td>
<td>σ_{zz} Pa.</td>
</tr>
<tr>
<td>vz</td>
<td>Velocity at z m/s.</td>
<td>sxy</td>
<td>σ_{xy} Pa.</td>
</tr>
<tr>
<td>pre</td>
<td>Pressure Pa</td>
<td>syz</td>
<td>σ_{yz} Pa.</td>
</tr>
<tr>
<td>exx</td>
<td>γ_{xx}</td>
<td>szx</td>
<td>σ_{zx} Pa.</td>
</tr>
<tr>
<td>eyy</td>
<td>γ_{yy}</td>
<td>veps</td>
<td>Volumetric strain</td>
</tr>
<tr>
<td>ezz</td>
<td>γ_{zz}</td>
<td>vsig</td>
<td>Mean stress Pa.</td>
</tr>
<tr>
<td>exy</td>
<td>γ_{xy}</td>
<td>q</td>
<td>Deviator stress Pa.</td>
</tr>
<tr>
<td>eyz</td>
<td>γ_{yz}</td>
<td>type</td>
<td>Material type.</td>
</tr>
<tr>
<td>ezx</td>
<td>γ_{zx}</td>
<td>satu</td>
<td>Saturation.</td>
</tr>
<tr>
<td>sxx</td>
<td>σ_{xx} Pa.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- mgf file

  mgf file can be viewed in the Micro AVS as animation.
Appendix

- conout file

  The *contout* file is for continuing calculation and generated by the program, thus it should not be modified.