Nazarbayev University

Numerical modelling of multiphase flow in poorly consolidated sandstone reservoirs

By

Furkhat Khamitov

A thesis submitted to the School of Engineering and Digital Sciences in partial fulfilment of the requirements for the degree of

Doctor of Philosophy
in
Science, Engineering and Technology

Nur-Sultan, 2022

Declaration

I declare that the research contained in this thesis, unless otherwise formally indicated within the text, is the original work of the author. The thesis has not been previously submitted to this or any other university for a degree and does not incorporate any material already submitted for the degree.

Signed: 

Dated: 09.03.2022

Abstract

The oil and gas production from reservoirs of weakly-consolidated or non-consolidated strata is often accompanied by sand production. The sand production phenomenon is encountered in many oil fields of Kazakhstan, such as Karazhanbas, North Buzachi, Kalamkas, Zhalgiztobe, and Kenkiyak. It can dramatically affect production flowrate, damage downhole and surface equipment, raise critical failure probability. Understanding and predicting the sand production process are one of the main challenges for safe and profitable hydrocarbon production in Kazakhstan and worldwide.

The technical and operating procedures as drilling, perforations and hydrocarbon production, the stresses around the rock materials increase and fulfil the failure criterion and material can change from elastic to the plastic state. The plastic zone is an important part of the sand production models and is still not understood. In this research, the plastic zone properties during the perforation process of the material were investigated by the Discrete Element Method (DEM) approach. The contact model for weak sandstone developed by (Rakhimzhanova et al., 2018) was used to mimic cement material behaviour. The perforation process created a damage zone of about $3r$ to $4r$ thickness of the unbonded particles with nearly complete bond breakage in this zone, where r is the radius of the perforation tunnel. A compacted lower porosity zona $2r$ to $4r$ (minimal porosity is 30.5%) was embedded inside the perforation damage zone.

The reservoir and fluid properties can significantly impact sand production. The effects of fluid type and fluid-phase composition on sand production was investigated using coupled Computational Fluid Dynamics and Discrete Element Method (CFD–DEM) approach. The coupled CFD-DEM multi-scale models, where fluid is considered single-phase or multi-phase, are implemented to investigate the bond breakage behaviour during sand production. It was found that very different microstructural changes for light and heavy oils inside the sandstone samples. Light oil facilitated more significant particle movement and porosity change within the damage zone around the central perforation tunnel. The heavy oil outflow mobilized a more significant proportion of the sample's particles and would lead to more severe sand production.

The CFD-DEM simulations with Karazhanbas oil field properties (reservoir pressure $4.5\ MPa$ and overburden stress $5.9\ MPa$) were conducted. It was found that sand production with heavy oil provides a high bond breakage rate, and this behaviour is changed only after gas/water breakthrough. The fingering effect and formation of new channels due to injection of more mobile fluid were observed.

The CFD-DEM models have been compared qualitatively with the semi-analytical model results and laboratory data.

Acknowledgments

I would like to thank and express my gratitude and appreciation to my first Supervisor at Nazarbayev University, Professor Nguyen Hop Minh, who kept supporting and working with me as an external Supervisor throughout the whole period of my PhD work. Professor Nguyen got me acquainted with the DEM and introduced me to the international research community.

Also, let me express my special thankfulness to my Lead supervisor Professor Michael Yong Zhao, for continuous involvement and sharing his greatest experience in CFD modelling, who helped me find a newly-glanced approach in multiphase flow modelling.

Thanks to my Co-Supervisor, Professor Sergey Spotar, for his support in revealing methodology weaknesses and finding new opportunities in CFD-DEM modelling usage in sand production issues.

Thanks to my advisors for their brilliant ideas, patience, and hard work within these six years of working together. My personal finding and resume are that there are no impossible things in scientific research, even in the so-called "tough" and difficult situations. There is always a way to figure out the right solution and an elegant path to find something new.

I would like to give special thanks to our "dream" team with Dr. Ainash Shabdirova and Dr. Ashirgul Kozhagulova for their kind support and always being ready to help in my work.

Also, my appreciation to Dr. Yerlan Amanbek, who was Principal Investigator of the research grant, No AP08052762, from the Ministry of Education and Science of the Republic of Kazakhstan. With his contribution, I grew to the purely modern CFD-DEM modelling level.

Thanks to Professor Luis Ramón Rojas-Solórzano for his individual approach to each PhD student in the School of Engineering and Digital Sciences and support in general and personal issues.

Thanks to my colleagues in the Oil & Gas industry, experts Dr. Akzhan Kassenov and Dr. Andrey Sveshnikov. Their vast experience, scientific knowledge and continuous support allowed me to join up Science and actual engineering challenges in the Oil & Gas industry.

My endless gratitude to my parents, Ablakhat Khamitov and Aitkul Khamitova, for their support and motivation for being engaged in scientific research.

I would like to thank my wife Ainura and daughter Deya for their positive energy and belief in me within the whole not easy path of Science.

Scientific Contributions

Journal papers:

Khamitov F, Minh N.H, Zhao Y. (2021). "Coupled CFD–DEM numerical modelling of perforation damage and sand production in a weak sandstone formation". Geomechanics for Energy and the Environment, December 2021, 100255. https://doi.org/10.1016/j.gete.2021.100255

Khamitov F., Minh N.H., Zhao Y. (2022). "Numerical investigation of sand production mechanisms in weak sandstone formations with various reservoir fluids". Submitted to the *International Journal of Rock Mechanics and Mining Sciences.* Status: *accepted for publication.*

Shabdirova A., Khamitov F., Kozhagulova A., Amanbek Y. (2021). Sand production characteristics in two-phase flow experiments. Submitted to the *Journal of Petroleum Science and Engineering*. Status: *Under review.*

Conference Proceedings:

Shabdirova A., Khamitov F., Kozhagulova A., Amanbek Y., Minh N.H., Zhao Y. "Experimental and numerical investigation of the plastic zone permeability". Published ARMA 2020.

Khamitov F., Minh N.H., Zhao Y., Fok S.C. (2019). "Implementation of JKR Contact Model in LIGGGHTS" proceeding for "8th International Conference on Discrete Element Methods" July 21-26 (2019), Netherlands.

Rakhimzhanova A., Khamitov F., Minh N.H. & Thornton N.H. (2018). "3D DEM simulation of triaxial compression tests of cemented sandstone" proceeding of IS Atlanta 2018 Symposium on Geomechanics from Micro to Macro in Research and practice, Atlanta

Contents

List of Figures

CHAPTER 1 - INTRODUCTION

This Chapter describes the research topic, novelties, hypothesis, and objectives of the work.

1.1 Sand production phenomenon

70% of the world's oil and gas reserves are found within reservoirs of weakly-consolidated or non-consolidated strata. The petroleum industry spends huge amounts of money each year to prevent and control sand production (Ranjith et al., 2013). Material degradation is a critical process leading to sanding. Drilling operations, cyclic effects of shut-in and start-up, operational conditions, reservoir pressure depletion, and strength-weakening effect of water may gradually lead to sandstone degradation around the perforations and boreholes. A high-pressure gradient due to fluid flow also facilitates the detachment of sand particles. In addition, fluid flow is responsible for the transport and production of cohesionless sand particles or detached sand clumps to the wellbore (Rahmati et al., 2013).

1.2 Numerical methods to study sand production

Sand production problems are investigated using physical model testing, analytical and empirical relationships, and numerical models. These models help to understand the main mechanisms and describe the sand production behavior during oil and gas production. Over the past decades, numerical models have become the most powerful tools for predicting sand production due to a significant increase in computing power. They can be combined with analytical correlations to obtain the results more efficiently. Experimental results are also utilized to calibrate or validate the numerical model.

1.3 CFD-DEM approach for sand production

Cundall and Strack (1979) was introduced the Discrete Element Method (DEM) to investigate the microscopic behaviour of granular materials in microscale. Particles interact with each other as they are subjected to the boundary conditions. DEM is widely used to model soils and rocks, including sandstone. The micromechanical parameters (particles shape, size distribution, contact model) of the DEM approach cannot be easily obtained from experimental tests and should be calibrated depending on the contact model(Coetzee, 2017; Hanley et al., 2011; O'Sullivan, 2011; Rahmati et al., 2013).

The hydrocarbon reservoir rock is usually saturated by fluid (oil and gas) in DEM conditions, which means that the particles are surrounded by fluid. Several fluid-coupling techniques are used to describe the interaction of liquid and particles (Zhu et al., 2007). The sand production phenomena using the CFD-DEM approach was initially introduced by Tsuji et al. (1993) to simulate the formation

of bubbles in gas-fluidized granular beds. Particles in microscale are combined with fluid in mesoscale.

1.4 Problem Statement

Sand production is the cause of many subsequent problems in the Kazakhstan oil and gas industry. Oil and gas production with sand materials is present in many oil fields in Kazakhstan, such as Karazhanbas, North Buzachi, Kalamkas Zhalgiztobe, and Kenkiyak ("Kazakhstan Upstream Oil and Gas Technology and R&d Roadmap," 2013)**.** It can dramatically affect the well performance, damage downhole and surface equipment, raise the capability of critical failure (Acock et al., 2004). On the other hand, when sand is extracted from the reservoir, the porosity close to the well increases, a zone of increased permeability is formed, which expands, providing a greater flow of liquid into the wellbore (B. Dusseault & El-Sayed, 1999; Geilikman & Dusseault, 1997; Tremblay et al., 1996). Therefore, understanding the main features of the mechanisms of sand production, the ability to predict the production dynamics of both hydrocarbons and sand materials become key to safe and profitable production.

The technical and operating procedures as drilling, perforations and hydrocarbon production, the stresses around the rock materials increase and fulfil the failure criterion and material can change from elastic to the plastic state. Bratli and Risnes (1981) detected a plastic zone of the failed materials around a sand arch after a shear failure. First, the in-situ stresses near the sand arch are calculated. The most significant difference between the stresses was found to occur at the inner surface, and this region fails first. The material behaves elastically up to the limit given by the Mohr-Coulomb (Labuz & Zang, 2012) failure criterion. The study extended through investigating of the stresses around the open borehole and cased-hole completions by Risnes et al. (1982b). Following these studies, the formation of the plastic zone had been reported in several experimental and numerical studies(Pierre Cerasi et al., 2005; Goshtasbi et al., 2013; Tovar, 2007; Tronvoll & Fjær, 1994; H. Wang et al., 2019), The plastic zone permeability study limited and various approach; it could be a varying value across the plastic zone in Bratli and Risnes (1981), or a single value of one-tenth of the intact zone permeability in Risnes et al. (1982b). The plastic zone is an important part of the sand production models and is still not understood. In this research, the plastic zone properties of the material are studied during the perforation process.

The reservoir and fluid properties can significantly impact sand production(Veeken et al., 1991). The laboratory experiments show that cumulative sand volume and decline period for heavy oil wells can differ in hundred times than for light oil wells. The displacing fluids indicated sand production initiation and final sand mass (Ahmad & Miskimins, 2019; Al-Awad et al., 1999; Harris

et al., 2005; Hughes et al., 2008; B Wu et al., 2016). However, the mechanisms of fluid type influence were not addressed in previous studies. Therefore, the second problem statement can be stated as studying the effects of fluid type on sand production behavior at the microscale.

Highly permeable and porous sandstones characterize Kazakhstan oil fields with weak sandstone reservoirs with initial flooding (usually above 20%), low depth, viscous oil, etc. Newly drilled wells have an initial water cut that increases during the well operations' period (Abishev et al., 2018; Collins et al., 2008; Nie et al., 2010). The conventional oil and gas reservoirs are usually presented by multiphase fluid. The effect of fluid phase composition on sand production has been investigated in many experimental and numerical studies(Crowe et al., 2011; Kozhagulova et al., 2021; E Papamichos et al., 2010; N. A. Patankar & Joseph, 2001; Wan et al., 2004). The researchers have found that the sandrate increases with an increase in water cut. Veeken *et al.* (1991) detected that the sand production occurs in well with an average water cut of 19% with no-sand production wells with 2% respectively. The effect of water cut is to reduce perforation strength and promote sand production was reported by Bailin Wu et al. (2005). At the same time, the other authors observed the opposite effect: sand production decreased with increasing water cut (Collins et al., 2008; Tananykhin et al., 2021). A typical well with sand production (Ustyurt-Buzachi sedimentary basin, Kazakhstan) is illustrated in Figure 1.1. The sandrate is reducing from the initial high to the small constant values, while the watercut is only increased. This the opposite conclusions in the literature and the Kazakhstan oil fields are the primary motivation to the current research.

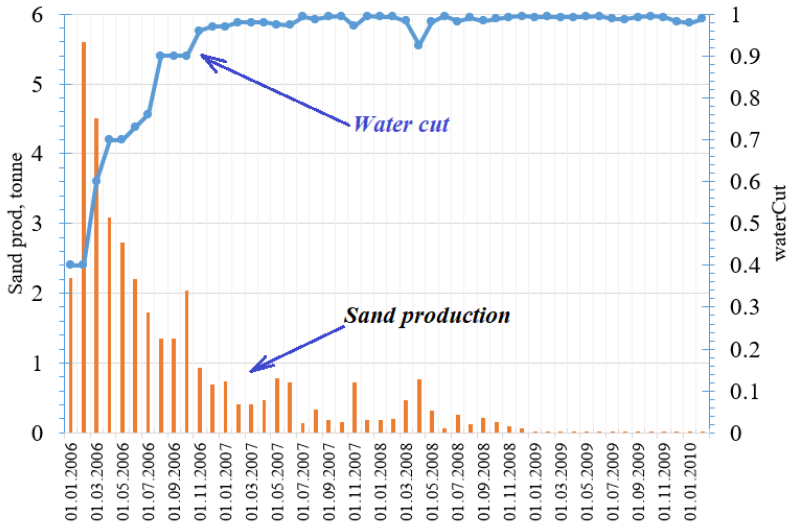


Figure 1.1. The sand production history of *Nwell* in Ustyurt-Buzachi

Experimental and analytical models of the sand production problem still play an essential role in finding a solution for the sand production problems. Numerical modelling has become the most

powerful tool for predicting sand production (Rahmati et al., 2013). This can be done by incorporating the physics of fluid and solid interactions at the microscale to understand better the disaggregation of particles from the rock mass and its transport under the effect of fluid flow.

The thesis focused on improving the previous DEM, CFD-DEM numerical studies (Rakhimzhanova et al., 2018-2020; Wu et al., 2016) in terms of a more comparable cylindrical geometry, a larger simulation scale, different flow directions and the multiphase flow simulation, as well as the detailed scientific visualization similar to Climent *et al.* (2014) to provide greater insight into the complex sanding processes. The contact model for weak cement sandstone (Rakhimzhanova et al., 2018) is coupled with one, two-phase fluid flow and the application of fluid flow-solid coupling analysis on sand production problem. Four-way coupling methods are encountered in this thesis: interactions between solids and fluid phases, particle-particle collisions, and interaction in fluid phases (Norouzi et al., 2016). The developed models and methods are then validated by experimental and analytical results and employed to provide greater insight into the mechanisms of sand production, which cannot be obtained easily by experimental and analytical methods. The "bottom-up" approach is used in this study. Sand production processes are modelled and investigated microscale to explain macroscopic outcomes.

Therefore, the primary motivation behind this study was to investigate the sand production at microscale associated with elastoplastic material properties, different single and multiphase reservoir fluid effects.

1.5 Research Hypothesis

The literature review shows that most numerical and analytical approaches investigate the sand production phenomena using continue approaches at the macroscale(Norouzi et al., 2016; Nouri et al., 2006; E. Papamichos et al., 2001; Taghilou & Rahimian, 2014; Wan et al., 2004; Y. Wang & Xue, 2002). The material continuity means that the material cannot be separated into even smaller particles. This assumption of continuity may not consider the rock and fluid properties in reservoirs with high fidelity and usually cannot simulate and reveal physical processes at the microscale. The phenomena of plastic zone generation in the weak cement sandstone, the erosion and transportation of sand materials in porous media saturated with multiphase fluid at microscale are still not well studied, and therefore the following hypothesis and research objectives arise.

Hypothesis:

1) Perforation process causes the creation of the plastic zone in the weak cemented material.
2) The type of reservoir fluid affect sand production pattern.

3) The multiphase properties of fluid flow in numerical models play a significant role in simulating sand production processes in Kazakhstan oil fields.

1.6 Research Objectives

1) Improve numerical CFD-DEM study of weak cement sandstone by employing advanced scientific visualization and parallel computation.
2) Develop the weak cement contact model in a $3D$ numerical multiscale multiphase model with irregular boundary (CFD-DEM) for parallel computation.
3) Investigate the mechanical behavior of weak sandstones during the perforation and sand production process.
4) Examine the mechanisms of the sand production phenomenon with different reservoir fluids.
5) Investigate the key mechanisms of sand production with multiphase fluid flow under laboratory experiments and Kazakhstan oil reservoir conditions with realistic overburden stresses and fluid flow conditions.

1.7 Key novelties

The key novelties of the current research can be summarized as follows:

1) Implementation of the weak cement contact model in CFD-DEM model with multiphase fluid flow.
2) The investigation of perforation and sand production processes with complex geometry using advanced scientific visualization and advanced post processing.
3) Identifying the elastoplastic zones and investigating their behaviors during perforation.
4) Defining of optimal CFD-DEM model coupling time.
5) Numerical simulation of the sand production processes in realistic Kazakhstan oil field conditions using multiphase fluid CFD-DEM model and uncovering the fluid fingering and channeling phenomena.

1.8 Thesis Organization

There are eight chapters in the thesis.

The current chapter is the introduction to the contents of the thesis. It describes the sand production phenomenon in Kazakhstan, the methods and approaches applicable to the study, and

justifies the necessity of building a new CFD-DEM models with multiphase fluid. The chapter also states the aim and objectives of the research and describes the contributions to the scientific literature.

The second chapter presents fundamental material which is relevant to the research. In this chapter the main mechanisms of sand production and sand prediction methods are presented. The detailed literature review of numerical approach for sand production with their limits and features is given.

Chapter three presents the Mathematical Formulations and Numerical methodology by describing CFD-DEM for one-phase and two-phase fluid models. The formulation of DEM for the weak cement formation is also described.

Chapter four presents the numerical model setup and boundary conditions for the CFD-DEM system. The four different model setups were constructed in order to mimic laboratory experiments and Kazakhstan oilfield reservoir conditions.

In Chapter five, a coupled CFD-DEM simulation method was implemented to study the bond breakage behaviour during perforation and sand production processes in a poorly consolidated sandstone using a HPC system.

In Chapter six, a three-dimensional CFD-DEM simulation method was developed to examine the influence of different reservoir fluids on sand production. The fluid properties and reservoirs were prescribed and modelled after the conditions in the Kenkiyak and Uzen oil fields in Kazakhstan.

In Chapter seven, the CFD-DEM model was developed to investigate the sanding mechanisms with two-phase fluid flow, where the DEM model was created similarly as in Chapter 5. Two different DEM samples with overburden stresses $3MPa$ and $5.9MPa$ were used to mimic laboratory experiments and reservoir development.

Finally, conclusions, limitation of the study and future work are discussed in Chapter eight.

CHAPTER 2 - LITERATURE REVIEW

2.1 Mechanisms of sand production

The hydrocarbon production from the weak sandstone reservoirs is typically accompanied by sand production, which is a reason of breakage of underground equipment, wiping the tubing and surface tubing, clogging of perforation intervals, curvature of the borehole, increasing the cost of cleanup operations and more (Figure 2.1).

a) b)

Figure 2.1. Sand production problems in the oil field. a) Surface pipeline clogging; b) Clogging of tubing

Sand production can be prevented before well production by the installation of downhole sand control systems such as gravel packs, sand screens and chemical consolidations, which could inhibit the movement of sands into the underground and surface facilities (Tronvoll et al., 2001). If no sand exclusion is used to stop the sand production, the conventional sand management approach is to reduce the hydrocarbon production rate to minimize the produced sands, and this can significantly reduce wellbore productivity. On the other hand, when sands are extracted from reservoir, it increases the porosity close to the wellbore. The zone of increased permeability provides greater flow of liquid into the wellbore (B. Dusseault & El-Sayed, 1999; Geilikman & Dusseault, 1997; Tremblay et al., 1996). Understanding the mechanisms of sand production improves the ability to predict the production dynamics of both hydrocarbons and sand materials, which are key to safe and profitable operation.

Veeken et al. (1991) classified sand production into three categories: transient sand production, continuous sand production and catastrophic sand production. Decreasing the sand concentration under the constant production rate conditions is considered transient sand production. The constant sandrate behavior is classified as continuous sand production. Catastrophic sand production refers to the case where a high-rate sand influx causes the well suddenly choke and/or die. The parameters influencing sand production can also be grouped into three categories (Table 2-1): the physical properties of the reservoir formation and the fluid, the well installation and completion, and the type of oil and gas field development. The permeability in Table 2-1 can be determined from reservoir rock porosity using the Kozeny- Carman equation (Carman, 1939; Kozeny, 1927). The sand control strategies depending on porosity values were discussed by Ben Mahmud et al. (2020).

Table 2-1. Parameters influencing sand production (Veeken et al., 1991)

FORMATION	
Rock	• Strength • Vertical and horizontal in-situ stresses (change during depletion) • Depth (influences strength, stresses and pressures)
Reservoir	• Far field pore pressure (changes during depletion) • Permeability • Fluid composition (gas, oil, water) • Drainage radius • Reservoir thickness • Heterogeneity
COMPLETION	
• Wellbore orientation, wellbore diameter • Completion type (open hole/perforated) • Perforation methods (height, size, density, phasing, under/overbalance) • Sand control (screen, gravel pack, chemical consolidation) • Completion fluids, stimulation (acid volume, acid type) • Size of tubulars	
PRODUCTION	
• Flow rate • Drawdown pressure • Flow velocity • Damage (skin) • Bean-up/shut-in policy • Artificial lift technique • Depletion • Water/gas coning • Cumulative sand volume	

Sand production is a coupled hydro-mechanical process (Figure 2.2) which involves the following steps: near wellbore rock failures due to applied stresses during drilling and perforation operations and afterwards sands are produced and transported due to fluid drag force because fluid flow removes or erodes the failed rock. The hydrodynamic forces are assumed as too weak to erode the intact rock (Kim, 2012; Euripides Papamichos, 2006).

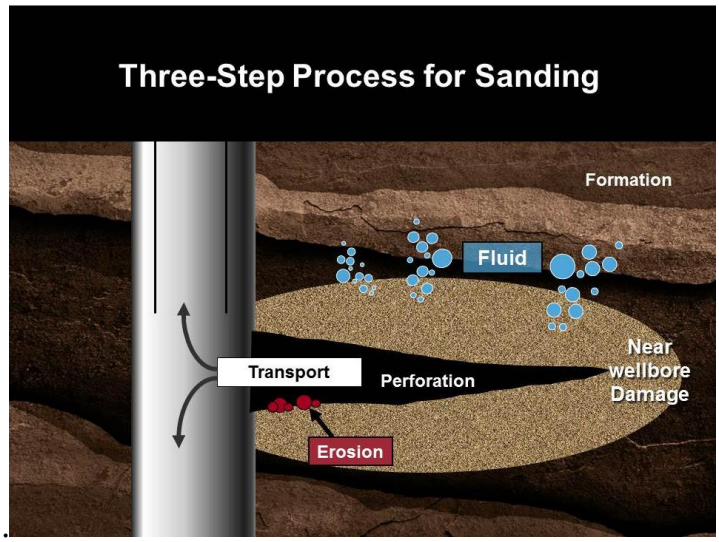


Figure 2.2. Sand production mechanism (Kim, 2012)

2.1.1 Influence of fluid properties on sand production

One of the main controlling factors of sand production can be grouped by the physical properties of the reservoir formation and fluids (Table 2-1). There are big differences between the sand production patterns for gas, light, and heavy oil wells. The cumulative sand volume and decline period for heavy oil well can differ in hundred times than for light oil wells. The type of displacing fluids had an indicative effect on sand production initiation and final sand mass (Al-Awad et al., 1999; B Wu et al., 2016).

The influence of water cut on sand production shows contradictory results in the literature. Veeken *et al.* (1991) also highlight the influence of water cut parameter on sand production: sand and no-sand production wells are characterized by an average water cut of 19% and 2%, respectively. Bailin Wu et al. (2005) conducted several experimental tests where water was introduced into the flowing stream of either oil or gas at various stages of the tests to simulate water cut. The authors conclude that the water cut effect reduces perforation strength and promotes sand production. At the same time, the other authors observed the opposite effect: sand production decreased with increasing water cut parameter (Collins et al., 2008; Tananykhin et al., 2021).

The viscosity of the fluid plays important role in sand production. The study of particles transportation by injected fluid is important for hydraulic fracturing. Harris et al. (2005) classified the fracturing fluids according to the transport capability. For the single fluids of the same viscosity (e.g., oil and water), the transportation is due to viscosity alone and settling may occur during the transport depending upon flow rate and viscosity. Hughes et al. (2008) used a new viscoelastic surfactant (VES-CO_2) fluid system as a foamed fracturing fluid to stimulate wells. Wells treated with VES-$CO2$ showed better cumulative production due to low formation damage, superior proppant transport capability. The viscosity of VES- CO_2 is higher than that of the single fluid. Low viscosity fluid (e.g., fresh water) is not an effective transport system for large particles with relatively high densities (Ahmad & Miskimins, 2019).

The traditional injection of gas or water in a reservoir with oil can create a condition whereby the interface of two fluids bypasses reservoir sections as it moves along, creating an uneven, or fingered, profile. Fingering is a relatively common condition in reservoirs with water-injection wells. The result of fingering is an inefficient sweeping action that can bypass significant volumes of recoverable oil and, in severe cases, an early breakthrough of water into adjacent production wellbores. Viscous fingering is the unstable displacement of a more viscous fluid by a less viscous liquid. Understanding the fingering dynamics is beneficial for controlling the fingering patterns and is relevant to many engineering fields, such as CO_2 sequestration and storage and enhanced oil recovery. Fingering is also the main challenge that could reduce areal sweep efficiency during waterflooding in oil reservoirs (Kargozarfard et al., 2019).

The immiscible displacements in porous media with capillary and viscous effects are characterized by two dimensionless numbers, the capillary number and the ratio M of the two viscosities. Lenormand et al. (1988) identified the fluid-fluid displacement patterns into three groups, viscous fingering, capillary fingering, and stable displacement. They consider different stages of the displacement of a very viscous oil by air and mercury displacing three various types of oils.

Moortgat (2016) investigates the viscous and gravitational fingering refer to flow instabilities in porous media that are triggered by adverse mobility or density ratios, respectively. These instabilities have been studied for single-phase flow and multi-phase immiscible and incompressible flow. In multiphase flow, relative permeabilities affect the mobility contrast for a given viscosity ratio.

Doorwar & Mohanty (2017) studied the viscous fingering and its influence on relative permeability at different viscosity ratios and flow rates. The author conducted the several waterflooding tests with core samples. The core initially saturated by oils with different viscosities

and water flood was initiated from the bottom end of the core under constant injection flowrate. As the viscosity of oil increases, the mobility ratio of water to oil increases too, and water breakthrough occurs earlier, and the remaining oil saturation is higher. Authors also conclude that effect of viscosity ratio displacing, and saturated fluids is more important than the effect of capillarity for the experimental recoveries.

2.1.2 Kazakhstan oil fields with sand production

The main task of hydrocarbons developments is maximizing the oil recovery factor (recoverable amount of hydrocarbon initially in place). The oil recovery factor depends on geological, physical, and technological factors. It is determined by the lithological composition of the reservoir, heterogeneity of the productive horizon (formation), rock permeability, and effective oil-saturated thickness. The different enhanced oil recovery approaches exist: primary methods use only the natural energy of the formation; secondary methods are associated with maintaining in-situ energy by injecting water and natural gas into the reservoir; Tertiary production methods make it possible to intensify oil inflow and ensure increased oil recovery (Alvarado & Manrique, 2010; Fanchi, 2002).

Hydrocarbon production in Kazakhstan oilfields, such as Karazhanbas, North Buzachi, Kalamkas Zhalgiztobe, and Kenkiyak, accompanied by sand production ("Kazakhstan Upstream Oil and Gas Technology and R&d Roadmap," 2013). Highly permeable and highly porous sandstones typically characterize these reservoirs with initial saturation of water above 20%, low depth, viscous oil, etc. Newly drilled wells have an initial water cut that increases during the well operations' period(Abishev et al., 2018; Collins et al., 2008; Nie et al., 2010). The fluid in reservoir conditions should be considered as multiphase.

In Kazakhstan, oil fields are mainly developed by secondary and enhanced oil recovery techniques. Bealessio et al. (2020) reviewed the existing enhanced recovery techniques and investigated their successes and challenges. Recommendations for current and potential applications of EOR in Kazakhstan were also discussed. One of the widespread oil recovery methods with sand production is Cold Heavy Oil Production with Sand (CHOPS). CHOPS is a non-thermal oil recovery method. Sand production creates high permeability channels (wormholes) for heavy oil flow. The created channels artificially increase the drainage zone. Field production has demonstrated that aggressive sand production improves the recovery performance leading to the CHOPS process. The produced sand volume is speculated to produce wormholes – depleted high permeability sand channels within the heavy oil reservoir. This means that, as long as the sand production exists, the fundamental geological parameters like porosity and formation compressibility are changed (Haddad

& Gates, 2015; Rangriz Shokri & Babadagli, 2012). Collins et al. (2008) described the geological setting of Karazhanbas oil field, reservoir and petrophysical properties, and other geological information in order to compare with Canadian fields with successful CHOPS history. CHOPS was found an economic success at Karazhanbas, allowing recovery factors of $8\ to 15\%$ from zones as thin as $3\ to\ 10\ m$, for which thermal technologies are not economically viable. The KBM case shows that CHOPS can be successfully applied to lower viscosity oils. It means CHOPS will be followed by steam stimulation to take advantage of the enhanced permeability generated by the CHOPS process.

2.2 Sand production prediction

The prediction of the sand production dynamics under different reservoir and field development plans is a paramount part of the study for the best well development, completion and exploitation strategy. There are different methods for sand prediction. Rahmati et al. (2013) define physical model testing, analytical and empirical relationships and numerical models to predict or help to understand the sand production problem (Figure 2.3). There is no unique approach for sand production study, each method has its own limitation and merits. Thus, physical models can predict volumetric sand production but at the same time they are time-consuming and expensive. Analytical models are easy to use but often most of them are able to capture only few mechanisms of sanding. Nowadays the numerical models are one of the attractive tools for predicting sand production. These model formulations are feasible to the scale of modelling and it is necessary to conduct a validation and verification tests for them (Rahmati et al., 2013).

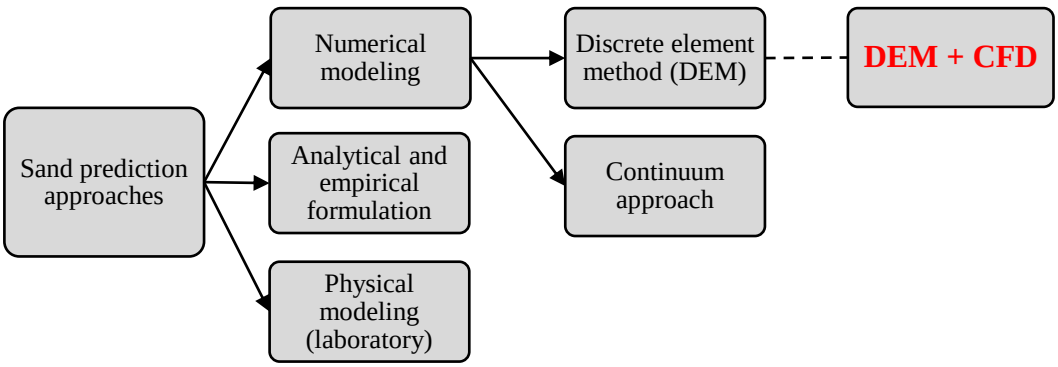


Figure 2.3. Sand prediction models

2.2.1 Analytical and empirical methods

In the empirical sand prediction techniques, the field or laboratory experiments' results are used to establish a correlation between sand production data and investigated parameters. The list of

the parameters that might impact sand production is shown in Table 2-1, the parameters' behaviors might vary in each oil and gas fields (Veeken et al., 1991). A small number of parameters are usually used due to the complexity of gathering and recording data from all wells. One parameter correlation such as cut-off depth criteria for the installation of sand control is measured relating to cut-off drawdown of the inner wall. Multi-parameter correlations can improve the resolution between sand and no-sand production but require extensive data.

Shabdirova et al. (2019) has developed an analytical model to predict the volume of produced sand based on the field data of sand production from weak sandstone formations in Kazakhstan. The produced sand is considered as the volume of the plastic zone of the failed materials surround the perforation hole. The model was validated by 20 wells from Kazakhstan oil field, sensitivity analysis was conducted to reduce the number of cases.

Kozhagulova et al. (2020) conducted an experimental investigation of the sand production pattern during well production and shut-in periods, based on which an analytical model was built. The model is in good agreement with the experimental and field data (Figure 2.4).

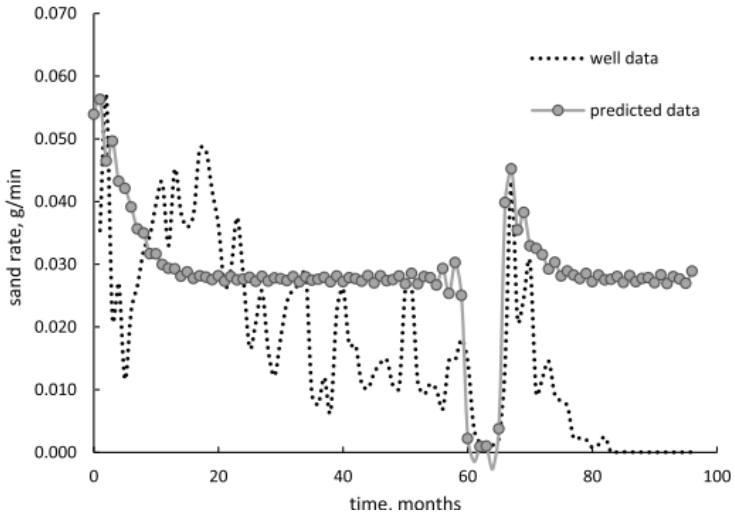


Figure 2.4. Actual and analytical model sand flow rate per each perforation (Kozhagulova et al., 2020b)

2.2.2 Laboratory methods

Sand production tests directly on production wells are costly and ineffective. The benefit of laboratory experiments compared to the field data is their ability to artificially manage the environmental conditions. In many tests researchers used modified thick wall cylinders (Figure 2.5) for tests for these purposes (Kozhagulova et al., 2021; E. Papamichos et al., 2001; Euripides Papamichos, 2006; A. D. Shabdirova et al., 2020; Veeken et al., 1991). The laboratory test provides the opportunity to narrow down the research focus and investigate the influence of various fields and

operational parameters on sand production (Rahmati et al., 2013). These tests have limitations, for example, the stronger rock materials are used more often than weak or uncemented rock. Another option for using thick wall cylinders for weak materials is to use of artificial sandstone (Kozhagulova et al., 2020a).

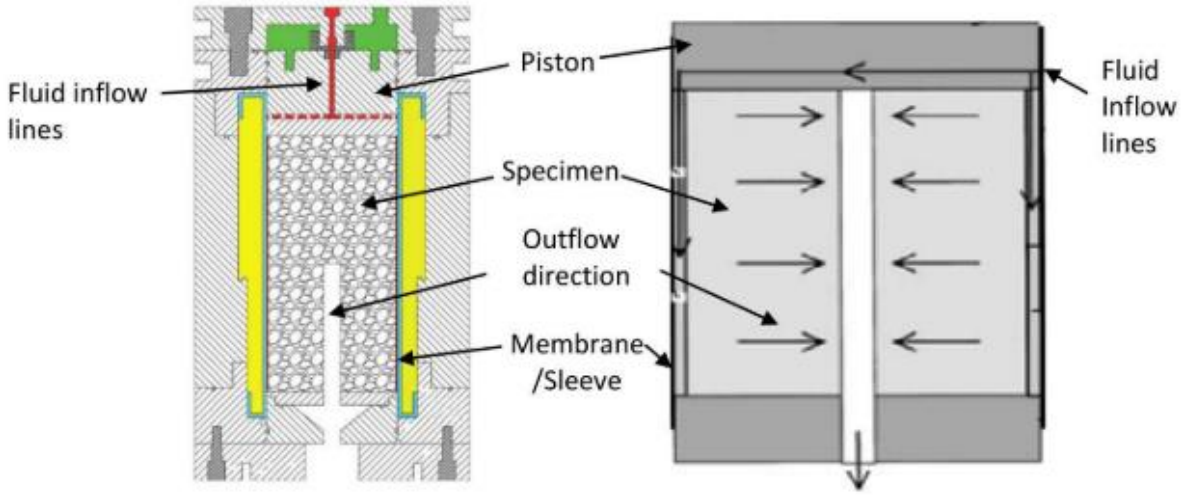


Figure 2.5. A typical sand production setup with sand trap located below the cell (Kozhagulova et al., 2021)

Al-Awad et al. (1999) conducted several sand production tests using Hoek cell (Alsayed, 2002) to investigate the effects of confining pressure, fllow rate, and the displacing fluid viscosity on sand production mechanism in unconsolidated sandstone formations (Figure 2.6). The authors concluded that, the cumulative sand production was strongly dependent on both flow rate and confining pressure. In the case of water, or low viscosity crude oil, which were saturating the rock, the sand production problem could be managed by controlling flow rate. In case of heavy crude oil, sand production mechanism became different and therefore, controlling only the flow rate could not stop sand production. It should be noted that in this paper sand rock had no cementing material.

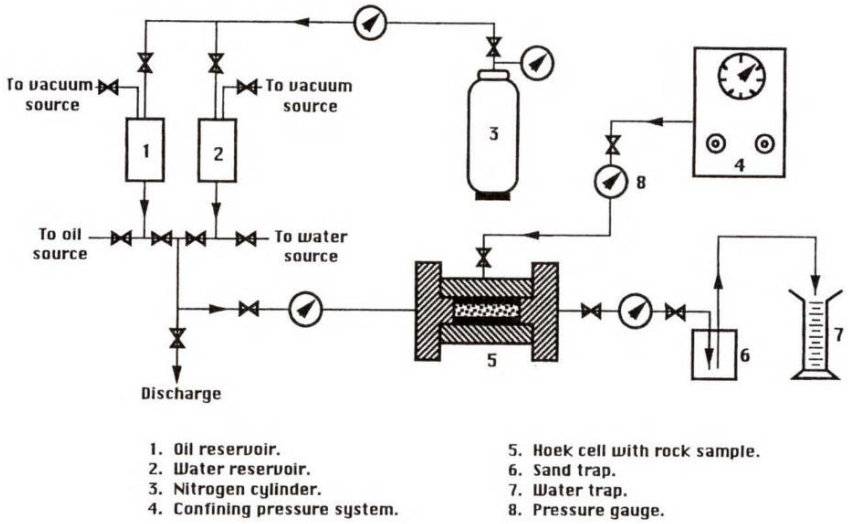


Figure 2.6. Experimental set-up(Al-Awad et al., 1999)

The type of injected fluid has an important role on sand production behaviour and final cumulative sand mass. Ray et al. (2014) conducted sand production experiments with oil, brine, and gas under irreducible water saturation on Saltwash and Castlegate sandstone samples. The results showed that the samples could withstand higher stresses during multiphase flow, which is not aligned with the field observations and the experimental studies referenced above. However, the authors conclude that the main difference between single-phase and two-phase flow was capillary pressure, especially for brine and gas flow cases. The capillary strength of the rock depends on the particle size distribution and cement structure (Brignoli et al., 1995; G. Han et al., 2002). Thus, two-phase flow and irreducible saturation of the non-wetting phase may have different effects depending on the sandstone type, which should be studied at the microscale.

Cerasi et al. (2015) studied the gas flow in hollow cylinder tests and found that sanding onset occurs at higher stresses for the gas flow than the brine flow. The authors suggested two explanations: first, the sample strengthening due to desiccation and lower drag forces due to the gas flow, which was reconsidered in terms of non-linear flow in porous media. Their hypothesis was numerically verified by Wang et al., (2019).

Wu *et al.* (2016) conducted sand production experiments on seven weak sandstones with various strengths. Experimental results for weakly-consolidated and consolidated sandstones showed that the predominant mechanism for borehole failure was shear failure around the borehole. This was followed by erosion of the loose materials due to flowing fluid in the region of rock degradation (failure zone).

Kozhagulova et al. (2020) used artificial sandstone in experimental studies, which was carried out using a High-Pressure Consolidation System (HPCS). The authors focused on the experimental investigation of the sand production pattern during well production and shut-in periods. The apparatus was equipped with a servo-hydraulic compression load frame of $5000kN$ capacity, which provided vertical compression. A cylindrical cell diameter of $300\ mm$. Fluids were injected into the sample from the outer radius of the cylindrical cell. The maximum fluid pressure could vary from 0.1 to $7\ MPa$ and flow rate up to $5\ l/min$. The general and schematic picture of the experimental system are given in Figure 2.7.

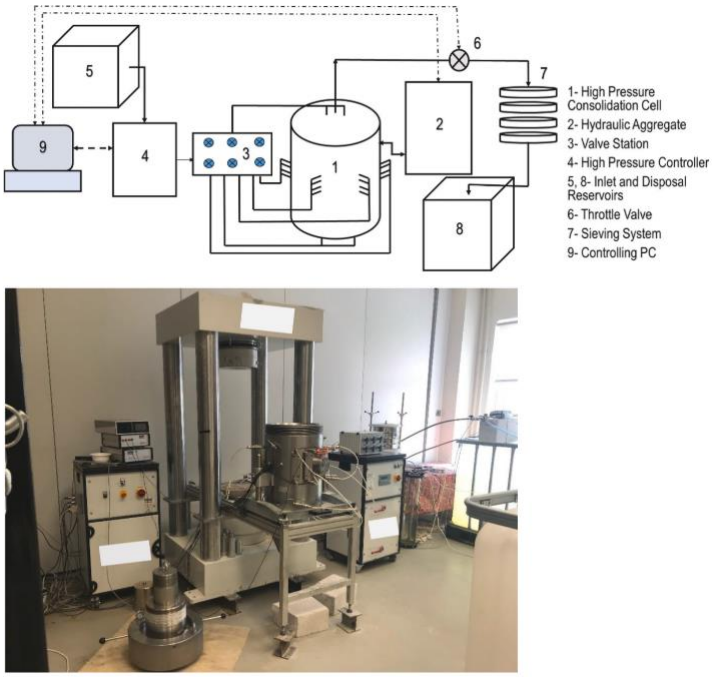


Figure 2.7. HPCS for sand production (Kozhagulova et al., 2020b)

The laboratory experiments can be used to validate the analytical and numerical sand prediction models. Furthermore, the laboratory sand production experiments can be considered as a field sand prediction tool after scaling operations.

2.2.3 Numerical models

Sand production numerical modelling consider of the two main mechanisms. The first is instability and rock breakage around the well, the second mechanism is hydromechanical impact of produced flow on transportation of sand materials surrounding the perforation or openhole. Commonly, numerical methods in the modelling are classified as continuum and non-continuum approaches.

The main idea of the continuum approach is that the substance is considered as continuum media, which means that the material cannot be separated into smaller pieces. Substances are represented as continuous in terms of differential equations. The benefit of continuum-based modelling is the ability to perform multi-scale modeling and it is less demanding on computational resources, making it possible to use the approach for large scale applications (Anderson & Jackson, 1967; X. Chen & Wang, 2014; Ding & Gidaspow, 1990; Isehunwa & Farotade, 2010; Liu et al., 2019; Rahmati et al., 2013; Wan et al., 2004).

In contrast to continuum approaches DEM becomes one of the most popular tools to investigate sand production especially the mechanism of sanding at micro-scale. DEM considers each particle of sand independently and requires significant computational power. Due to the large amount of computational time this approach is still not applicable for large- scale simulations. The calibration the parameters of the DEM model are also challenging with uncertainties. (Cundall & Strack, 1979; K. D. Kafui et al., 2002; Peng et al., 2020; Aigerim K. Rakhimzhanova et al., 2019; Colin Thornton, 2015).

Sand production is a continuous and dynamic process that occurs at the microscopic scale and the rock becomes a discontinuous in nature. This behaviors can be modelled microscopically using several numerical particle-fluid coupling techniques, among which are the two-fluid model (TFM) and the coupled computational fluid dynamics and discrete element method (CFD–DEM) as discussed by Zhu et al. (2007). The continuum approach cannot capture important features of the sanding problem, such as erosion and localized rock failure. Therefore, the non-continuum approach is more promising to simulate phenomena such as detaching small inseparable particles from the reservoir rock.

2.3 Overview of the fluid-particle coupling approaches

Sand production is a coupled fluid–solid interaction process that primarily involves two mechanisms: mechanical instabilities that lead to localized plastic behavior and failure of the rock around the cavity, and the subsequent transportation of sand particles due to the fluid-particle interaction. Most of the numerical models for sand production are obtained by combining fluids and particles models. To model fluid–particle interaction the combination of Euler (for fluid phase as continua) and Lagrange (particle phase as elements) models are used: Euler - Euler, Euler - Lagrange, Lagrange – Lagrange.

The fluid and particle models adopted are classified as micro-, meso-, and macro-scale ones depending on the modelling scale considered (Derksen et al., 2013; Y. Y. Yan, 2007).

The combination of fluid and particle interaction with different modeling scales creates different modeling approaches as shown in Figure 2.8. The clear understanding of the limits and features of these approaches is crucial for the development of adequate sand prediction models.

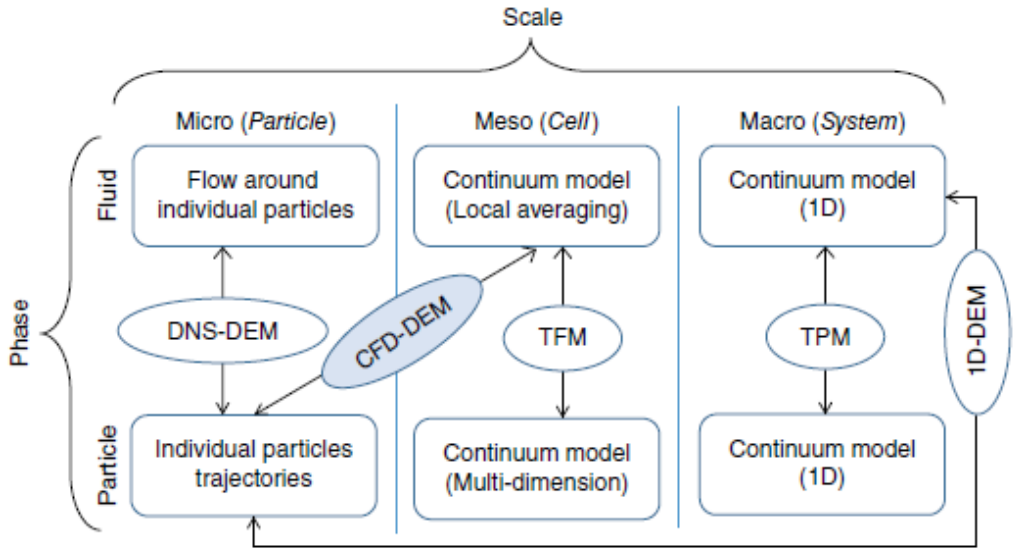


Figure 2.8. Modeling scale in fluid-particle systems (Norouzi et al., 2016)

Euler-Euler method. In this approach both fluid and solid phases are considered as continuum. The Euler-Euler model is also commonly known as two-fluid model (TFM) (Anderson & Jackson, 1967). The conservation equations of mass, momentum, and energy are obtained through appropriate averaging process. The constitutive relations for solid phase are usually closed using kinetic theory of granular flow (Ding & Gidaspow, 1990). In the continuum methods physical matter is treated as continuous in terms of differential equations. The continuity detachment means that the material cannot be separated into even smaller particles. The TFM has been successfully utilized to obtain the flow behavior of various non-reacting and reacting multiphase flows in laboratory, pilot, and industrial scales (Norouzi et al., 2016). Several sand production models of this approach are developed. Wan et all. (2004) considered the sand production as an erosion problem coupled with hydro- and geo-mechanical effects. The authors included a gas phase in the governing equations within the framework of mixture theory and multiphase flow in a four-component system, namely: solids, fluidized solids, oil and gas. Ahomogeneous, isotropic, linearly elastic rock mass model with macroscopic Darcy's law in radial form for fluid model developed to predict the onset of sand production (Isehunwa & Farotade, 2010).

Chen at al. (2014) compared the TFM and Euler-Lagrange (CFD-DEM) methods for modeling impinging gas–solid flows. The TFM method showed more potential for industrial applications since

it is less computational demanding compared to the CFD-DEM method. But at the same time TFM fails to predict the well-known particle trajectory crossing effect.

Liu *et al.* (2019) simulated the gas injection in a ladle using a porous plug using both the TFM and Euler-Lagrange approaches. The effects of various forces, bubble sizes, and bubble injection frequencies on the flow pattern are modelled. For the bubble coalescence and breakup, according to the comparison of two modeling approaches, the Euler-Lagrange approach was found as more accurate.

Euler–Lagrange method. The fluid phase as a continuum and the particulate second phase is considered as single particles. This approach predicts the particle trajectories in the fluid phase as a result of forces acting on particles. The combination of computational fluid dynamics and the discrete element method (DEM) is referred to as CFD-DEM. The surrounding fluid will interact with particles, generating various particle–fluid interaction forces, in addition to the buoyancy force. The fluid-particle interaction forces, mostly the drag force, is the driving force for fluidization. Therefore, these forces must be properly considered. To date, a number of such forces have been implemented in CFD-DEM simulation, including particle–fluid drag force, pressure gradient force, viscous force and other forces (Climent et al., 2014; Crowe et al., 2011; Yong Li et al., 1999; Zhou et al., 2010).

CFD-DEM for fluid-solid flows is used for modelling such as fluidized beds and conveyers, spouted beds, sand production and more (Cheung, 2010; Climent et al., 2014; D. K. Kafui et al., 2011; K. D. Kafui et al., 2002; O'Connor et al., 1997; Potyondy & Cundall, 2004; Aigerim K Rakhimzhanova et al., 2020; Tsuji et al., 1993; Zhou et al., 2010, 2011).

Lagrange–Lagrange method. Both fluid and solid phases are considered as non-continuum phases. Lattice Boltzmann method (LBM) is a class of CFD methods for fluid simulation which is an improved version of lattice gas automata (Hardy et al., 1973). LBM models the fluid with fictive particles, and such particles perform consecutive propagation and collision processes over a discrete lattice mesh, thus the method is relatively easy to code and ideally suited for parallel computing platforms (Blunt et al., 2013).

The LBM is widely used in multiphase and multi-component fluid flow applications. Many results are obtained in porous media at the pore scale (Dauyeshova et al., 2018; Huang et al., 2011; Parmigiani et al., 2011; Taghilou & Rahimian, 2014).

Chen *et al.* (2016) using LBM-DEM model examined the experimental observation of "initial sand production is evoked by localized failure" and then illustrated that the influence of flow rate on sand production depends on the bonding strength.

Han et al. (2017) had implemented a coupled LBM-DEM model using an immersed boundary scheme (IBM) in the Particle Flow Code. The LBM-DEM-IBM model was validated with the analytical solutions in a couple of hydrodynamics and fluid-particle interaction problems and perforation cavity.

Compared with the Euler approach, the LBM has the advantage that it does not require complex discretization of partial differential operators for multiphase fluids, but the computational speed is usually lower due to different scales of simulations. (Lohner et al., 2014).

2.4 DEM modelling

In the last decades, granular mechanics has been numerically studied using the Discrete Element Method (Cundall and Strack, 1979). In DEM, all particles in a computational domain are tracked explicitly by solving the governing equations of particle motion, which provides the detailed information of the interparticle contact forces and the particle's trajectory. Materials can be simulated in DEM in terms of the physical properties of the particles and the input parameters of the contact models, which allow for different configurations of linear and nonlinear elasto-plastic behaviour at the contact points between discrete particles(Berger et al., 2015; Hertz, 1881; K.L. Johnson, 1976; D. K. Kafui et al., 2011; Kenneth Langstreth Johnson, 1971; Lian et al., 1998; Mindlin & Deresiewicz, 1954; Colin Thornton, 2015; B. Yan & Regueiro, 2018). The cemented rock material can be modelled as an assembly of fully, or partially bonded particles. The translational and angular accelerations of the particles are computed based on the momentum balances that consider the bonding effect in the calculation of the contact forces. Numerical sandstone and sand production models-based on DEM require significant computing power which is strongly corelated to the particles number in simulation. To handle a problem of this nature the capabilities of parallel computations is commonly used (Berger et al., 2015; D. K. Kafui et al., 2011; B. Yan & Regueiro, 2018).

There are many advantages for the utilization of the DEM in sand production. The technique can be used to develop complex models that link the information at the micro-, meso- and macro scale levels. As such, it can directly relate the macroscopic performance to the intrinsic properties that define the particle specifications. Through these complex relations, the results can provide detailed information about the intricate interaction inside particle systems (Colin Thornton, 2015).

The one of the first software products for DEM modelling is TRUBAL, that was originally written by Cundall A. for the simulation of dry and moist particles systems using an explicit finite difference formulation (Lian et al., 1998). In recent decades, the numerous open-source DEM software packages have been developed. These include YadeDEM, LIGGGHTS, ESyS-particle and many others. Basically, the implemented programs solve the particles movement based on Newton's

second law using mainly the explicit finite difference method. The fundamental differences among these software lay in the particle forms, contact models, operating systems, type of parallelization, input and output capabilities, boundary conditions, object oriented programming, and visualization approaches (Kloss & Goniva, 2011; Kozicki & Donzé, 2009; Utili & Nova, 2008).

Mechanical and filtration properties of the sandstone are important parameters for sand production studies. The DEM method is a useful tool to understand the mechanism of sanding, it also has several disadvantages. The representation of the rock material in DEM computationally costly and therefore the size of problems is limited. Moreover, DEM uses micromechanical quantities and parameters of material that cannot be easily obtained from experimental tests or physical sample and should be calibrated (O'Sullivan, 2011; Rahmati et al., 2013).

2.4.1 Contact models

Elastic model. Increase in compression between two spheres, ignoring the possibility of fracture or crushing, the material behavior becomes more complex as it passes through a series of phases, namely elastic, elastoplastic, perfectly plastic and finite plastic deformation(Mesarovic & Fleck, 1999; Tomas, 2001). The simplest elastic contact force model (Hertz-Mindlin) is to assume that, during contact, two interacting bodies are connected, both normally and tangentially, by linear springs (Hertz, 1881; Mindlin & Deresiewicz, 1954; Colin Thornton, 2015).

Most materials exhibit some form of energy dissipation, either viscoelastic or plastic, and traditional elasticity models are not able to describe these behaviours. Makse *et al.* (2004) conducted experiments on isotropic compression of a granular assembly of spheres and showed that the shear and bulk moduli vary with the confining pressure faster than the $1/3$ power law predicted by Hertz-Mindlin effective medium theories of contact elasticity. For pressures larger than $\mathbf{10}\,\boldsymbol{MPa}$ the increase of the coordination number and volume fraction induces other nonlinearities and therefore no simple scaling behavior can be defined.

Elastic–plastic model. It has been shown in Finite Element Method (FEM) simulations and experiments (Müller & Tomas, 2012; Vu-Quoc & Zhang, 1999) that the relationship between the force and displacement is non-linear for elastic materials and for elastic–plastic materials immediately after the plastic yield displacement. Generally the elastic–plastic models are complex because they have to take into account the transition between elastic and plastic behaviour and between loading, unloading and reloading stages (Kamrin, 2010; C Thornton, 1997; Vu-Quoc & Zhang, 1999). Most of the models that have been developed use a piecewise approach to the different stages—that is

different force–displacement relationships are used for elastic, elastic–plastic and unloading behaviours.

Zhang and Vu-Quoc (2002) discussed the modelling of the coefficient of restitution as a function of the incoming velocity in elastoplastic collisions with normal frictionless impact, and compare the results from nonlinear finite-element analysis to those of two recent normal force displacement models: one by Thornton and one by Vu-Quoc and Zhang. (C Thornton, 1997; Vu-Quoc & Zhang, 1999). Rathbone et al. (2015) presents an accurate model for the normal force–displacement relationship between elastic–plastic spheres in contact for use in DEM simulations. Models results from FEM simulations and the DEM implementation has been seen to correctly switch between loading, unloading and reloading curves. In this study the yield stress value varied from $\mathbf{0.405}$ to $\mathbf{160}\ \boldsymbol{Mpa}$ (Rathbone et al., 2015).

Despite elastic–plastic models more accurate, simple elastic models are still using in modelling because they are less computationally expensive to calculate thus allowing the simulation of larger systems using DEM.

Adhesive contact models. The first a normal force displacement contact model with elastoplastic deformation and adhesion was introduced by Thornton and Ning (Colin Thornton & Ning, 1998). For cohesive solids, the plastic deformation in the contact region causes a larger effective radius of the deformed contact region, and upon unloading a larger resultant pull-off force is observed. They modelled such behaviour using a modified JKR (Kenneth Langstreth Johnson, 1971) curve with a larger contact radius.

Thakur *et al.* (2014) introduced an adhesive elastoplastic DEM contact model with 3D non-spherical particles to achieve quantitative prediction of cohesive powder flowability. The model comprises a nonlinear hysteretic spring model to account for the elastic–plastic contact deformation and an adhesive force component that is a function of the plastic contact deformation. Under confined compression, the porosity reduces more gradually for the load-dependent adhesion compared to constant adhesion. Author also concluded that the common adhesion models including JKR (Kenneth Langstreth Johnson, 1971), DMT (Derjaguin et al., 1975), which are elastic contact models, may not be able to capture the stress history dependent behaviour shown in experiments of cohesive powders.

Climent *et al.* (2014) calibrated dry DEM model sample against a well-known poroelastoplastic analytical solution by Risnes (1982). In this paper, the linear and parallel-bond contact models(Potyondy & Cundall, 2004) was used. In the linear contact model, the load displacement relationship between two contacting bodies is represented by linear springs.

Thornton et al. (2017) investigated elastic-plastic normal contact force models with and without cohesion. Authors conclude that in the context of non-adhesive, elastic perfectly plastic interactions involving idealized spheres, for piecewise linear models, a variable unloading stiffness needs to be scaled to the square root of the maximum relative approach. In the case of elastic perfectly plastic adhesive interactions of spherical particles, some existing models are unphysical and lead to qualitatively wrong dependencies of the coefficient of restitution on the magnitude of the impact velocity.

The modification of JKR model was developed by Rakhimzhanova et al. (2018). The authors developed new contact model for weak cemented sandstone in Kazakhstan. In this study a combination of JKR and Hertz model was used to mimic the cemented material behaviour.

2.5 CFD-DEM modeling

Tsuji et al. (1993) firstly introduced coupled particle and the fluid models to investigate the sand production phenomena. The coupled CFD-DEM model simulation was developed to investigate the particle motions in a $2D$ gas-fluidized bed system.

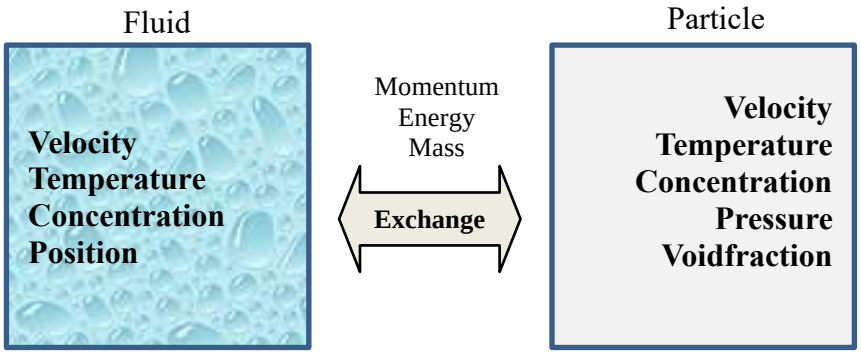


Figure 2.9. Exchange between solid and fluid phases

In CFD-DEM model fluid and particles coupling can take place through the exchange of momentum, energy, and mass among phases as shown in Figure 2.9. Particles considered in micro-scale and fluid in meso or macro scale. The trajectories of individual particles are calculated through the equation of particle motion in particle size, fluids parameters averaged locally. Therefore, implementation of efficient numerical algorithms and parallelization is necessary for any CFD-DEM code.

O'Connor et al. (1997) establish the DEM approach to investigate sand production mechanism. The Darcy flow model with DEM was combined in 2D numerical simulation. The Darcy's law has

own limits thus it is valid for slow, isothermal and viscous flow. These conditions are not always met during sand production stages (Irmay, 1958; Narsilio et al., 2009). Cheung (2010) used a coupled $1D$ fluid - $3D$ DEM model to examine the influence of fluid flow on sand mechanisms. The simplified $1D$ fluid model considers the drag force only in one direction. The fluid in terms of the Navier-stokes equation is radially injected from a cylindrical sample's outer boundary to a perforation hole. The influence of particles forces on fluid flow was ignored.

Zhou et al. (2011) built a coupled $3D$ DEM - $3D$ CFD models. Authors demonstrated that CFD-DEM model is able to capture the sand erosion processes in weak formation. The fluid-particle interaction force is the key driving force for sand production.

Kafui *et al.* (2011) has been developed a coupled parallel CFD-DEM code to simulate powder bed comprising one million particles was fluidized in the bubbling regime. Author indicates that with increasing of processors number, the global communication becomes prohibitive, leading to a degradation of the full CFD-DEM performance.

Climent (2016) used a $3D$ CFD-DEM model for sand production processes. The CFD model was built in terms of Navier-Stokes's equations. Authors used the parallel-bond model to simulate the strong cement bonds between particles (Potyondy & Cundall, 2004). The fluid flow velocity affects the sand production and plastic zone.

Rakhimzhanova et al. (2018) introduced a simple bond contact model based on the adhesive theory (Kenneth Langstreth Johnson, 1971) to investigate the behaviour of a weak sandstone in Kazakhstan oil fields conditions, which was found to have good agreement with laboratory experiment results. The model was used in further development to study the perforation and sand production of the same material (A. Rakhimzhanova et al., 2019; Aigerim K Rakhimzhanova et al., 2020), where a $3D$ coupled CFD-DEM model with resolved CFD grid was used.

Shabdirova *et al.*, (2020) improved the numerical studies by Rakhimzhanova et al. (2018, 2019, 2020) in terms of a more appropriate computation domain geometry to represent the laboratory experimental cylindrical spacemen instead of the simple parallelepiped spacemen in the previous studies, a larger simulation scale, radial flow directions and the advance scientific visualization similar to Climent *et al.* (2014). The authors investigate the permeability of the plastic zone around borehole. The porosity values of the plastic zonesin numerical study were converted into permeabilities and compared to the permeability from the semi-analytical results based on the experimental data (A. D. Shabdirova et al., 2020). It was shown that the permeability reduction patterns were similar in both numerical and experimental studies.

In the CFD-DEM simulation, the interactions between the fluid and the particles are captured in terms of the particle–fluid drag force, pressure gradient force, viscous force and other forces (Climent et al., 2014; Crowe et al., 2011; Yong Li et al., 1999; Zhou et al., 2010), of which the drag force is the main driving force for fluidization. Many correlations have been proposed to calculate the particle–fluid drag force (Di Felice, 1994; Ergun, 1952; Gidaspow et al., 1992; Koch & Hill, 2001). Kafui et al. (2002) marked that the Di Felice's (Di Felice, 1994) expression leads to a smooth variation in the drag force as a function of fluid void fraction. Agrawal et al. (2018) investigated the effect of the drag force models on CFD–DEM prediction of bubbling fluidized beds with Geldart-D (Geldart, 1973) type particles and compared the numerical results with two sets of experimental data. It was concluded that the Di Felice's model provided better prediction than the more conventional Gidaspow's model in terms of the average particle height for the bubbling fluidized bed systems. Marchelli et al. (2020) conducted the CFD-DEM simulation of two spouted beds containing Geldart-D type particles using different drag force models. Porosity and the Reynolds number are the key parameters that affect the performance of the drag force models; for the larger values, the Rong's (Rong et al., 2013) and the Di Felice's models provided better prediction, while the other models overestimated the particles' velocity.

2.5.1 Fluid resolution

In the CFD-DEM simulation the solid particles are fully resolved, and each of them is tracked in the system. The size of particle may vary depending on features of a specific task. In case of modeling small particles, inter-particle forces may become prevalent and have to be included in the equation of motion of particles (K.L. Johnson, 1976; Kenneth Langstreth Johnson, 1971; Khamitov et al., 2019).

The coupling of particles (DEM) with the fluid phase CFD) depends on the desired resolution in the fluid phase. This length scale is defined relatively to the particle size. There are two types of CFD-DEM models: resolved and unresolved approaches as seen in Figure 2.10. In resolved CFD-DEM the size of the fluid cells is much smaller than the particle size, while in unresolved approach as in opposite figure on the right the fluid cells are match larger.

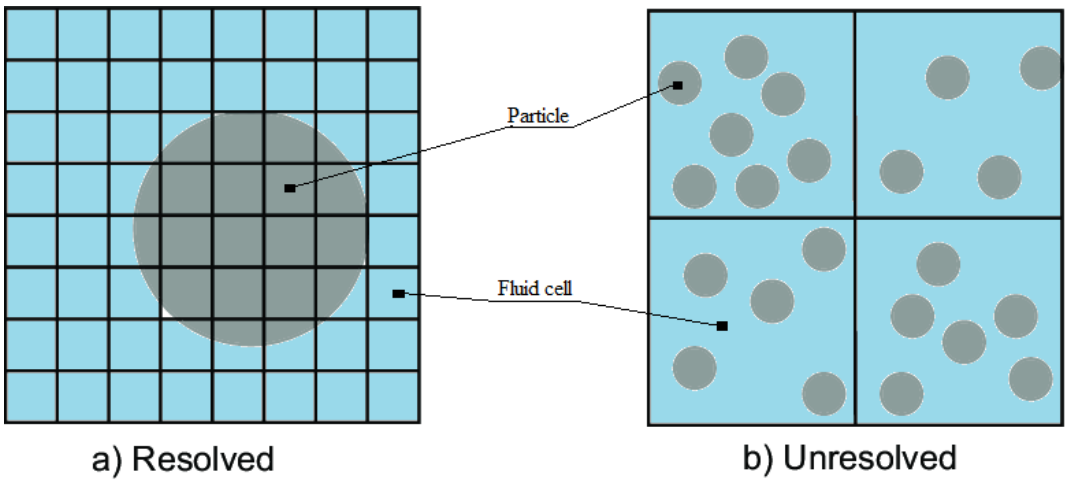


Figure 2.10. Different representations of the particle-fluid systems: (a) resolved surface method and (b) unresolved

The sand production applications can use both resolved and unresolved CFD-DEM models. Using unresolved models, it possible to examine the behavior of fluid-particle flow in the length scales which is much larger than the particle size. This gives the opportunity to run models containing even millions of particles. The coupling between solid phase at particle scale and fluid phase at fluid cell scale is performed via drag force and with empirical or theoretical correlations developed for the flow across a single particle or an assembly of particles. The effect of the presence of particles to flow profile inside the fluid cell is neglected (Christoph Goniva et al., 2010; D. K. Kafui et al., 2011; Norouzi et al., 2016; Zhou et al., 2010).

In some situation the size of fluid cell is comparable to the size of sand particles. As a result, fluid cell may be covered by a particle with a few cells touching surface of the particle or not enough number of particles can reside in each cell. In order to count the effect of "big" and rigid particles, complex and moving boundaries the Immersed boundary method (IBM) is used. Implementation of the IBM is not simple, formulation of additional force is not straightforward and requires modification of coupling approaches between particles and fluid which affect to the computation speed (Guo et al., 2013; Y. Han & Cundall, 2017; Kallemov et al., 2016; Kloss et al., 2012; Peskin, 2002).

2.5.2 Volume of fluid

The physical description of multiphase CFD-DEM can be split into two parts: modeling of the multiple fluids and of the porous media presented by particles. In multiphase CFD, the Volume of Fluid (VOF) and the Eulerian two-phase method are widely used (Enwald et al., 1996; Gueyffier et al., 1999; Hirt & Nichols, 1981; Nichols & Hirt, 1975).

VOF is a free-surface modelling technique for tracking and locating the free surface and fluid–fluid interface was introduced by Nichols and Hirt (1975). The VOF method consists of three parts:

a scheme to locate the surface, an algorithm to track the surface as a sharp interface moving through a computational grid, and a means of applying boundary conditions at the surface (Hirt & Nichols, 1981). The interface can be modeled with accuracy and fractional volume of fluid can be conserved at the same time. The disadvantages of the method lay in the computational complexity in calculating the curvature of the interface. In contrast to VOF, the Eulerian two-phase method models each of the different fluids with its own set of governing equations, which interact through added source and sink terms (Enwald et al., 1996).

The typical CFD grid for interface tracking in VOF is shown in Figure 2.11. The volume fraction in each fluid cell varies from 0 to 1, where 0 is air and 1 corresponds to water, all intermediate values are in transient zone.

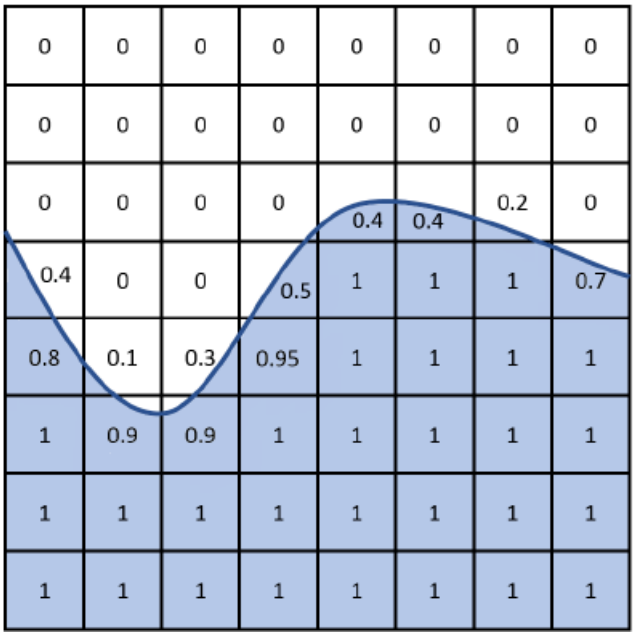


Figure 2.11. Visualization of a fluid interface on a CFD grid using the VOF method (Vångö, 2019)

The VOF is successfully implemented in OpenFoam open-source code for the interface tracking of two immiscible fluids in an Eulerian framework (Ringwood, 2015; Spitzenberger et al., 2020).

CHAPTER 3 - MATHEMATICAL FORMULATIONS AND NUMERICAL METHODOLOGY

3.1 DEM formulation

In the last decades, granular mechanics has been numerically studied using the Discrete Element Method (Cundall and Strack, 1979). In DEM, all particles in a computational domain are tracked explicitly by solving the governing equations of particle motion, which provides the detailed information of the interparticle contact forces and the particle's trajectory. Materials can be simulated in DEM in terms of the physical properties of the particles and the input parameters of the contact models, which allow for different configurations of linear and nonlinear elasto-plastic behaviour at the contact points between discrete particles (Berger et al., 2015; Hertz, 1881; K.L. Johnson, 1976; D. K. Kafui et al., 2011; Kenneth Langstreth Johnson, 1971; Lian et al., 1998; Mindlin & Deresiewicz, 1954; Colin Thornton, 2015; B. Yan & Regueiro, 2018). The cemented rock material can be modelled as an assembly of fully, or partially bonded particles. The translational and angular accelerations of the particles are computed based on the momentum balances that consider the bonding effect in the calculation of the contact forces.

In CFD-DEM simulation, a fluid phase is coupled with the particle phase (Zhu et al., 2007). In addition to the particle-particle contact forces, fluid-particle interaction forces are included in the second Newton's law for a particle p as follows (Christoph Goniva et al., 2012):

$$\boldsymbol{m_p \frac{du_p}{dt}} = \boldsymbol{F_{p,n} + F_{p,t} + F_{p,\mathrm{f}} + F_{p,P} + F_{p,vis} + F_{p,b}}, \tag{3.1}$$

$$\boldsymbol{I_p \frac{d\omega_p}{dt}} = \boldsymbol{r_{p,c} \times F_{p,t}}, \tag{3.2}$$

where $\boldsymbol{u_p}$ is linear and $\boldsymbol{\omega_p}$ - angular velocities, $\boldsymbol{I_p}$ – moment of inertia, $\boldsymbol{m_p}$- mass, $\boldsymbol{r_{p,c}}$-radius of the particle. $\boldsymbol{F_{p,t}}$ and $\boldsymbol{F_{p,n}}$ are the tangential and normal and contact forces; $\boldsymbol{F_{p,\mathrm{f}}}$ is the drag force applied from the fluid to the particle; $\boldsymbol{F_{p,Pres}}$ is pressure force, $\boldsymbol{F_{p,vis}}$ is viscous force, $\boldsymbol{F_{p,b}}$ are other forces affecting the particles.

The pressure force $F_{\mathrm{p,P}}$ defined as:

$$F_{\mathrm{p},Pres} = -\nabla P \cdot V_p \tag{3.3}$$

where V_p is volume of particle and ∇P is the fluid pressure gradient.

The viscous force $F_{\mathrm{p},vis}$ can be calculated as:

$$F_{p,vis} = -(\nabla \cdot \tau)V_p \tag{3.4}$$

where $\boldsymbol{\tau}$ is the fluid stress tensor (see Eq. 3.15).

The Eq. 3.2 is used for the rotational, while Eq. 3.1 describe the translational motion of the particle. In the case of fluid absence, the fluid-particles interaction forces are equal to zero. Different models may specify the contact forces between particles, referred to as physical properties of materials. The implementation of contact models varied among different simulation codes and programs. (Lian et al., 1998; Colin Thornton, 2015). In this research, program LIGGGHTS® (Kloss et al., 2012) for the DEM simulation was used, while the CFD phase is simulated by OpenFOAM (OpenCFD, 2021). The theoretical formulation of fluid, particle phases, and their interaction are described in this section.

The spring-dashpot model implemented in LIGGGTS opensource program is shown in Figure 3.1. The normal force is defined by:

$$\boldsymbol{F_{p,n} = -k_{p,n}\delta_p + c_{p,n}\Delta u_{p,n}} \tag{3.5}$$

where $F_{p,n}$ is the normal contact force (Eq. 3.1); $\Delta u_{p,n}$ is the normal relative velocity at the contact point; $k_{p,n}$ and $c_{p,n}$ are the normal spring stiffness and damping coefficient; δ_p is the normal overlap that for two particles i and j can be given as:

$$\boldsymbol{\delta_p = R_i + R_j - d} \tag{3.6}$$

where R_i, R_j are the radii of particle i and j, respectively; d is the distance between the particles' centers.

The tangential force can be written as:

$$\boldsymbol{F_{p,t} = min\left\{\left|k_{p,t}\int_{t_{c,0}}^{t}\Delta u_{p,t}dt + c_{p,t}\Delta u_{p,t}\right|, \mu_c F_{p,n}\right\}} \tag{3.7}$$

where $\Delta u_{p,t}$ is the relative tangential velocity (Eq. 3.7); $k_{p,t}$ is the tangential spring stiffness, $c_{p,t}$ is the tangential coefficient of damping; μ_c - friction coefficient. The Eq. 3.7 shows summation of spring energy since the time $t_{c,0}$ to the current time, t. The maximum of the tangential force defined by the Coulomb force.

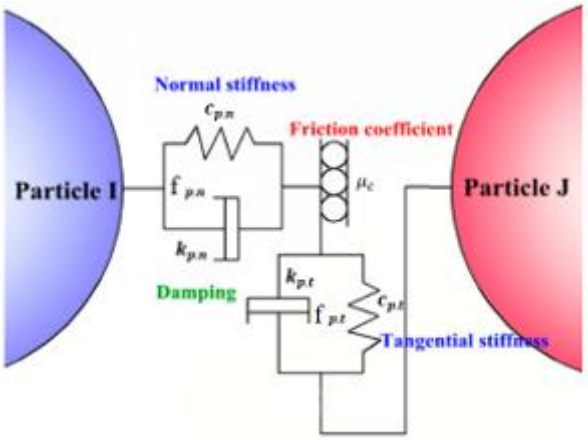


Figure 3.1. Graphic representation of the linear-spring/ soft-sphere model (Cheng et al., 2018)

The interaction between two particles shown in Figure 3.1. This interaction depends on the radii of the particles, the surface energy properties, the particles overlap and the material properties. The contact interaction may be expressed by various elastic, elasto-plastic, linear, and non-linear constitutive contact models. Hertz's theory (Hertz, 1881) considers the particles perfectly elastic with zero resistance to tensile force that separates particles along the radial direction. The normal stiffness $k_{p,n}$ for two particles i and j (see Eq. 3.5) can defined as Young's modulus function Y_i, Y_j and the Poisson's ratio v_i, v_j of the two particles in contact as:

$$\boldsymbol{k_{p,n} = \frac{4}{3} Y^* \sqrt{R^* \delta_p}} \tag{3.8}$$

where $Y^* = \frac{Y_i Y_j}{\left[Y_j\left(1-v_i^2\right)+Y_i\left(1-v_j^2\right)\right]}$ and $R^* = \frac{R_i R_j}{R_i+R_j}$.

The Rayleigh time is a critical DEM time-step is given by the following equation (Yanjie Li et al., 2005):

$$dt_r = \pi \cdot r_{min} \cdot \sqrt{\frac{\rho_p}{G}} / (0.1631 \cdot v + 0.8766) \tag{3.9}$$

where G is particle shear modulus, r_{min} is minimum radii of particles and ρ_p-density of particle.

JKR theory describes the behaviour of the auto-adhesive contact due to the appearance of Van der Waals forces between two particles For the small finite particles, the VW forces become significant, and particles tend to attract to each other (Colin Thornton, 2015). But in this study, the

JKR model is merely modified, and its parameters are calibrated to mimic the bonding forces in cemented particles as a model to simulate the behaviours of sandstone.

The overlap and the normal contact force dependences in the JKR contact model is shown in Figure 3.2. When the two particles are located nearby, they stick to each other, which is presented by a jump in the value of $F_{p,n}$ from point O to point A in Figure 3.2. Note that $F_{p,n}$ is negative at the point A as it is a tensile force. As two particles overlap each other between point A and point B, the curve presents a loading procedure as $F_{p,n}$ increases with the value of the normal overlap δ_p value. This relation can be define as (Kenneth Langstreth Johnson, 1971):

$$(\boldsymbol{\delta_p R^*})^{3/2} = \frac{\mathbf{3}}{\mathbf{4}} \boldsymbol{R^* Y^*} \left[\boldsymbol{F_{p,n} + 3\gamma\pi R^* + \sqrt{6\gamma\pi R^* F_{p,n} + (3\gamma\pi R^*)^2}} \right] \tag{3.10}$$

where γ is cohesion energy of particles.

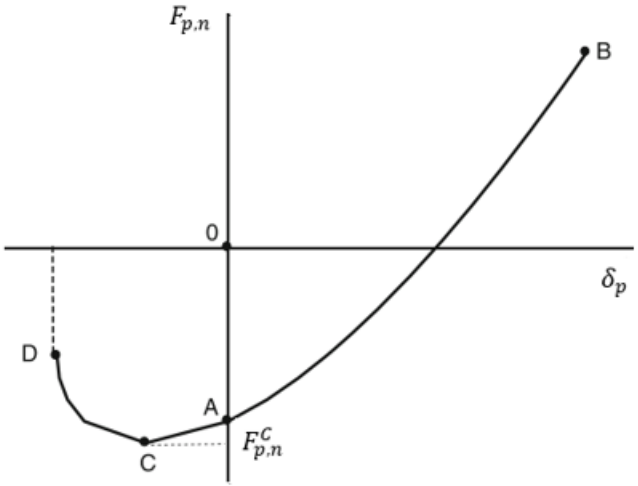


Figure 3.2. Normal contact force behavior for JKR model (Colin Thornton, 2015)

The unloading process have happened from F_B to F_A when the normal force decreases with the overlap. At the point A, during unloading, the two particles overlap starts to be negative. In this case, particles are still stuck together. The tensile force continues to a maximum value $F_{p,n}^C$ at point C, which is defined as $F_{p,n}^C = 1.5\pi\gamma R^*$. The particle's contact will break only at point D. In the case of adhesion energy is equal to zero, the JKR model transforms to the Hertz model.

Time integration. In order solve the differential equations Eq. 3.1-3.2, the derivatives must be approximated with Taylor expansions depending on the selected integration scheme, which similarly, varies in complexity, order of accuracy and stability. Some common available integration schemes include the simple first order, explicit Euler or semi-implicit Euler-Cromer schemes (Cromer, 1981). In LIGGGHTS®, the second order accurate Velocity-Verlet (Verlet Loup, 1967) scheme is used,

which uses a two-step approach to integrate the velocities (Seil, 2016). It is explicit, unconditionally stable, and accurate to second order. In the first step, the particle position x is integrated over the full-time step using the half step velocity as:

$$u_p\left(t+\frac{\Delta t}{2}\right)=u_p(t)+\frac{\Delta t}{2}\frac{du_p}{dt}(t) \tag{3.11}$$

$$x(t+\Delta t)=x(t)+\Delta t u_p\left(t+\frac{\Delta t}{2}\right) \tag{3.12}$$

In the second step, the velocity is integrated for the remaining half step as:

$$u_p(t+\Delta t)=u_p(t+\Delta t/2)+\frac{\Delta t}{2}\frac{du_p}{dt}(t+\Delta t) \tag{3.13}$$

where $(t+\Delta t)du_p/dt$ is calculated from the forces in Eq. 3.1

Weak cement sandstone model. Rakhimzhanova et al. (2018) combine the original JKR with Hertz contact models to mimic the weak cemented sandstone behaviour. The authors made two assumptions. First, the bond contact breaks in the JKR model occur at the point C. The second assumption is that the bonds are formed only once at the beginning of the simulation. In order to mimic cemented material behaviour, during the following compression conditions, the bond contacts can break, and the particles formed Hertz contacts. The particle sizes were taken from the real sand material of study by Shabdirova et al. (2016), and they are much larger than silt as in Van der Waals theory (Margenau, 1939). The Van der Waals forces was not used for sand production. But in our study, the JKR model is merely modified, and its parameters are calibrated to mimic the bonding forces in cemented particles as a model to simulate the behaviours of sandstone. Therefore, this cement model is used in our study and will call it as Weak Cement Model (WCM).

Similar to the experiment of Kozhagulova *et al.* (2020) numerical simulation was carries out in the following steps: particle generation based on experimental size distribution and their deposition by gravity; sample consolidation, cementation by fixed overburden stress, perforation and sand production (Figure 3.3). Initial sample preparation process consists of three main stages: particle generation and pluvial deposition, consolidation, cementation. For this purpose, the original JKR and WCM contacts models were used.

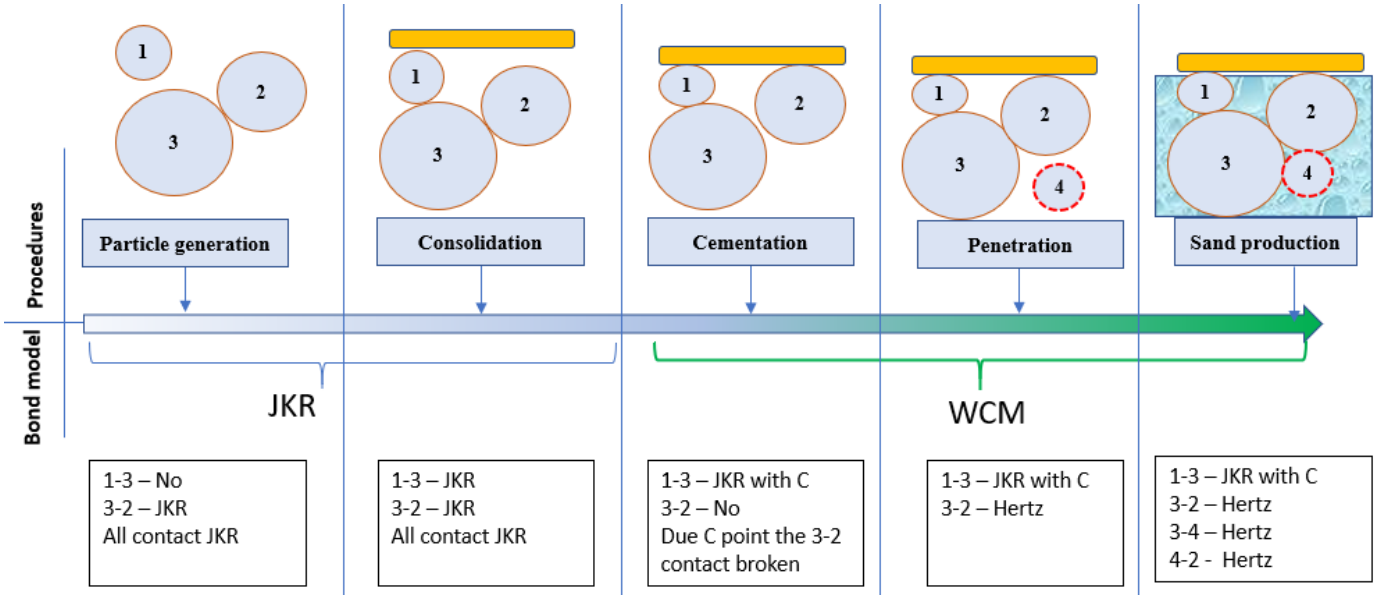


Figure 3.3. The sand production simulation workflow

Figure 3.3 shows the three particles' interactions during the following simulation's steps: particle generation, consolidation, cementation, sample penetration and sand production. The original JKR contact model describes the normal contact force behavior for particle generation and consolidation steps. The generated auto-adhesive particles are compressed by an impermeable wall (yellow colored) to achieve predetermined overburden stress. Further, the WCM model combines the modified JKR (bond contact breaks at point C, see Figure 3.2, and Hertz contact models is used for cementation, penetration, and sand production steps. The WCM mimics cement material behavior; thus, contact cannot stick back once broken, and particles form new Hertz contacts. The have broken contact between particles 2 and 3 at the cementation step in the next penetration step will interact in Hertz terms. The block diagram of contact models is shown in Figure 3.4.

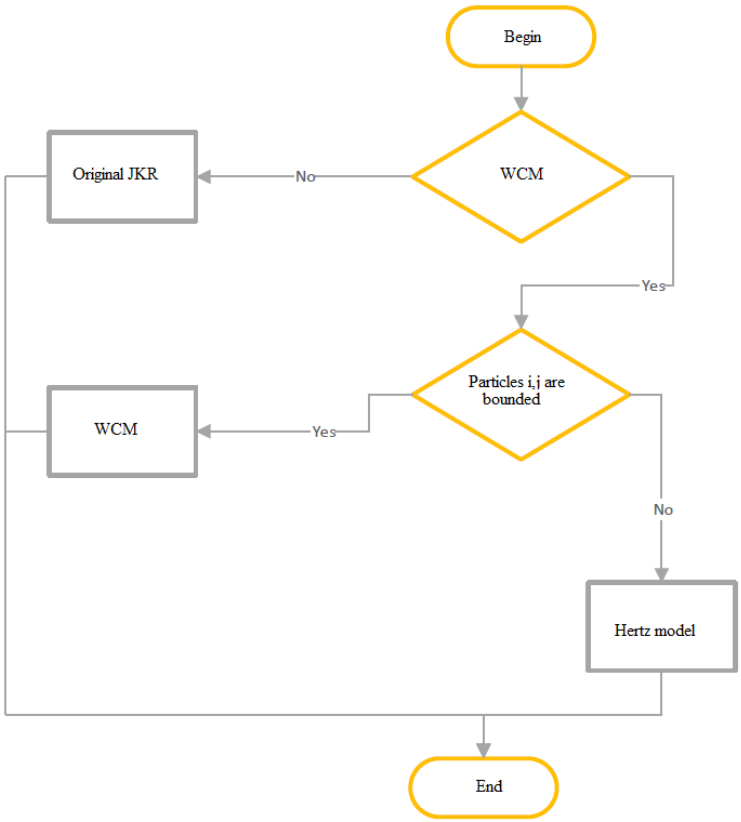


Figure 3.4. The block diagram of contact model for i, j particles pair

In this research the WCM contact model is described above was successfully implemented in opensource program LIGGGHTS(C. Goniva et al., 2015). The WCM contact model simulation results were verified with theoretical solutions of modified JKR and Hertz models (Khamitov et al., 2019). In this research, the sand production phenomena were investigated by using the CFD-DEM model with new developed WCM.

3.2 One-phase fluid CFD-DEM model

Tsuji et al. (1993) introduced the coupled CFD-DEM model to investigate a sand production phenomenon. The CFD-DEM simulations in this study, the fluid is considered a continuum, and the solid phase as discontinues matters. The solid particles are fully resolved, and each of them is tracked in the system. Depending on CFD cell and particle sizes, the unresolved and resolve approaches are used in CFD-DEM modelling. The unresolved approach assumed that the particle sizes are smaller than CFD cell (Norouzi et al., 2016). Individual particles do not completely fill a computational fluid cell. The coupling between solid phase at particle scale and fluid phase at fluid cell scale can be done via drag force and with empirical or theoretical correlations developed for the flow across a single particle or an assembly of particles(Christoph Goniva et al., 2010; D. K. Kafui et al., 2011; Norouzi et al., 2016; Zhou et al., 2010).

Zhou et al. (2010) assessed different CFD-DEM models theoretically and practically and recommended Model A for CFD–DEM modelling. In this model the governing equations of a fluid phase in the presence of a secondary particulate phase can be given as follows:

$$\begin{cases} \dfrac{\partial(\varepsilon_{\mathrm{f}})}{\partial t} + \nabla \cdot (\varepsilon_{\mathrm{f}}\, u_{\mathrm{f}}) = 0 \\ \dfrac{\partial(\varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}u_{\mathrm{f}})}{\partial t} + \nabla \cdot (\varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}u_{\mathrm{f}}u_{\mathrm{f}}) = -\varepsilon_{\mathrm{f}}\nabla P - F_{\mathrm{pf}}^{set\ II} + \varepsilon_{\mathrm{f}}\nabla \cdot \tau + \varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}g \end{cases} \tag{3.14}$$

where $\boldsymbol{\varepsilon_{\mathrm{f}}}$ is the fluid volume fraction, $\boldsymbol{u_{\mathrm{f}}}$ – velocity of fluid, $\boldsymbol{\rho_{\mathrm{f}}}$ the fluid density, $\boldsymbol{P}$ the pressure, $\boldsymbol{F_{\mathrm{pf}}^{set\ II}}$ – volumetric fluid-particle force, $\boldsymbol{g}$ the gravity constant, $\boldsymbol{t}$ the current time.

The $\boldsymbol{\tau}$ -stress tensor with bulk $\boldsymbol{\lambda_{\mathrm{f}}}$ and shear $\boldsymbol{\mu_{\mathrm{f}}}$ viscosities is calculated as:

$$\tau = \mu_f(\nabla u_{\mathrm{f}} + (\nabla u_{\mathrm{f}})^T) + (\lambda_{\mathrm{f}} - \frac{2}{3}\mu_{\mathrm{f}})I\nabla \cdot u_{\mathrm{f}} \tag{3.15}$$

The volumetric fluid-particle interaction force $\boldsymbol{F_{\mathrm{pf}}^{set\ II}}$ determine by:

$$F_{\mathrm{pf}}^{set\ II} = \frac{1}{\Delta V}\sum_{p=1}^{n}(f_{p,\mathrm{f}}) \tag{3.16}$$

$$f_{p,\mathrm{f}} = f_{p,d} + f_p'' \tag{3.17}$$

where $\Delta\boldsymbol{V}$- volume of CFD cell, $\boldsymbol{n}$ is a total particle number in CFD cell, $\boldsymbol{f_{p,\mathrm{f}}}$ is fluid-particle force in Eq. 3.1; $\boldsymbol{f_{p,d}}$ - a drag force, $\boldsymbol{f_p''}$ is the sum of fluid- particle forces on particles other than the drag force.

The Di Felice drag force correlation (Di Felice, 1994):

$$f_{p,d} = 0.125C_{d0,p}\rho_{\mathrm{f}}\pi d_p^2\varepsilon_p^2\left|u_{\mathrm{f}} - u_p\right|\left(u_{\mathrm{f}} - u_p\right)\varepsilon_p^{-\chi} \tag{3.18}$$

where unknown χ , $C_{d0,p}$ and Re_p defined by:

$$\chi = 3.7 - 0.65\exp\left[-(1.5 - \log_{10} Re_p)^2/2\right] \tag{3.19}$$

$$C_{d0,p} = (0.63 + 4.8/Re_p^{0.5})^2 \tag{3.20}$$

$$Re_p = \rho_f d_p \varepsilon_p |u_\mathrm{f} - u_p| / \mu_\mathrm{f} \tag{3.21}$$

u_p is particle velocity, $\varepsilon_p = 1 - \sum_{j=1}^{n} V_j / \Delta V$, d_p is particle diameter, $p = \overline{\overline{1 \div n}}$.

The size of the fluid cell in the CFD modelling should be related to the particle sizes in the DEM modelling. For the CFDEM software in this study, the CFD grid size was selected according to the following condition (DCS Computing GmbH, n.d.):

$$\frac{\Delta x_{cfd}}{d_p} > 3 \tag{3.22}$$

where Δx_{cfd} is the length of the fluid cell in the $\boldsymbol{x}$ - direction

Finite volume discretization. By treating a fluid as a continuum, its motion can be described by the *Navier-Stokes* (NS) equations. Direct solutions of these are virtually impossible to obtain, hence in CFD, fluid flow problems are solved numerically by approximating the continuous NS equations as algebraic ones, applied to discrete points on a computational grid (Batchelor, 1967). In this section, governing equations, discretization methods as well as the solution procedure of CFD, utilizing a VOF approach for the multiple fluids were covered. The presented theory in this section mainly follows Moukalled et al. (2015) unless specific references are given.

In order to practically solve the continuous partial differential equations (Eq. 3.14), they must be discretized. In OpenFOAM, the finite volume method (FVM) is used, where the domain is discretized into finite volumes and the quantities in question solved at the volume elements' (cells) centroids (OpenCFD, 2021). The discretization procedure in the FVM by considering the generic incompressible convection-diffusion equation of ϕ can be expressed by:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (u_\mathrm{f} \phi) = \nabla \cdot (\mathrm{D} \nabla \phi) + S \tag{3.23}$$

where where D is the diffusion coefficient and S a source term. In a first step, Eq. 3.23 integrated over an arbitrary control volume c, and subsequently apply Gauss's theorem to the divergence terms which yields

$$\frac{\partial}{\partial t} \int_{V_c} \phi dV + \oint_{S_c} \nabla \cdot (u_\mathrm{f} \phi) d\boldsymbol{S} = \oint_{S_c} \nabla \cdot (\mathrm{D} \nabla \phi) d\boldsymbol{S} + \int_{V_c} S dV \tag{3.24}$$

where $\boldsymbol{S}$ is the bounding surface of c with the normal directed outwards from the control volume (Figure 3.5). This integrated equation is often referred to as the semi discretized form and in order to solve it, each term must be approximated.

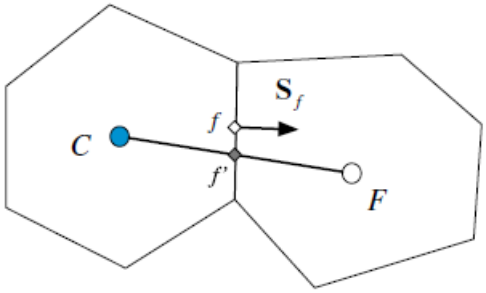


Figure 3.5. Parameters in finite volume discretization in 2-dimensional case (Moukalled et al., 2015)

Firstly, the transient term is treated with a finite difference (FD) approach to propagate in time. A combination of Taylor expansions with regards to time t can be employed to create various temporal discretization schemes (further described later in this section). Assuming a simple forward Euler scheme, the Taylor expansion reads:

$$\phi^{n+1} = \phi^n + \frac{\partial \phi^n}{\partial t}\Delta t + \frac{\partial^2 \phi^n}{\partial t^2}\frac{\Delta t^2}{2!} + \cdots \tag{3.25}$$

in which the superscripts $\phi^{n+1} = \phi(t + \Delta t)$ and $\phi^n = \phi(t)$ denote the solved or/and old-time step respectively. The temporal derivative is then approximated as:

$$\frac{\partial \phi}{\partial t} = \frac{\phi^{n+1} - \phi^n}{\Delta t} + O(\Delta t) \tag{3.26}$$

if cut of the series from the Δt^2 term. The last term $O(\Delta t)$ represents the truncation error of the neglected terms and is commonly used to indicate a discretization scheme's order of accuracy. Hence, the forward Euler scheme is first order accurate. Consequently, the first term in Eq. 3.24 can be discretized following:

$$\frac{\partial}{\partial t}\int_{V_c} \phi dV = \frac{\phi^{n+1} - \phi^n}{\Delta t} V_c \tag{3.27}$$

with the selected forward Euler scheme.

Furthermore, in the FVM, the convective term is discretized by summing the flux of ϕ over the control volume's faces f as:

$$\oint_{S_c} \nabla \cdot (\boldsymbol{u}_{\mathrm{f}}\phi) d\boldsymbol{S} \approx \sum_f \boldsymbol{u}_{\mathrm{f}f}\phi_\tau \cdot \boldsymbol{S}_f \tag{3.28}$$

and similarly for the diffusive term if D is assumed a scalar, it is approximated as:

$$\oint_{S_c} \nabla \cdot (\mathrm{D}\nabla\phi) d\boldsymbol{S} \approx \sum_f D_f(\nabla_f\phi \cdot \boldsymbol{S}_f) \tag{3.29}$$

, where $\nabla_f\phi$ is the face normal gradient. Finally, if the source term S is a function of ϕ, it is firstly linearized then integrated over the control volume as

$$S(\phi) = S_p\phi + S_u \tag{3.30}$$

$$\int_{V_c} S(\phi) dV = S_p\phi V_c + S_u V_c \tag{3.31}$$

Moreover, in order to propagate Eq. 3.24 in time, it can be enclosed in time integrals as

$$\begin{aligned} \int_t^{t+\Delta t} \left[\frac{\partial}{\partial t}\int_{V_c} \phi dV + \oint_{S_c} \nabla \cdot (u_{\mathrm{f}}\phi) d\boldsymbol{S}\right] dt \\ = \int_t^{t+\Delta t} \left[\oint_{S_c} \nabla \cdot (\mathrm{D}\nabla\phi) d\boldsymbol{S} + \int_{V_c} S dV\right] dt \end{aligned} \tag{3.32}$$

Similarly, as for the transient term, a temporal FD discretization scheme is employed depending on desired level of accuracy and stability. Beyond the previously presented forward Euler scheme, some popular ones include the first order implicit backward Euler and the second order blended Crank-Nicolson schemes(Crank & Nicolson, 1947). Continuing with an implicit Euler formulation and inserting the discretized spatial terms, Eq. 3.32 can be written as:

$$\frac{\phi_c^{n+1} - \phi_c^n}{\Delta t} + \sum_f u_{\mathrm{f}f}\phi_f^{n+1} \cdot \boldsymbol{S}_f = \sum_f D_f(\nabla_f\phi_f^{n+1} \cdot \boldsymbol{S}_f) + S_p V_c \phi_c^{n+1} + S_u V_c \tag{3.33}$$

As shown above several terms in Eq. 3.33 are represented by the face values ϕ_f and how to obtain them are an intrinsic part of the FVM. Face values have to be obtained from the neighboring cell-values based on a selected interpolation scheme, which also vary in accuracy and stability. These schemes range from the simple first order *upwind* and second order *central difference* schemes, to more advanced ones as e.g. the *Quadratic Upstream Interpolation for Convective Kinematics (QUICK)* (Leonard, 1979) or *FROMM* (Fromm, 1968) schemes.

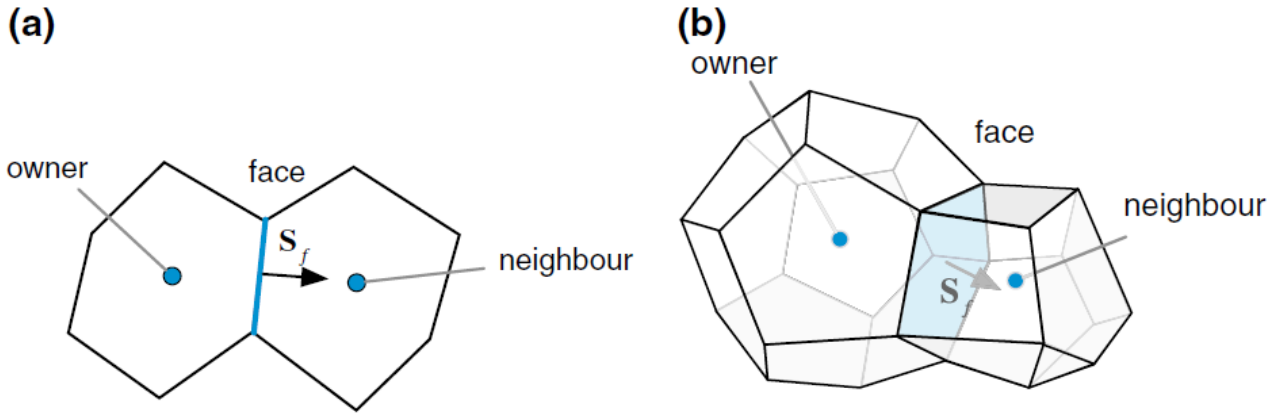


Figure 3.6. Owners, neighbors, and faces for a) 2-dimensinal and b) 3-dimensional elements(Moukalled et al., 2015)

If Eq. 3.33 is written in the general form:

$$a_c \phi_c^{n+1} + \sum_N a_N \phi_N^{n+1} = b_c \tag{3.34}$$

in which N denotes the neighboring cells of c (Figure 3.6) The system of equations for all cells in the computational domain can be assembled in matrix form as:

$$[A][\phi] = b \tag{3.35}$$

, where $[A]$ is a matrix containing the coefficients a_c and a_N, $[\phi]$ is the ϕ's and b the source terms.

In the system (Eq. 3.35), the coefficients of the unknown variables constituting matrix A are the result of the linearization procedure and the mesh geometry, while vector b contains all sources, constants, boundary conditions, and non-linearizable components. The approaches for solving linear systems of algebraic equations are generally classified into direct (Gauss elimination, LU factorization, Tridiagonal and Pentadiagonal matrix algorithms) and iterative methods (Jacobi, Gauss-Siedel, Incomplete LU factorization, and the conjugate gradient methods), with many sub-groups in each category (Press et al., 2007; Saad, 2003). Due to flow problems usually are non-linear,

the coefficients resulting from their linearization process are generally solution dependent (Moukalled et al., 2015).

Solution of the momentum equation. In practice when solving the fluid's governing equations, evaluating the complicated system of equations by inversing A (Eq. 3.35) is not feasible due to the nonlinearity of the momentum equation. Hence, iterative methods such as the popular *SIMPLE* (S. Patankar, 1980) and *PISO* (Issa, 1986) algorithms are employed. PISO is a pressure and velocity calculation procedure for the Navier-Stokes equations, originally developed for non-iterative calculation of non-stationary compressible flow, but it has been successfully adapted to stationary problems. OpenFOAM uses a combination of the SIMPLE and PISO algorithms called *PIMPLE*(G. Chen et al., 2014) to solve the pressure-velocity coupling for transient simulations. In essence, the PIMPLE algorithm allows for outer corrector steps in addition to the PISO loop to enable some benefits from the SIMPLE algorithm, e.g., under-relaxation. However, throughout this work a standard PISO solver was used to solve Eq. 3.11 as no outer corrector steps were utilized.

By employing the previous discretization methodology, the momentum equation Eq. 3.14(b) can be written in the semi-discretized form:

$$a_c u_{\mathrm{f}c} = \mathbf{H}(u_{\mathrm{f}}) - \nabla p^* - g \cdot x \nabla \rho_f + F^\sigma \tag{3.36}$$

where a_c is the matrix coefficient of c and $\mathbf{H}(u_{\mathrm{f}})$ containing the viscous term and the matrix coefficients of the neighboring cells as (Jing et al., 2016):

$$\mathbf{H}(u_{\mathrm{f}}) = \nabla \cdot \tau - \sum_N a_c \, u_{\mathrm{fN}} \tag{3.37}$$

Correspondingly, the first step of the PISO loop is performed by predicting the velocity as:

$$u_{\mathrm{f}c} = \left[\mathbf{H}(u_{\mathrm{f}}) - \nabla p^* - g \cdot x \nabla \rho_f + F^\sigma\right] / a_c \tag{3.38}$$

followed by a pressure correction procedure, in which the Laplace pressure equation

$$\nabla \cdot (\nabla p^* \frac{1}{a_c}) = \nabla \cdot [\left(\mathbf{H}(u_{\mathrm{f}}) - \nabla p^* - g \cdot x \nabla \rho_f + F^\sigma\right) \frac{1}{a_c}] \tag{3.39}$$

is solved, and its solution used to update the velocity to enforce mass conservation. Thus, Eq. 3.39 was obtained by substituting Eq. 3.38 into the continuity equation Eq. 3.14(a). The pressure correction is performed a prescribed number of times.

CFD-DEM coupling. The information in the CFD-DEM model is exchanged between solid (DEM) and of the fluid (CFD) phases computations. The fluid-particle interaction force $f_{p,\mathrm{f}}$ is a "bridge" in the CFD-DEM model. The DEM sends the new coordinated of particles to the CFD parts, where the fluid volume fraction ε_{f} is calculated in each CFD cell (see Eq. 3.14). The CFD provides the updated velocity and pressure fields for all forces acting on particle. (see Eq. 3.1 and Eq. 3.14). The blog diagram of coupling the CFD and DEM simulations has shown in Figure 3.7.

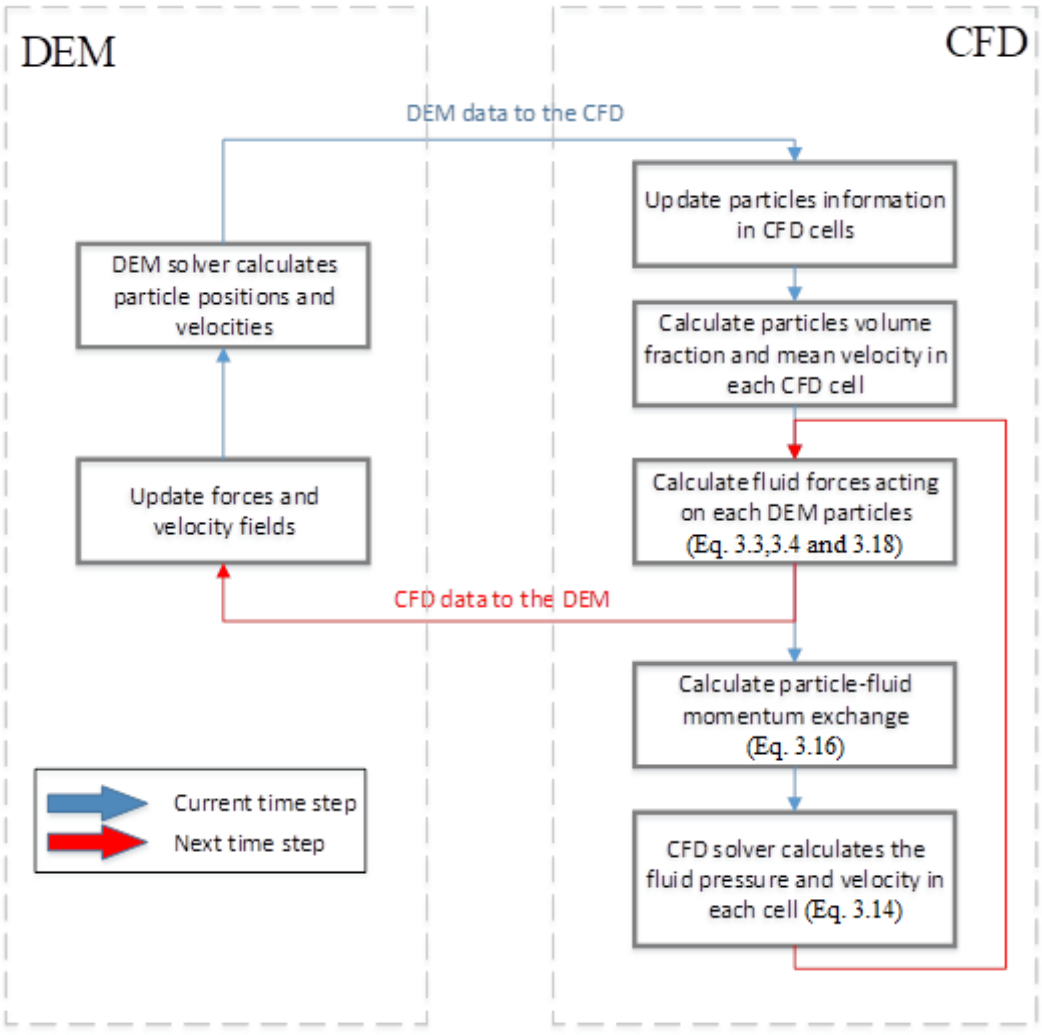


Figure 3.7. The blog diagram of the CFD-DEM coupling

The viscous force calculation is included in the DEM formulation (Eq. 3.4). The model described above is already implemented in CFDEM (Christoph Goniva et al., 2012). In this model,

the non-resolved approach was implemented. The size of particles is several times smaller than the CFD cell, and particles can not fill a CFD cell. This approach allows running simulation with million particles. The relation between particles sizes and CFD grid size can be operated by Eq. 3.22.

3.3 Two-phase fluid CFD-DEM model

The governing equations describing the motion of a two-phase fluids flow with particles ($phase1$ and $phase2$) using the well-known interface tracking VOF (volume of fluid) method can be defined by (Hirt & Nichols, 1981):

$$\begin{cases} \dfrac{\partial(\varepsilon_{\mathrm{f}}\alpha_{ph1})}{\partial t} + \nabla\cdot\left(\varepsilon_{\mathrm{f}}\,\alpha_{ph1}u_{\mathrm{f}}\right) - \nabla\cdot\left(u_c\,\alpha_{ph1}\alpha_{ph2}\right) = 0 \\ \dfrac{\partial(\varepsilon_{\mathrm{f}})}{\partial t} + \nabla\cdot\left(\varepsilon_f u_{\mathrm{f}}\right) = 0 \\ \alpha_{ph1} + \alpha_{ph2} = 1 \\ \dfrac{\partial(\varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}u_{\mathrm{f}})}{\partial t} + \nabla\cdot(\varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}u_{\mathrm{f}}u_{\mathrm{f}}) = -\varepsilon_{\mathrm{f}}\nabla P - F_{\mathrm{pf}}^{set\ II} + \varepsilon_{\mathrm{f}}\nabla\cdot\tau + \varepsilon_{\mathrm{f}}\rho_{\mathrm{f}}g \end{cases} \quad (3.40)$$

where ε_{f} is the fluid fraction, $\alpha_{ph1}$and α_{ph2} are volume fraction of $phase1$ and $phase2$, u_c - denotes the compression velocity $u_c = u_{\mathrm{f},ph1} - u_{\mathrm{f},ph2}$ at the interface between the phase1 and phase2, u_{f} - fluid velocity, ρ_{f} the fluid density, P the pressure, $F_{\mathrm{pf}}^{set\ II}$ – volumetric fluid-particle force, g the gravity constant and t the current time. The τ – the tensor is calculated similar to previous section.

The fluid density ρ_{f} and viscosity μ_{f} determines using the volume-weighted fluid-mixture properties:

$$\boldsymbol{\rho_{\mathrm{f}} = \alpha_{ph1}\rho_{ph1} + \alpha_{ph2}\rho_{ph2}} \quad (3.41)$$

$$\boldsymbol{\mu_{\mathrm{f}} = \alpha_{ph1}\mu_{ph1} + \alpha_{ph2}\mu_{ph2}} \quad (3.42)$$

Solution of the volume fraction equation. As previously stated, in this work, the VOF method is used to capture the interfaces. Among the alternatives, the level-set method is very popular due to its accuracy of the curvature computation, but suffer from poor mass conservation (Menon, Sankar; Nilsson, 2016; Osher & Sethian, 1988; Sussman et al., 1994). While the VOF method is inherently mass conservative, its main difficulties lie in maintaining sharp interfaces due to numerical diffusion arising from the solution of the advection equation (Wacławczyk, 2017). Traditionally, two different approaches exist to tackle this issue, often referred to as interface-reconstruction (geometrical) and interface-sharpening (algebraic) methods. In the former, the interface is first reconstructed using a reconstruction algorithm as e.g. simple line interface calculation (SLIC) (Noh & Woodward, 1976) or piecewise linear interface calculation (PLIC) (YOUNGS, 1982), followed by an advection of the

interface. These methods are generally very complex, especially in three dimensions, and are relatively computationally. The latter however, involves sharpening of the interface through the solution of the advection equation by the use of discretization schemes intended to reduce the numerical diffusion (Nguyen & Park, 2017). Such an approach was used in this work by the means of the *Weller-VOF method* (OpenCFD, 2021), implemented in the *interFoam*(S. M. Damián., 2013; Deshpande et al., 2012) family of solvers in OpenFOAM.

Following previous discretization methodology, Eq. 3.40a can be discretized with the forward Euler scheme as

$$\frac{\alpha_o^{n+1} - \alpha_o^n}{\Delta t} V_c + \sum_f \alpha_{of}^n F^n + \sum_f \alpha_{of}^n \alpha_{wf}^n F_c^n = 0 \tag{3.43}$$

where F and F_c are fluxes defined as $F = S \cdot u_{ff}$ and $F_c = S \cdot u_{cf}$. The compressive flux is given as:

$$\boldsymbol{F_c = n_f \min[C_{\alpha_o} \frac{|F|}{|S_f|}, \max(\frac{|F|}{|S_f|})]} \tag{3.44}$$

$$\boldsymbol{n_f = \frac{(\alpha_o)_f}{|(\alpha_o)_f|} \cdot S_f} \tag{3.45}$$

where C_{α_o} is a user-defined constant ($C_{\alpha_o} = 1$ throughout this work).

As previously stated, excessive numerical diffusion at the interface is problematic and requires with an algebraic method, an intricate interpolation treatment of the face values α_{ow} in order to obtain a good combination of boundedness, accuracy and convergence. For this, the *interFoam* solvers uses the *Multidimensional Universal Limiter with Explicit Solution* (MULES) solver, which uses a limiter defined as 1 and 0 close to, and distant from the interface respectively (Deshpande et al., 2012). The limiter ensures that a high order scheme is used at the interface, while employing a simple bounded upwind scheme elsewhere. The high order scheme is composed as a blend of central and upwind ones, evaluated using a limiter function based on total variation diminishing (TVD) to avoid oscillations of the solution (Santiago Márquez Damián, 2009; Deshpande et al., 2012).

CHAPTER 4 - MODEL SETUP AND BOUNDARY CONDITIONS

In this chapter, the initial and boundary conditions for the CFD-DEM system are defined. The four different numerical model setups were constructed to mimic laboratory experiments and reservoir conditions of the Kazakhstan oilfield.

The cylindrical geometry shape of the numerical sample used (Figure 4.2) in this study was created by analogy with the laboratory sample of Kozhagulova et al. (2021)(Figure 4.1). The size of the numerical sample is much smaller than in the experimental study due to the computational complexity of DEM, which in turn is strongly depends on the number of particles in the simulation.

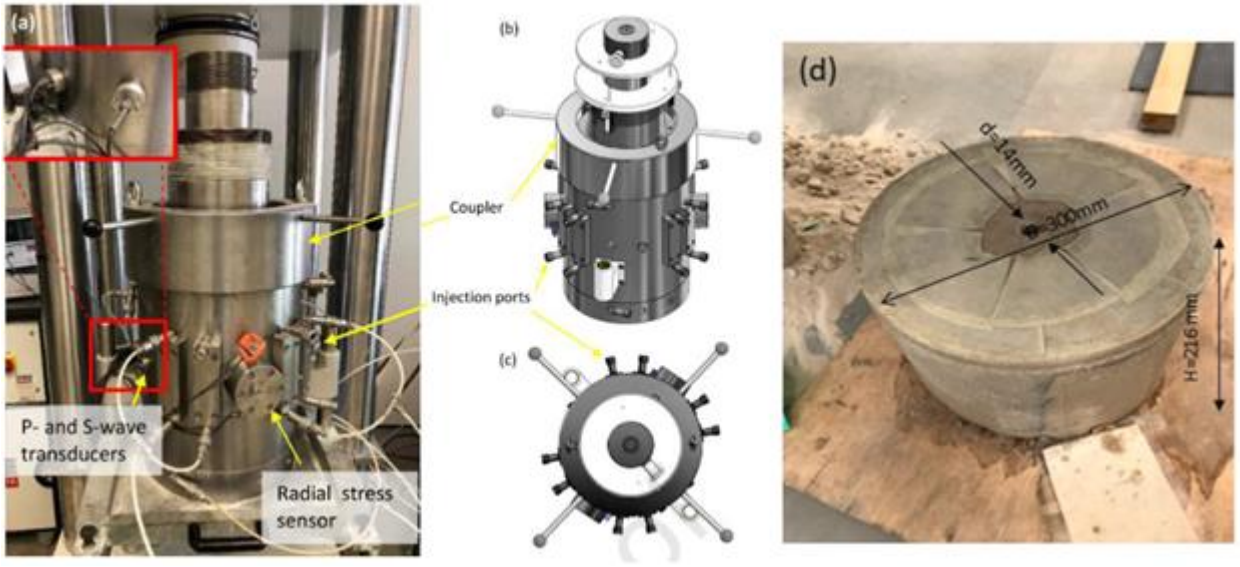


Figure 4.1. High pressure consolidation cell with top cap: (a) General view, (b) Schematic orthogonal view, (c) Schematic top view and (d) Experiment sample at the final stage (Kozhagulova et al., 2020b, 2021)

In the CFD-DEM model, the DEM and CFD computations are coupled by exchange terms. The shape and geometry of DEM and CFD computational domains are similar (Figure 4.2). The compressed particles with determined overburden stress inside the cylinder under the influence of forces able to move toward the central hole and delete after the border crossing. The initial and outlet pressure are the same at the initial time. The outlet pressure further is maintained at $p = const$. The fluid is injected from the side wall radially to the center of the sample with $v = const$.

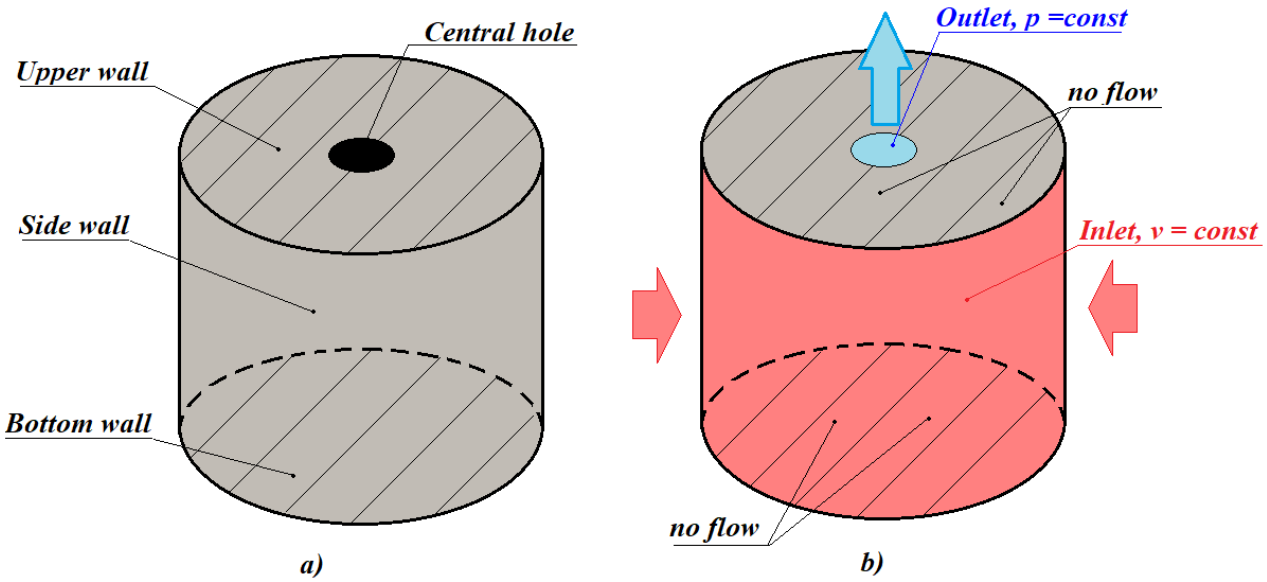


Figure 4.2. The CFD-DEM model setup: a) Surfaces; b) CFD boundary conditions

In this research, the one-phase and two-phase fluid flow models were developed. The fluid was considered incompressible with the constant density and the kinematic viscosity. The **Upper** and **Bottom** walls were set as impermeable walls. At these walls velocity of the fluid is equal to zero. For DEM parts, the **Side** wall is also impermeable. The central hole in the top wall is open to serve as the fluid outlet. All initial and boundary conditions of sand production numerical experiments are tabulated in Table 4-1.

Table 4-1. A summary of the CFD-DEM boundary conditions

	DEM	**CFD**			
Set	**Overburden stress, MPa**	**Inlet velocity, m/s**	**Outlet pressure, MPa**	**Fluid model**	**Fluid composition**
I	1	10^{-5}	1	One-phase	water
II	1	$5 \cdot 10^{-5}$	1	One-phase	heavy oil; light oil
III	3	$1.39 \cdot 10^{-2}$	1	Two-phase	gas-water
IV	5.9	$2.83 \cdot 10^{-2}$	4.5	Two-phase	water-heavy oil; gas-heavy oil

The setups $I - III$ are followed the laboratory experiment conditions (Kozhagulova et al., 2021; A. D. Shabdirova et al., 2020; Shabdirova Ainash et al., 2021). The dry compressed sample with average overburden pressure $1,3\ MPa$, while setup IV was built by analogy with Object 2 of Karazhanbas oilfield with overburden stress gradient is $20.6\ kPa/m$ (Collins et al., 2008).

The CFD-DEM boundary conditions (Table 4-1) improves the previous numerical studies(O'Connor et al., 1997; Oryem et al., 2016; A. Rakhimzhanova et al., 2019; A K Rakhimzhanova et al., 2018; Aigerim K Rakhimzhanova et al., 2020) in terms of a more comparable

cylindrical geometry, a larger simulation scale, different flow directions and of the multiphase flow simulations. A cylindrical geometry was selected for both the solid particle and fluid phases to be comparable to the experimental study of the same material (Kozhagulova et al., 2020b; A. D. Shabdirova et al., 2020). The radial fluid flow was injected towards the central vertical perforation, and discrete particles were produced with the flow dependent on the applied solid stress and fluid pressure conditions.

CHAPTER 5 - MICROSCOPIC BEHAVIOR OF PERFORATION DAMAGE AND SAND PRODUCTION IN SINGLE FLUID FLOW

This Chapter outlines the bond breakage behaviour investigated by coupled microscopic CFD-DEM method. The perforation and sand production simulations in a weak cemented sandstone were conducted by using a high-performance computer system. The perforation process creates a damage area which contains compressed sample of non-cemented particles. The compacted after perforation procedure sample has been quickly expanded under one-phase fluid flow condition and has released its unbonded particles to transport to the perforation channel.

5.1 Introduction

The Introduction section describes three-dimensional CFD-DEM simulations that were conducted by the open-source CFDEM program with a large numerical sample of 10^5 particles. The WCM has been executed in LIGGGHTS and coupled with the OpenFOAM. The numerical studies by Rakhimzhanova et al. (2018, 2019, 2020) were improved in terms of more equivalent cylindrical geometry instead of the parallelepiped geometry used in the studies earlier. The ongoing research encloses different method with usage of greater simulation scale, changed fluid stream directions and multiphase flow simulation (see Chapter 7), as well as the detailed scientific visualization similar to Climent *et al.* (2014) in order to deliver better perception into the entire sanding processes. Based on this conception, the development of a plastic zone according to perforation and the way it progresses during sand production, was investigated. A cylindrical geometry was chosen for both solid particle and fluid phases to be corresponding to the experimental study of the same material (Kozhagulova et al., 2020b; A. D. Shabdirova et al., 2020). The radial water flow was injected towards the central vertical perforation and sand particles were extracted with the one phase. The formation of a perforation damage area and how it progressed throughout the sand production process were embraced in the simulation and presented in this Section.

5.2 DEM simulations and results

The entire theoretical formulations of DEM are described in Chapter 3.1. This study depicts similar contact model by Rakhimzhanova et al. (2018) for the analogous sandstone material has been adapted and implemented in LIGGGHTS. The development has been executed in two stages: first to implement the original JKR model since it was not existed in LIGGGHTS and afterwards the new bond breakage form was introduced at point C. As a result, the transformation from the modified JKR model to the Hertz model occurred as soon as the bond breakage event happened. The implementation

process was executed successfully and simulation outcomes were verified with theoretical results (Khamitov et al., 2019). In those solutions previous work was upgraded further by applying the CFD-DEM simulation of the new contact model to simulate the sand production from the weak sandstone.

The coupled CFD-DEM formulation for one-phase fluid in Chapter 3.2 has been applied as ***Model A*** in the CFDEM Program. The CFDEM program is a highly parallelized computational program that is combines the LIGGGHTS program for the DEM modelling and the OpenFOAM Program for the CFD modelling (Christoph Goniva et al., 2012). All simulations in this Section were run on an HPC supplied by Intel Xeon® E5-2699 Version **4** with the **22** core. The clock speed of **2.20** GHz and **500** gigabytes of memory. The usual simulation could actually take more than a week.

The simulations in this study were performed in order to explore the sand production behaviour of a weak sandstone as part of a bigger experimental and simulation studies project to comprehend the significant sand production issue in some certain shallow oil reservoirs in Kazakhstan. The simulations goal was to deliver supplementary microscopic information to the experimental study (Kozhagulova et al., 2021), which has been delivered in the laboratory by the consecutive steps of consolidation, cementation, perforation and sand production to imitate what had occurred in the oil fields.

Alike to the practice in the laboratory, the entire simulation consists of the subsequent stages of particle generation and pluvial settling, consolidation, cementation, perforation and sand production

5.2.1 Initial sample preparation

The numerical model preparation method is composed of three key stages: particle generation and pluvial settling, consolidation, and cementation. The numerical particles were created randomly inside of a cylindrical wall and fell under gravity to the bottom wall of the space to simulate a pluvial deposition procedure. The number and the size of the particles were identified similar to the particle size distribution of a natural sandstone from the Ustyurt-Buzachi Sedimentary Basin (A. D. Shabdirova et al., 2016). A total of one hundred thousand frictional elastic auto-adhesive particles were produced to generate the numerical sample which is equal to a total mass of 1.57 g. Summary of the particle information is outlined in the Table 5-1, and the numerical particle size distribution is assimilated to the natural material and shown in Figure 5.1

Table 5-1. A characteristic of eight particles groups in the sample

Diameter of particle (mm)	**0.15**	**0.18**	**0.2**	**0.22**	**0.25**	**0.275**	**0.3**	**0.355**
Particle number in group	20754	15525	13346	13821	17510	9922	5418	3704
Mass ratio (%)	5.8	7.5	8.8	12.16	22.64	17.05	12.1	13.95
Particle mass (mg)	91.7	118.5	139.8	192.6	358.1	270.1	191.5	216.9

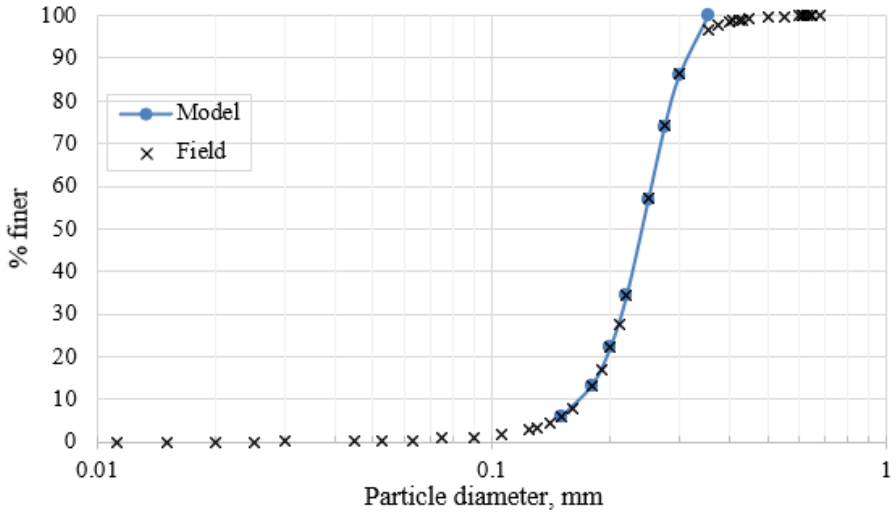


Figure 5.1. PSD of the numerical and natural sandstone samples

The micromechanical parameters for the DEM simulations were assimilated from the numerical studies of weak sandstones behaviour Rakhimzhanova *et al.* (2019) and are provided in the Table 5-2. Contributions' roles are outlined in the numerical formulation in the Section 3.

Table 5-2. Mechanical properties of particles in simulations

Friction coefficient (μ_c)	0.3
Cohesion energy (γ, J/m^2)	40
Particle density (ρ_p, kg/m^3)	2500
Poisson ratio (v)	0.3
Young's modulus (Y, Pa)	$7 * 10^{10}$

To simulate the overburden stress in the field, the deposited particles are compressed vertically by a Top wall while surrounded and constrained by the cylindrical wall. This approach simulates the one-dimensional compression form of the consolidation of the reservoir material. The JKR model was utilized for the particle creation and pluvial deposition, consolidation processes so that particles were sticked together, and a porous sample could be composed as in the field with an initial porosity of approximately 35%. As soon as the vertical stress was equivalent to the target value of 1 MPa, the consolidation simulation was finalized. The cementation of the numerical sample was activated as the contact model was switched from the original JKR (with D point) model to the modified JKR model (with C point). From now on and further, new formed contacts will be unbonded as specified by Hertz's model. The previous bond contacts are able to break, but new bond cannot be formed. Since the activation of modified JKR, a certain number of bonds had broken while passing the point

C in Figure 3.2. The numerical model was permitted to reach a new equilibrium and the sample preparation has been taken as complete. The initial sample after preparation is shown in the Figure 5.2a. The diameter (*D*) of the intact model is identical to $15.12\ mm$, and the height (*H*) is equal to $6.47 mm$. The intact sample was then exposed to perforation and sand production that create damages to its initial structure. This process was outlined in the microscopic simulation.

5.2.2 Sample perforation

The intact model was perforated to produce an escape path for the liquid flow similarly to the approach in the oil well, which was modelled as the penetration of a penetrometer with an octagonal cross-section as shown in the Figure 5.2b. The proportions of the penetrometer are: $\boldsymbol{r = 0.7\ mm, s = 0.579\ mm, d_1 = 10\ mm, d_2 = 1\ mm}$.

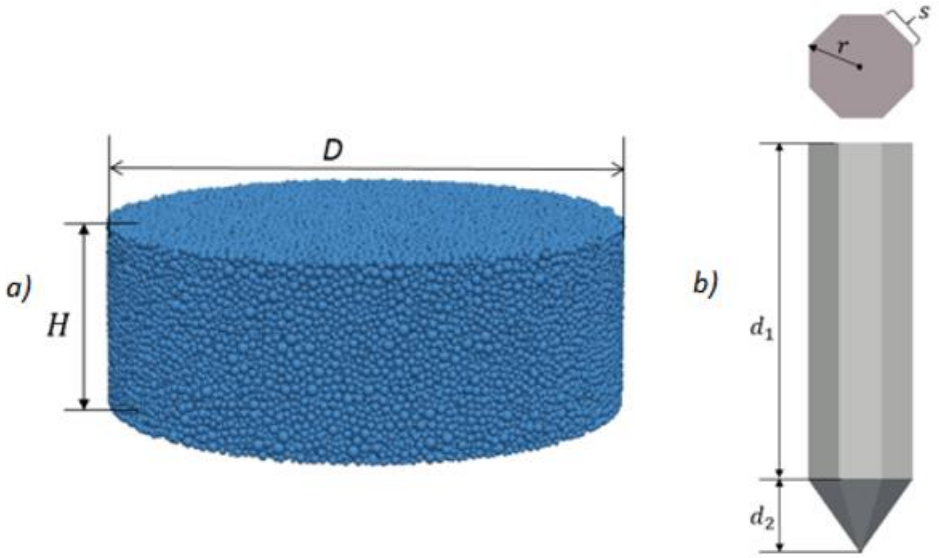


Figure 5.2. The geometry of numerical sample and penetrometer. a) The sample compressed by vertical stress equals to 1 MPa. b) The geometry of the penetrometer

The cylindrical model was divided into eleven annular cylinders with the identical thickness as the radius (*r*) of the penetrometer to control and depict the internal structure evolution of these various zones as a role of the radial distance from the centre of the sample. Annular cylinders in the Figure 5.3 were used for picturing and collecting information only, and it was not comprised in the computation.

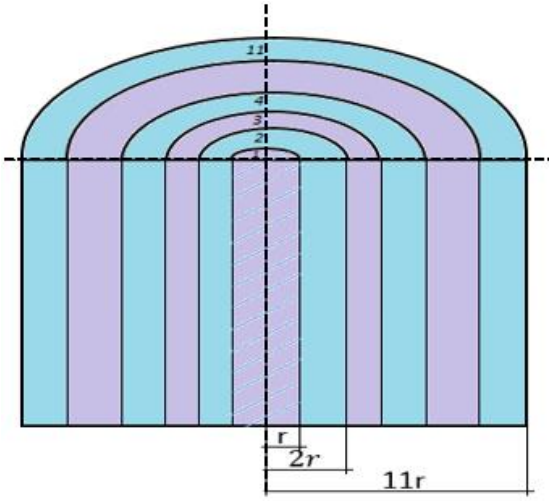


Figure 5.3. Schematic representation of the cylindrical sample segmentation for visualisation and information gathering

The vertical penetrometer moved lengthways the axis of the cylindrical specimen. Even though it is not shown in the Figure 5.4, the top wall used to compress the sample in previous step was fixed firmly to maintain the applied vertical stress and avoid particles from escaping the specimen. This is analogous to the experimental condition, where the perforation was formed through a central launch in the top cap (Kozhagulova et al., 2021). There was no communication between the top wall and the penetrometer in the simulation. The sample was perforated to the full depth and then the penetrometer was taken out as shown in the Figure 5.4. The penetrometer velocity in both stages was set as $12\ m/s$, while the time step of the DEM simulation was set at $7 \cdot 10^{-10}\ s$. The sample and penetrometer sizes was chosen analogous to Oryem et al. (2016) and O'Connor et al. (1997). The key criteria was the sample/penetrometer (equal 10.8) and penetrometer /particle ($3.94 \div 9.33$) ratio. Climent *et al.* (2014) utilized the averaging volumes to calculate stresses and the width of the rings that were selected in order to make each ring to have sufficient number of particles that is statistically significant stress value as could be computed. In this study the width of the rings (Figure 5.3) is equivalent to the penetrometer radius.

The color of the elements in the Figure 5.4 was encoded pursuant to the average number of bonds in each annular cylinder. Here, the average bond (N_b^i), the average contact (N_c^i), and the average contact total (N_t^i) numbers in an i^{th} annular cylinder are defined as:

$$N_b^i = \frac{\sum_{j=1}^{k_i} N_b^j}{k_i} \tag{5.1}$$

$$N_c^i = \frac{\sum_{j=1}^{k_i} N_c^j}{k_i} \tag{5.2}$$

$$N_t^i = N_b^i + N_c^i \tag{5.3}$$

where N_b^j and N_c^j are the number of bond and the number of contacts of a j^{th} particle in the i^{th} annular cylinder. The summary goes through the total number of particles (k_i) in the i^{th} cylinder, where $i = \overline{1 \div 10}$. It should be noted, that the modified JKR contacts were calculated as bonds, all of which were established before the perforation. The Hertz contacts formed after bond breakage were counted as contacts. A particle can have different contacts that contain both bonded (N_b) and unbonded contacts (N_c).

The maximum bond number has been perceived in the Figure 5.4a at the beginning of the perforation, where the single red color code shows the similarity of the intact sample. During the penetration progress the number of bonds fell quickly around the penetrometer. Contemporaneously, the materials nearby the boundary stayed in their intact state in terms of the maximum bond number. When the penetrometer was wholly extracted, as in Figure 5.4e, a damage zone of about zero bonded particles was formed within a radial distance of $3r$ from the axis of the sample. A larger zone of up to about $7r$ was identified around the penetrometer, where the particles in this zone lost certain bonds throughout the perforation process.

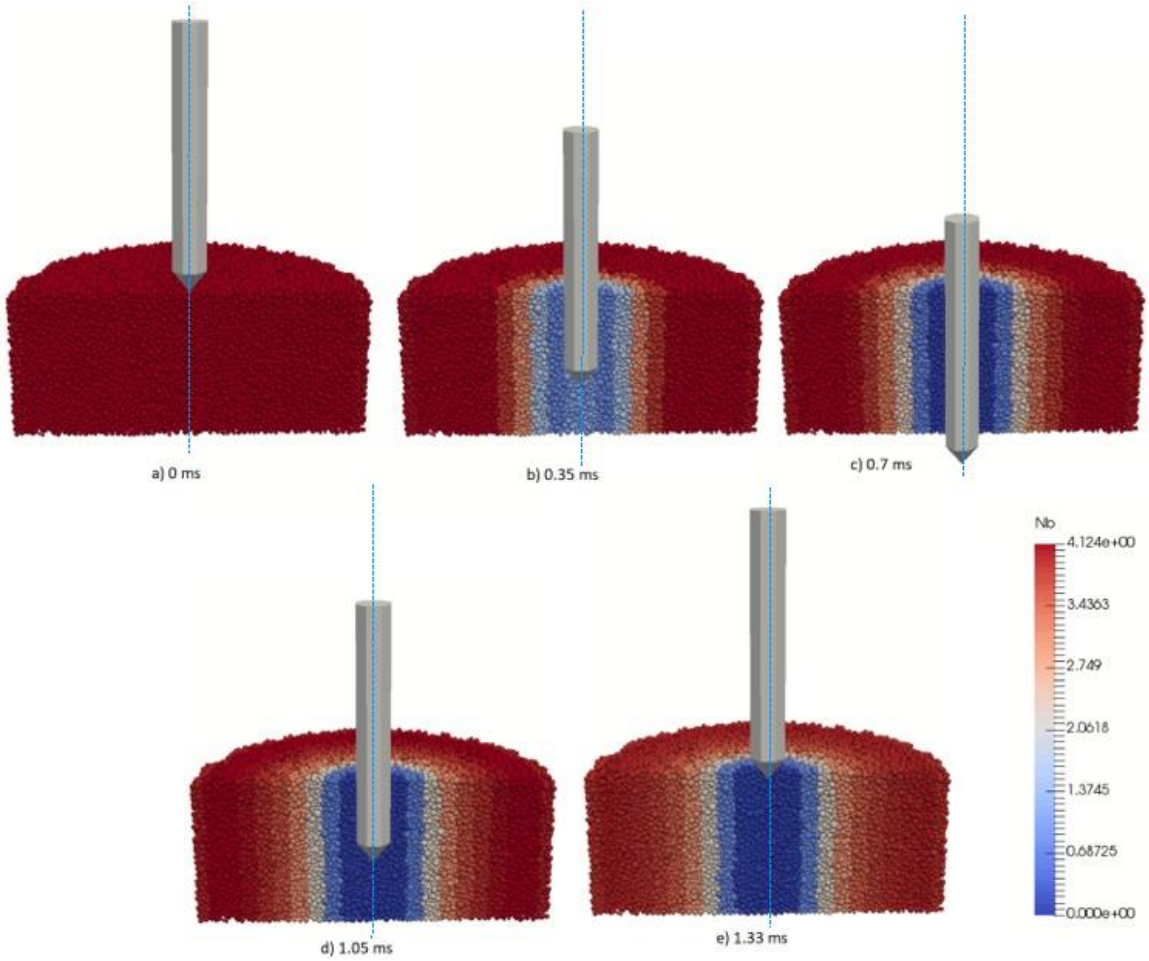


Figure 5.4. The bond number (N_b) behavior during $3D$ perforation simulation

The average bond and the average contact numbers were plotted in the Figure 5.5 as a function of the radial distance from the axis of the numerical sample. The time given in the legend relates to the different stages of the perforation procedure in the Figure 5.4. Note that the green lines start at the distance of $2r$, while the other curves start at the distance r. The reason is that the green lines is consistent with the full penetration as in the Figure 5.4c, and there was no particle within the distance r, this stage as the penetrometer occupied it.

The values of N_b and N_c are nearly constant in the intact state. The small values of N_c were formed between the two stages of consolidation and cementation when the modified JKR model was activated during the sample preparation process. N_b drops to values near to zero at $2r$ and is less than 1 at $3r$ at the end of perforation. N_c risen significantly in this zone when particles created new contacts as they reorganized after the bonds were broken. However, the development of new contacts is less than the breakage of bonds, and therefore the total contact number N_t is less than the value in the intact state.

New contacts were not developed from the distance $7r$ and further. Nevertheless, the bond number falls about 10% (equal to a decrease of about 0.4 in magnitude) in this zone, which could mean that the bond breakage is not appropriate to trigger significant particle rearrangement to create new contacts. As bond breakage was not convertible, it represents plastic deformation. Particles, however, do not rearrange, and the strain could be very small. One would argue that the materials between $7r$ and the boundary could be in the elastic (the term "elastic" is used in order to differentiate the plastic zone with the non-plastic one) state in a less precise sense, and the distance $7r$ from the sample's center can be considered as the interface between the plastic zone and the elastic zone that surround the perforation. The zone until $3r$, per contra, can be considered the zone of perforation damage, where the interparticle bonds were typically destructed, leading to noteworthy particle rearrangement.

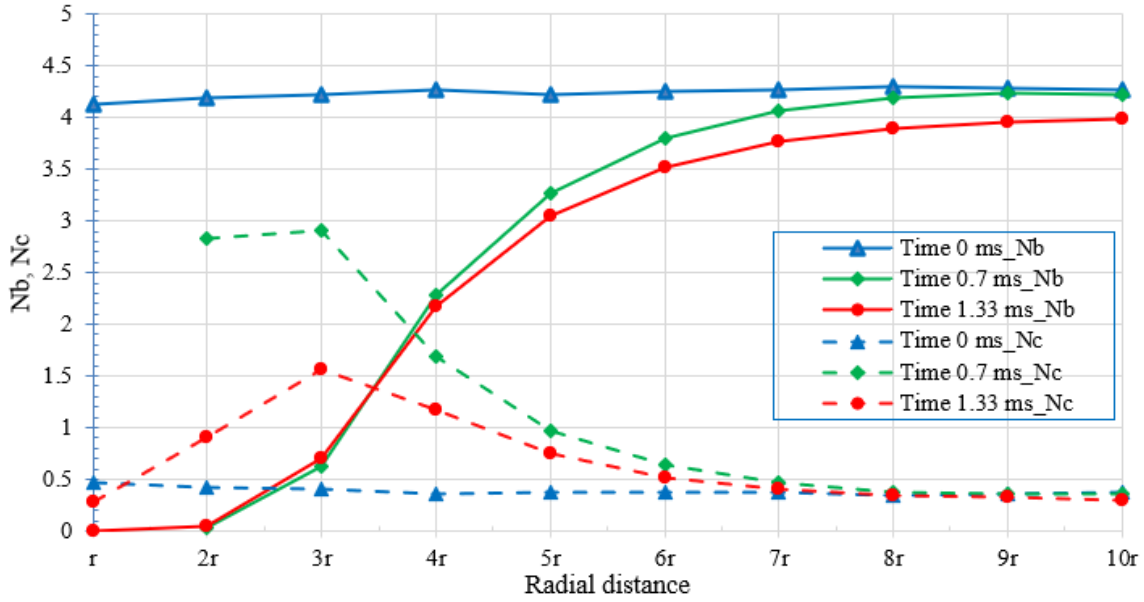


Figure 5.5. The average bond and contact numbers (N_b, N_c) versus radial distance during $3D$ perforation simulation

The sample was formed with particles of eight templates. The bond number N_b was calculated individually for each particle size (see Table 5-1). Overall, larger particles formed more contact and bond with nearby particles than the smaller particles. Yet, it is not still clear which particle size is more likely to bond breakage. This knowledge would be valuable in the later stage of sand production since the broken particles could be eroded very easy by the hydrodynamical forces from the fluid flow. The change in the N_b number can be counted as:

$$\Delta N_b(\%) = \left(\frac{(N_b)_t - (N_b)_{t0}}{(N_b)_{t0}}\right) * 100\% \quad (5.4)$$

where $(N_b)_t$ is the N_b value at time t; t -current time; and $t0$ -the time at the start of the perforation process in the Figure 5.4a. The outcomes of N_b and ΔN_b were calculated for the three stages of Figure 5.4a, 6c and 6e and tabulated in Table 5-3. At the initial state, the biggest particle size has in average 7.65 bonds, that is more than double the number of the smallest particle size ($\sim$3.14). At the end of perforation (Figure 5.4e), each particle size lost from 17% to 20% of its bonds, where the biggest number of breakage occurred between the initial state and the full penetration state in the Figure 5.4c. There is no significant difference in the bond breakage behaviour among various particle sizes. The largest particles still have double more average bond numbers than the smallest particles at the end of perforation.

Table 5-3. The change of bond number (ΔN_b) and bond number (N_b) distributions

Time step	***Diameter* (*mm*)**	**0.15**	**0.18**	**0.2**	**0.22**	**0.25**	**0.275**	**0.3**	**0.355**
0 *ms*	N_b	3.14	3.50	3.78	4.13	4.76	5.35	6.03	7.65
	ΔN_b (%)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.7 *ms*	N_b	2.72	3.01	3.25	3.55	4.09	4.61	5.22	6.70
	ΔN_b (%)	−13.25	−13.78	−14.12	−14.10	−14.06	−13.84	−13.53	−12.45
1.33 *ms*	N_b	2.59	2.84	3.04	3.31	3.81	4.28	4.81	6.19
	ΔN_b (%)	−17.38	−18.74	−19.65	−19.83	−20.06	−20.11	−20.22	−19.08

The permeability of the material is directly dependant on the porosity. The permeability of the plastic zone is a contribution that was discovered to significantly impact of the sand prediction using an analytical model (A. Shabdirova et al., 2019). The porosity was defined for the annular cylinders in the Figure 5.3. The total volume of particles V_p^i, located within each annular cylinder can be calculated as follows:

$$V_p^i = \sum_{j=1}^{m_i} V_j \quad (5.5)$$

, where V_j is the volume of the individual particle; m_i - total number of particles in the i^{th} cylinder, $i = \overline{1 \div 10}$. The porosity of the i^{th} cylinder (ϕ_i) can be defined as:

$$\phi_i = \frac{V_i - V_p^i}{V_i} \quad (5.6)$$

, where V_i is the volume of the i^{th} annular cylinder, which is the similar for all cylindrical regions.

As the penetrometer was pushed into the sample, particles in the surroundings were forced out radially. A compacted zone was developed within the distances of $2r$ to $6r$. When the penetrometer was removed, these zones slightly grew and were shown in the lighter reddish colours. The most compressed zone of the reduced porosity was between $2r$ to $3r$. Versus, a loose zone of unbonded particles was created in the lower half of the cavity left by the penetrometer. Note that in the zone of less than $3r$ radial distance, most of the bonds have been damaged (see Figure 5.5). The major difference in the porosity value depicts particle rearrangement and big deformation, which is aligned with the results of bond breakage in the previous part. The zone from $6r$ to the boundary did not illustrate any noteworthy change in the porosity. The external ring (dark blue) illustrated an anomaly because of the effect of the boundary condition.

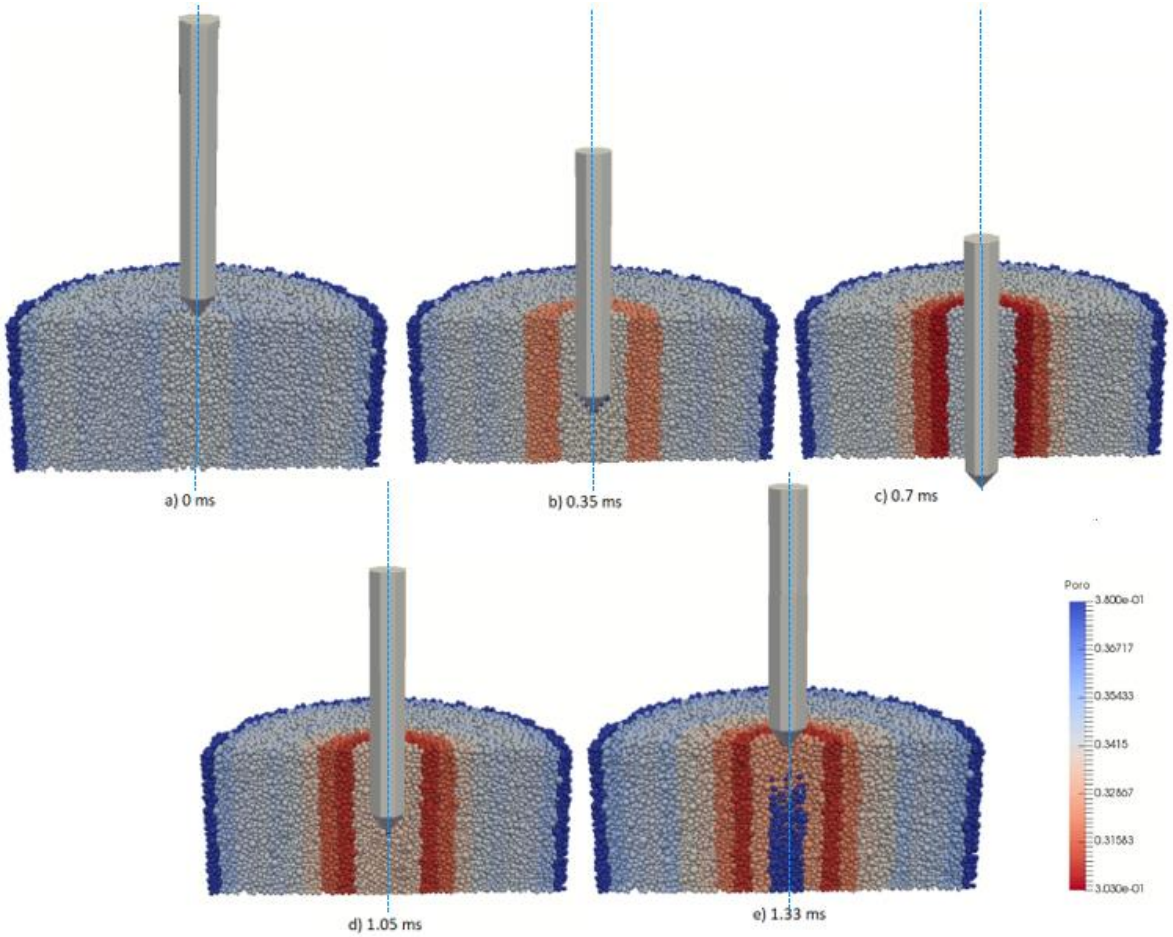


Figure 5.6. Porosity distribution during $3D$ perforation simulation. The colour indicates the porosity values

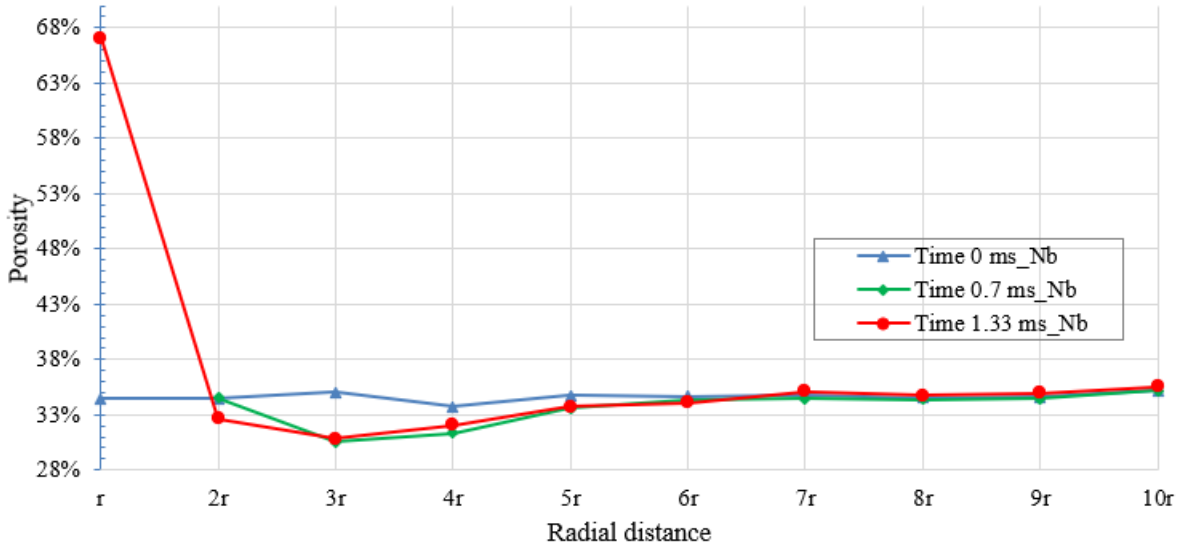


Figure 5.7. Porosity distribution versus radial distance

The radial porosity distributions while perforation are provided in the Figure 5.7. Similar to Figure 5.5, the green line indicating the full penetration state is began at the distance of $2r$ same as the space below r was captured by the penetrometer. The porosity distributions from $6r$ snd further do not illustrate any considerable change during the whole perforation procedure. If porosity change is deemed as the criteria that segregate the plastic and the elastic zones, then the boundary between them can be defined as $6r$ in the Figure 5.7, which is smaller than the interface at $7r$ in the Figure 5.5. The modest changes in the bond number and the contact number in the Figure 5.5 were not interpreted to a visible change in the porosity. The major porosity decrease has been gained at $3r$, the similar distance of the maximum particle rearrangement in the Figure 5.5. The porosity of the loose material inside the cavity upon perforation has reached to 65%, while the porosity in the elastic zone stays the same as the initial porosity. A more quantitative analysis of the change in porosity is provided in Table 5-4, the values determined similarly to Eq. 5.4. It is proved that the maximum porosity decreases in the surrounding regions of the penetrometer happened at $3r$. While the removal of the penetrometer, the materials in the zone until $2r$ grow significantly, while it is much less significant for the zone from $3r$ and further. Building of a zone of lower permeability at $3r$ would impact the fluid flow in the sand production stage. The change of the porosity in the zone between $6r$ and $10r$ can be deemed as insignificant.

Table 5-4. Porosity change ($\Delta\phi_i$) during perforation test

Time step	**Radial distance**	r	$2r$	$3r$	$4r$	$5r$	$6r$	$7r$	$8r$	$9r$	$10r$
$0.7\ ms$	$\Delta\phi_i$ (%)		0.2	−12.9	−7.1	−3.3	−1.1	−1.0	−0.4	−0.5	0.0
$1.33\ ms$	$\Delta\phi_i$ (%)	94.8	−5.2	−12.0	−5.0	−2.9	−1.8	0.7	0.6	0.9	1.0

The plastic zone properties like porosity, permeability, might be considerably different from the elastic zone properties (Daigle et al., 2017; Risnes et al., 1982b). The porosity values (Table 5-4) of the plastic zones were transferred into permeabilities and evaluated for the permeability from the semi-analytical outcomes based on the experimental study (A. D. Shabdirova et al., 2020). The permeability decrease templates of numerical model was found in good agreement with experimental studies (Figure 5.8). In order to qualitatively compare the results across different boundary conditions between the two studies the normalization was performed.

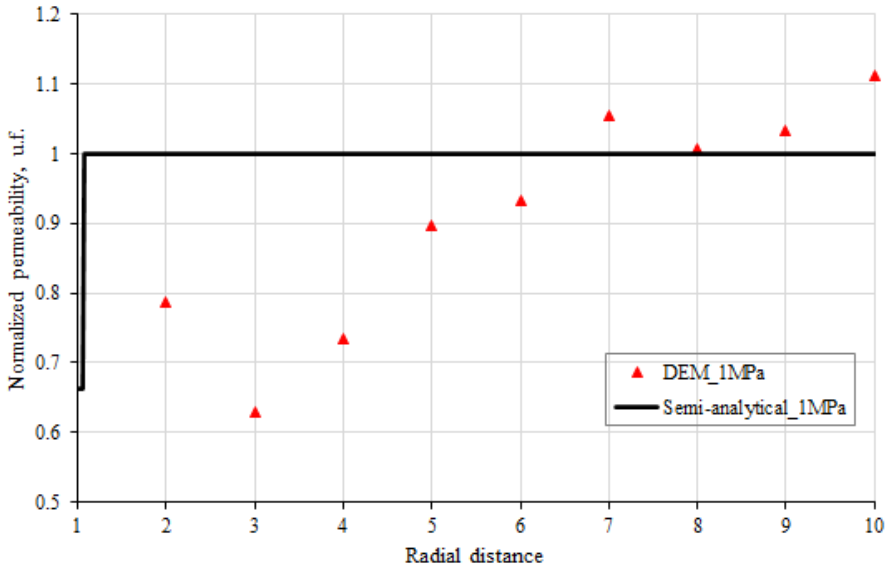


Figure 5.8. Comparison of the permeabilities obtained from the perforation test and the analytical model (A. D. Shabdirova et al., 2020)

5.3 Coupled simulations and sand production results

The loading conditions in the simulations were conducted analogously to the experimental studies of the same sandstone material in the experimental procedure (Kozhagulova et al., 2020b), which has been executed using a High-Pressure Consolidation System (HPCS). The general and schematic picture of the experimental system are given in Figure 2.7 and experimental sample is shown in Figure

5.9 The vertical compression, in-situ cementation, and sand production activities in the experiment are strictly simulated in this numerical simulation. Their relevant findings are compared in this study.

Figure 5.9. Experimental sample after sand production (Kozhagulova et al., 2021)

In this study, the numerical sample (Figure 5.2) was created by analogy with laboratory study (Figure 5.9) with cylindrical shape for both sample and penetrometer. Since DEM computational complexity depends on a number of particles, this number was carefully choosen to optimize calculation time and not lose physical similarity. In the first simplification, the height and diameter of the laboratory sample were decreased in 10 times, but it was required approximately $\approx 1.31 \cdot 10^9$ particles, which was computationally expensive. Finally, diameter of the numerical sample decreased to $15\ mm$. The numerical sample contains 10^5 particles (Table 5-5).

Table 5-5. Numerical and laboratory samples geometry characteristics

Variable (unit)	**Laboratory** (Kozhagulova et al., 2020a)	**Numerical sample**
Geometry of sample	Cylinder	Cylinder
Height, mm	216	6.471
Penetrometer shape	cone-shaped	cone-shaped
Penetrometer diameter, mm	14	1.4
Sample diameter, mm	300	15.12
Volume, $mm3$	$1526.8 \cdot 10^4$	1161.8
Number of particles	$\approx 1.31 \cdot 10^9$	10^5

The sample/penetrometer ratio in this work is about 10.8. The cylindrical sample was artificially divided into annular cylinders with depth equal to the penetrometer radius. The 11 rings are just introduced as a convenient way to monitor the distributions and variations of various parameters during perforation and sanding. They do not affect the accuracy of the simulation**.**

5.3.1 CFD-DEM coupled simulation

The perforated sample in the Section 5.2 was used for sand production simulation. Water was injected radially from the external boundary towards the vertical perforation tunnel at the center, similar to the experimental conditions.

The water phase was considered incompressible with the density $\boldsymbol{\rho_w = 10^3\ kg/m^3}$ and the kinematic viscosity $\boldsymbol{\mu_w = 10^{-6}\ m^2/s}$. The CFD computational domain in Figure 5.10 has $\mathbf{3920}$ cells that form a cylinder of the same size as the initial sample in Figure 5.2a. The boundary conditions are illustrated in Figure 5.10 (Table 4-1, set $\boldsymbol{I}$). The top and the bottom walls were set as impermeable walls with no-slip boundary conditions. There was a central opening on the top wall to serve as the fluid outlet. Water was injected from the whole circumferential wall of the cylinder at a constant velocity $\boldsymbol{v = 10^{-5}\ m/s}$. The outlet pressure was maintained at a constant $\boldsymbol{P = 1\ MPa}$.

The interactions between the fluid and the particle phases were simulated by a four-way coupling method (Rahmati et al., 2013). The motion of particles affects the motion of fluid and vice versa. The coupling information was given in terms of the pressure force, the viscous force, and the Di Felice drag force (Di Felice, 1994; Zhou et al., 2010), as explained in Eqs. 3.3, 3.4 and Eq. 3.18.

The DEM time step is $10^{-8}\ s$ that is 5.18% from Rayleigh time $dt_r = 1.93 \cdot 10^{-7}$(see Eq.3.9). The CFD time is $10^{-6}\ s$, under which a maximum Courant number of 0.0167. Similar to Goniva *et al.* (2012), the time-step size for the CFD solver has been chosen to be an integer multiple of the DEM time-step. This is to save computational time, i.e., the CFD simulation was run for every 100 time steps of the DEM simulation, and the coupling data was exchanged accordingly. The affect of coupling frequency on the efficiency and accuracy of CFD-DEM results will be shown in Chapter 7.

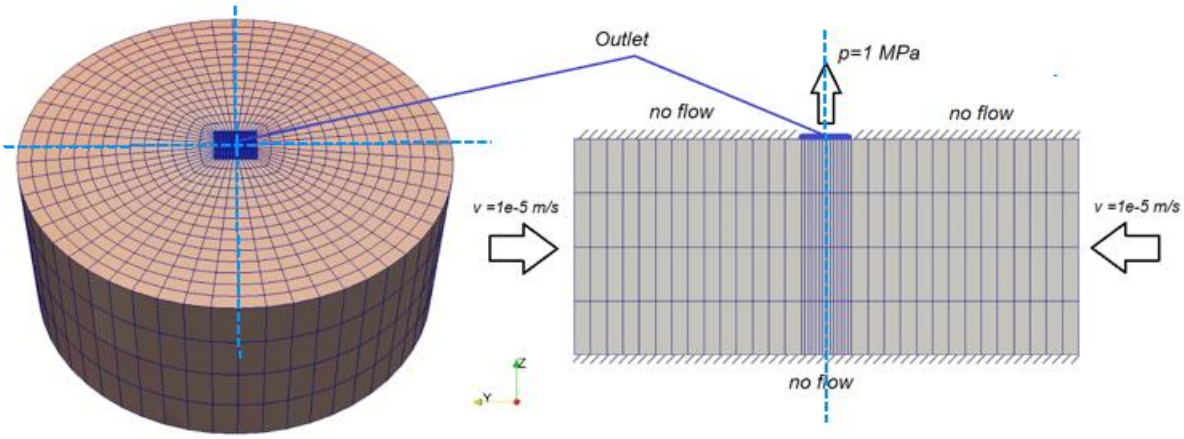


Figure 5.10. CFD simulation domain and boundary conditions

The coupled simulation last for more than 10 ms, throughout which there were a total of 19 particles that were extracted together with the leakage of water from the sample. The discrete events of sand production and the cumulative sand mass are provided in the Figure 5.11. There was no sand extracted after 7.5 ms, and the total produced sand mass was about 0.23 mg, which is 1.47 % of the total mass M_t. The sand production behaviour can be described as a two-stage procedure, where the sanding events were more frequent in the first stage when the time was less than 2.5 ms. In the second stage between 2 ms and 8 ms the sanding events were less frequent, and they were related to the production of larger particles represented by the higher bars and the bigger sand masses in the Figure 5.11. The sanding phenomenon commonly reduced with time passing, which can be defined as a transient sand production (Veeken et al., 1991).

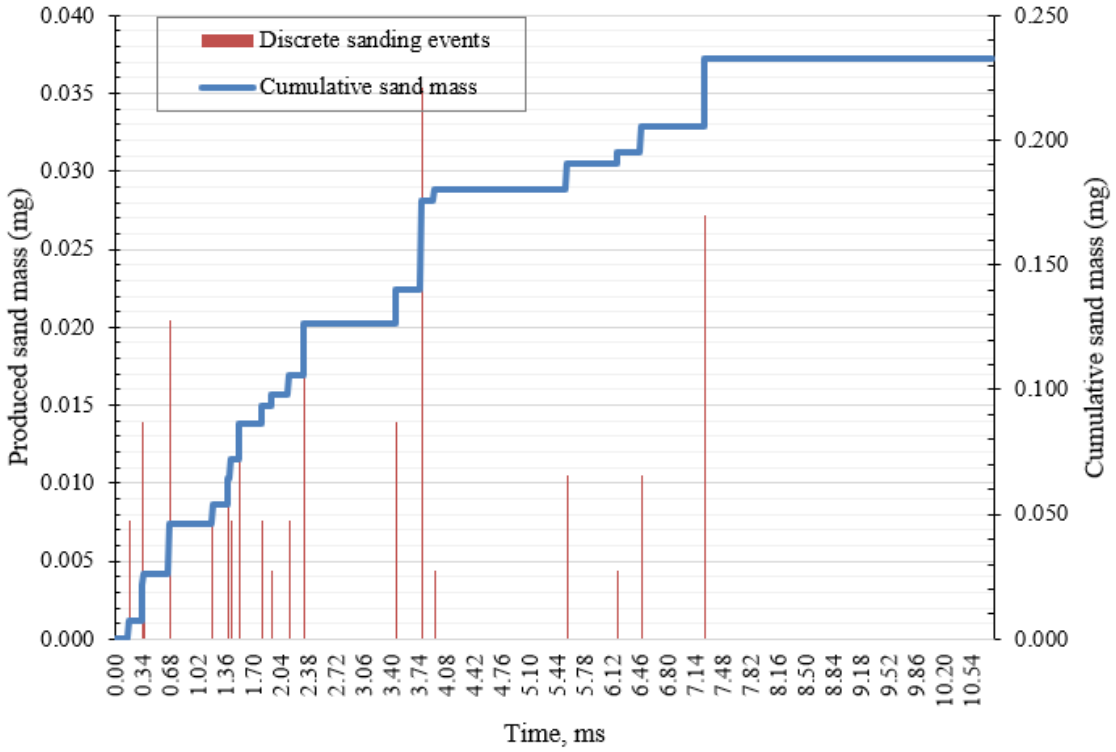


Figure 5.11. The produced and cumulative sand mass

The sand production outcomes in the Figure 5.11 can be divided into the production of various particle sizes in the Figure 5.12. The data were plotted for two different time periods of the first stage of frequent sand production and the second stage of a small sand production cases based on the results in the Figure 5.11. The five smaller particles were initially extracted, while the sixth and seventh biggest particles were extracted in the second stage. None of the eighth largest particles were produced during the entire sand production simulation.

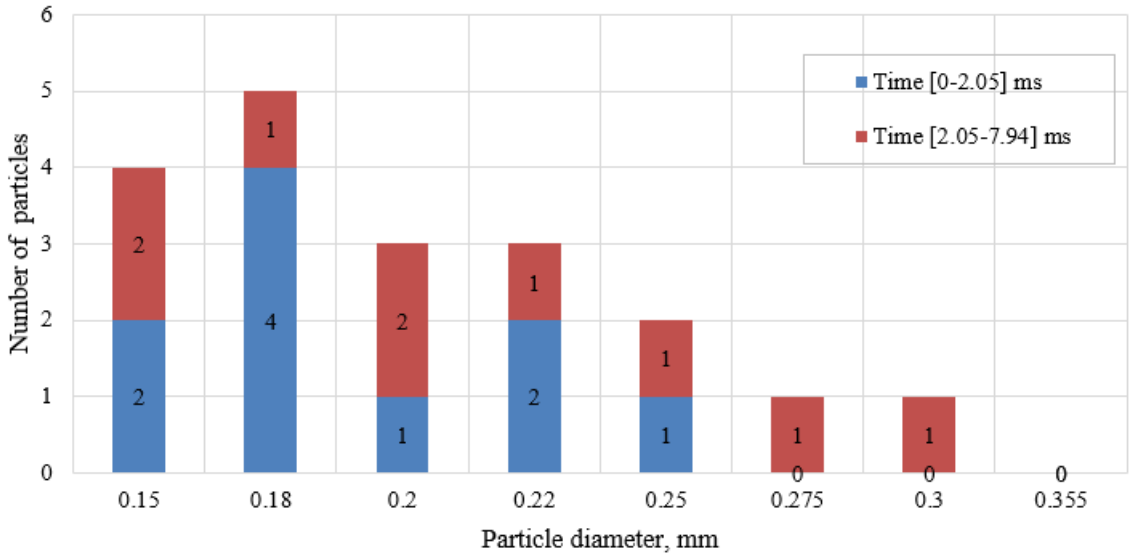


Figure 5.12. The distribution of extracted sand particles

5.3.2 Validation with laboratory and well data

In order to attempt to conduct the comparison of simulation results with laboratory and well data in a different time and geometric scale, the all results were normalized. The dimensionless time interval is defined as follows:

$$t_d = \left(\frac{\mathrm{t}}{t_{end}}\right) \tag{5.7}$$

where t -current time; and t_{end} – the time period up to the moment when sand production ends in produced liquid. The dimensionless time intervals vary between 0 and 1, where 0 corresponds to the start of production and 1 corresponds to the time at the end of sand production t_{end}.

The dimensionless cumulative sand production is given as:

$$Mcum_d^t = \left(\frac{\int_{t=0}^{t} M_t dt}{\int_{t=0}^{t_{end}} M_t dt}\right) \tag{5.8}$$

where M_t – is the instantaneous sand rate and ∇t is the time interval.

The sand production increments that reflect different sanding rates at different time intervals were calculated as:

$$M_d^i = \left(Mcum_d^i - Mcum_d^{i-1}\right) \tag{5.9}$$

where $i \in t_d$.

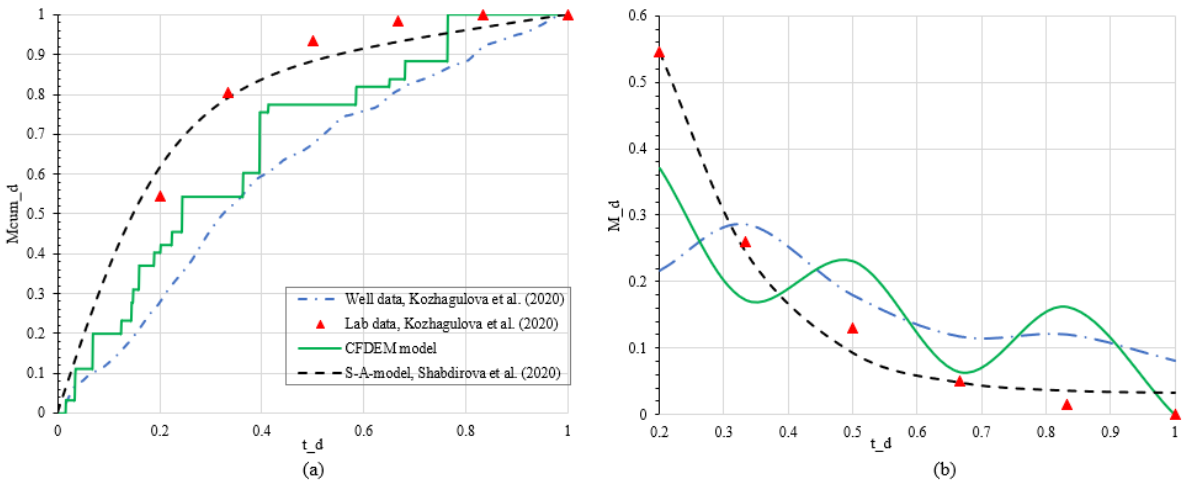


Figure 5.13. Sand production results comparison with experimental and field data. a) Cumulative sand production behavior; b) Sandrate behavior

The simulation outcomes were matched with the laboratory information, well data (Kozhagulova et al., 2020b) and semi-analytical model (A. Shabdirova et al., 2020) in the Figure 5.13. Regardless of the wide vary of the overburden stress, drawdown pressure, and geometrical scale among studies, the quality comparisons in the Figure 5.13 demonstrate same behavior of reducing sand production increments from the initial high sand rate of transient sand production. The quantitative analysis between different data sets will be conducted in the future studies.

5.3.3 Sand production results analysis

The outcomes of the CFD and the DEM simulations are compared in terms of the fluid and the particle velocities in the Figure 5.14. There was a large movement of the particles near the outlet at the start of the sand production simulation is related to a high fluid velocity distribution in the Figure 5.14a. The instant fluid flow moving out of the sample pushed the particles in direction to the exit, which triggered more particles produce in the beginning, as illustrated in the Figure 5.11. The dynamic zones of the particle and the fluid activities reduce in size in the Figure 5.14b and the Figure 5.14c, which indicates a stabilization stage in the granular structure of the sample as the majority of particle movements had stopped. Mentioned movements could occur due to the sand arching mechanism outlined in the studies of sand production in the well (Polillo & Graves, 1994; Risnes et al., 1982a) and in the laboratory (Hall & Harrisberger, 1970; B Wu et al., 2016). The velocity distributions of particles and fluid inside the sample converted to uniform in the Figure 5.14c, when the sand production stopped (see Figure 5.11). The fluid flow transferred to a steady regime since the reduction of the sand production. This aligns with the transient behavior of sand production described

in the experimental study by Kozhagulova et al. (2020). A steady-state fluid flow was found for a constant pressure draw-down after and sand production in the beginning.

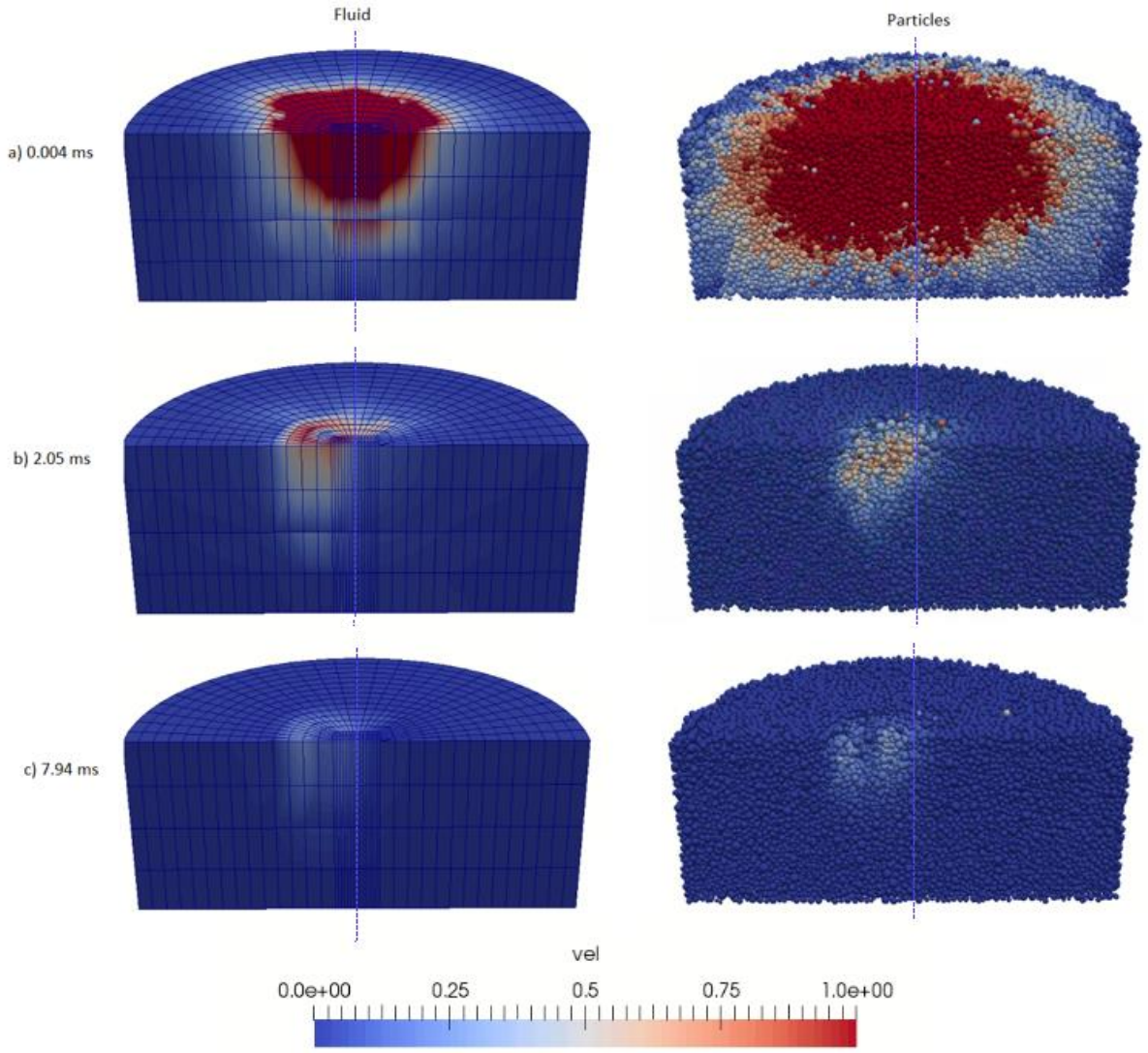


Figure 5.14. The distribution of particles and flow velocities in space and time during the $3D$ sand production test

Figure 5.15 depicts the bond and contact number values during sand production counted by usage of Eqs. 5.1 and 5.2; the timeline in the legend corresponds to the Figure 5.14. The biggest changes in the bond and the contact numbers occurred between the two states in Figure 5.14a and Figure 5.14b, while smaller change happened in the next stage of a few sand bursts as shown in Figure 5.11. The contact number grew across the sample's radius in Figure 5.15, which is in line with the results in Figure 5.14a. It has showed the majority of the sample's particles moving with high fluid velocities. This shows that the plastic zone's interface at $7r$ identified in terms of the particle rearrangement in

Figure 5.15 had grown to fill the whole sample under the fluid flow conditions of the sand production procedure. The progress of the contact number close to the perforation zone in Figure 5.15 would help the growth of the frictional resistance at the new contacts that could give rise to the development of a sand arch around the hole. The distance $3r$ was the place of the highest contact number identified earlier as the boundary of the perforation damage zone.

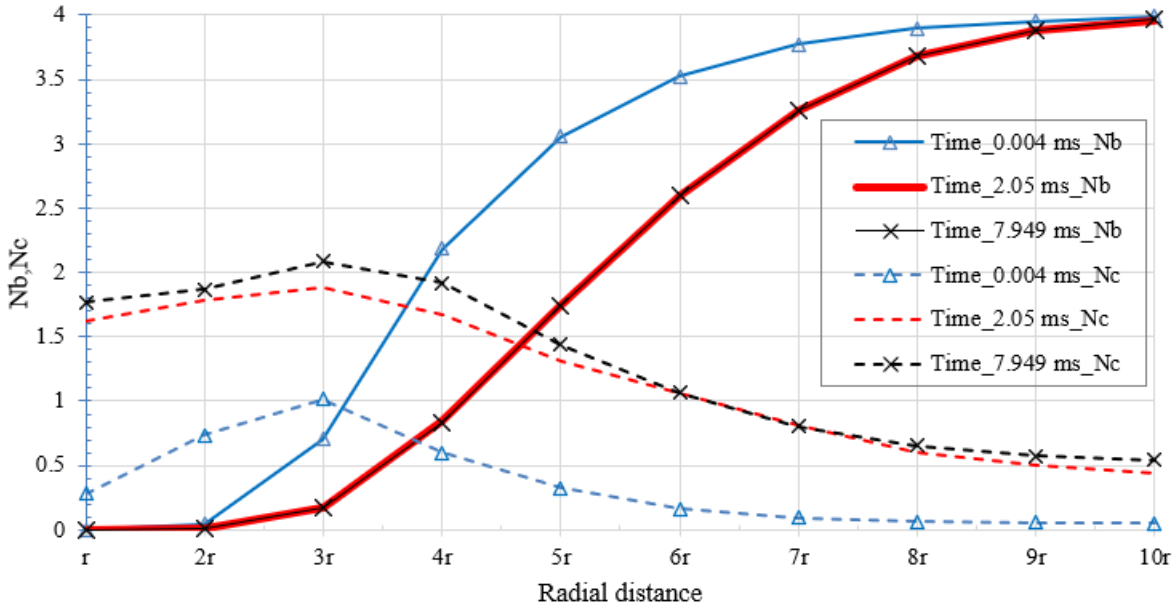


Figure 5.15. The average contact and bond numbers (N_c, N_b) versus radial distance during sand production test

Table 5-6. Number of particles (m_i) in the annular cylinders

Time step	**Radial distance**	r	$2r$	$3r$	$4r$	$5r$	$6r$	$7r$	$8r$	$9r$	$10r$
0.004 ms	m_i	474	2657	4659	6225	7923	9746	11289	12920	14686	16271
2.05 ms	m_i	808	2406	4190	6031	7969	9788	11444	13004	14786	16355
7.94 ms	m_i	799	2397	4204	6039	7954	9769	11467	13004	14791	16359

Table 5-7. The change of particles number (Δm_i) in the cylinders

Time step	**Radial distance**	r	$2r$	$3r$	$4r$	$5r$	$6r$	$7r$	$8r$	$9r$	$10r$
0.004 ms	$\Delta m_i(\%)$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.05 ms	$\Delta m_i(\%)$	**70.5**	**−9.4**	**−10.1**	**−3.1**	0.6	0.4	1.4	0.7	0.7	0.5
7.94 ms	$\Delta m_i(\%)$	**68.6**	**−9.8**	**−9.8**	**−3.0**	0.4	0.2	1.6	0.7	0.7	0.5

The numbers of particles placed in every annular cylinder are shown in Table 5-6, and the percentage of change of these values is depicted in Table 5-6. Particles moved from the zones between

$2r$ and $4r$ to the central perforation tunnel (within r) during the sand production simulation. The number of particles fell 10% in zones $2r$ and $3r$, while it grew for up to 70% in the perforation tunnel. Particles inside the zones from $5r$ and further did not move so far due to the fact they were still cemented by a larger bond number than the contact number (i.e., $N_b > N_c$). The extracted particles are only a small portion of the total particles, and they are among the unbonded particles which have transported from the perforation damage zone.

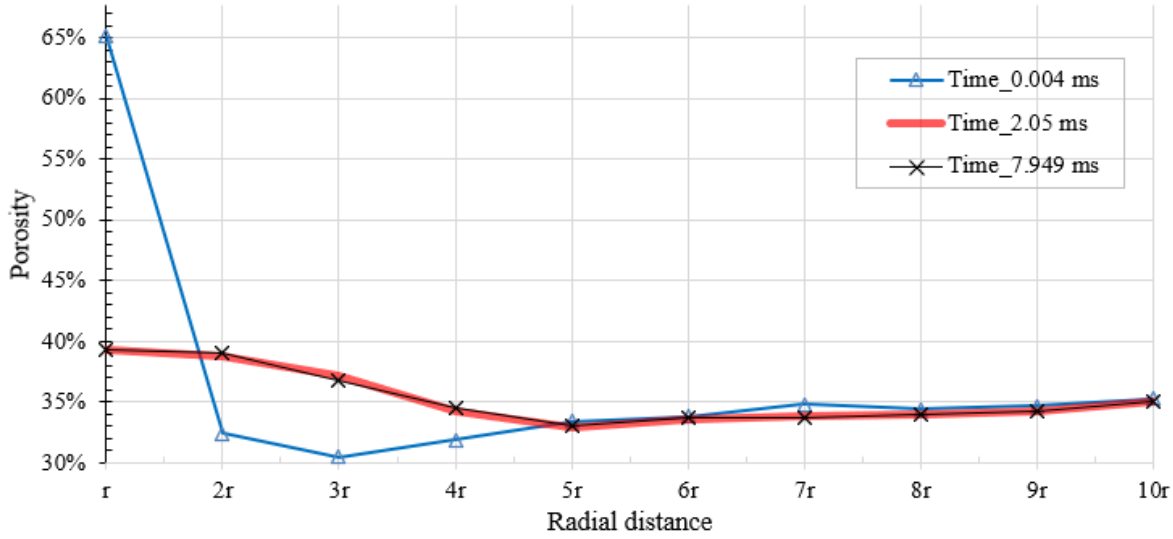


Figure 5.16. Porosity distribution versus radial distance

Table 5-8. Porosity change ($\Delta\phi_i$) during the sand production test

Time step	**Radial distance**	r	$2r$	$3r$	$4r$	$5r$	$6r$	$7r$	$8r$	$9r$	$10r$
$0.004\ ms$	$\Delta\phi_i(\%)$	-2.46	0.30	-0.28	0.29	-0.06	0.05	0.04	-0.03	0.08	0.07
$2.05\ ms$	$\Delta\phi_i(\%)$	-41.01	19.59	21.51	7.60	-1.50	-0.49	-2.57	-1.14	-1.30	-0.71
$7.94\ ms$	$\Delta\phi_i(\%)$	-40.88	20.31	20.02	8.47	-1.21	-0.26	-2.98	-1.09	-1.33	-0.78

The outcomes of the porosity change during sand production are shown in Figure 5.16 and Table 5-7. While the process of particles' migration to the perforation tunnel in the increasing way, the porosity in this zone was decreasing from 65% to 40%. And on the other hand, the porosity in the zones between $2r$ and $4r$ was increasing while the loose of the particles. It should be noted that these are the compressed zones of lower porosity formed after perforation, hence zones have been expanded mostly under the fluid flow condition. Even though the plastic zone in terms of bond breakage and new contact formation had increased to fill the whole sample (see Figure 5.15), the smaller perforation damage zone played a more significant role in the production of sand particles. It characterized the

clusters of the unbonded particles in a compressed state; when subjected to fluid flow, it rebounded easily and issued its particles to achieve a looser state of higher porosity.

The porosity inside the sample at the end of the transient sand production is actually bigger than the intact porosity value of 35%. Therefore, the perforation procedure formed more preferential condition for the outflow of fluid from the sample same as the design principle in the oil-gas industries.

5.4 Conclusion

A modified JRK model developed by Rakhimzhanova et al. (2018) has been implemented successfully in the open-source programs LIGGGHTS and CFDEM to simulate the bonding behavior of a weak cemented sandstone in Kazakhstan. A numerical sandstone sample was generated, perforated, and transferred to radial fluid flow from the boundary in a sequence of various simulation stages. The number of bonded and unbonded contacts was calculated in each step in order to evaluate the perforation damage and the plastic zone during the entire simulation procedure. Moreover, the cylindrical sample has been subdivided into annular cylindrical zones around the central perforation tunnel. The porosity was counted for every zone, that guaranteed the ability of the internal porosity distribution across the sample. The perforation procedure formed a damage zone of approximately $3r$ to $4r$ thickness of the unbonded particles with almost complete bond breakage in this zone, where r is the radius of the perforation tunnel. A compressed zone of lower porosity was inserted inside the perforation damage zone. Within the fluid flow conditions it has grown quickly to get a greater porosity than the intact porosity as the particles moved towards the perforation tunnel during sand production. The simulation defined a transient sand production phenomenon when small particles were produced initially in a series of rapid events followed by bigger particles in less frequent sand bursts. Particle migrations inside the sample stopped once there was no longer sand production.

CHAPTER 6 - MICRO AND MACROSCOPIC BEHAVIOR OF SAND PRODUCTION IN SINGLE FLUID FLOW WITH DIFFERENT RESERVOIR FLUIDS

In this Chapter, a three-dimensional CFD-DEM simulation method was developed to investigate the influence of light and heavy oil reservoir fluids on sand production. A different sand production pattern is observed for each heavy and light oils.

6.1 Introduction

In this Section, the dry sample developed in Chapter 5 was used to investigate the sanding mechanisms for different reservoir fluids. The fluid properties and reservoirs were prescribed and modelled after the conditions in the Kenkiyak and Uzen oil fields in Kazakhstan. The coupled CFD-DEM model was optimized in terms of selecting optimal the CFD mesh. The heavy oil was found to cause more severe sand production than light oil due to its higher transport capability and the formation of a more regular particle velocity trajectory pattern that mobilized a larger proportion of the sample's particles for sand production.

6.2 Simulation process

6.2.1 Perforated sample preparation

The numerical simulation by analogy with Section 5.2 was conducted in three subsequent stages of the consolidated sample preparation, the perforation and the sand production. The numerical formulation of CFD-DEM systems described in Sections 3.1-3.2. The initial sample was prepared with the same number of particles and their distribution as Section 5.2. A summary of the particle information is given in Table 6-1, while the generated particles are shown in Figure 6.1. It should be noted that the number of particles for each particles template in Table 6-1 slightly differs from Table 5-1, since the particles in each study were randomly generated.

The numerical sample was then compressed to achieve vertical stress of $1\ MPa$. The diameter (D) of the intact sample is equal to $15.12\ mm$, and the height (H) is equal to $6.008\ mm$.

Table 6-1. A summary of the particle numbers and sizes in numerical simulation

Particle diameter, $\boldsymbol{d_p}$ (mm)	0.15	0.18	0.2	0.22	0.25	0.275	0.3	0.355
Number of particles	20758	15529	13357	13825	17507	9929	5414	3681
Mass ratio (%)	5.8	7.5	8.9	12.2	22.7	17.1	12.1	13.7
Mass (mg)	91.7	118.5	139.9	192.7	358.1	270.3	191.3	215.6

A hollow tunnel was created by driving a penetrometer along the central axis of the intact sample to simulate the perforation process in well completion. The penetrometer was removed once it reached the end plate. The perforation tunnel served as the drainage boundary to facilitate fluid flow through the sandstone toward the outlet that is positioned at the top end of the tunnel or the centre of the circular top surface of the sandstone sample in Figure 6.1. The perforation created damages to the sample's initial structure, which affected the sand production process in the final stage of the microscopic simulation. Further details on the simulation processes can be found in the Chapter 5.

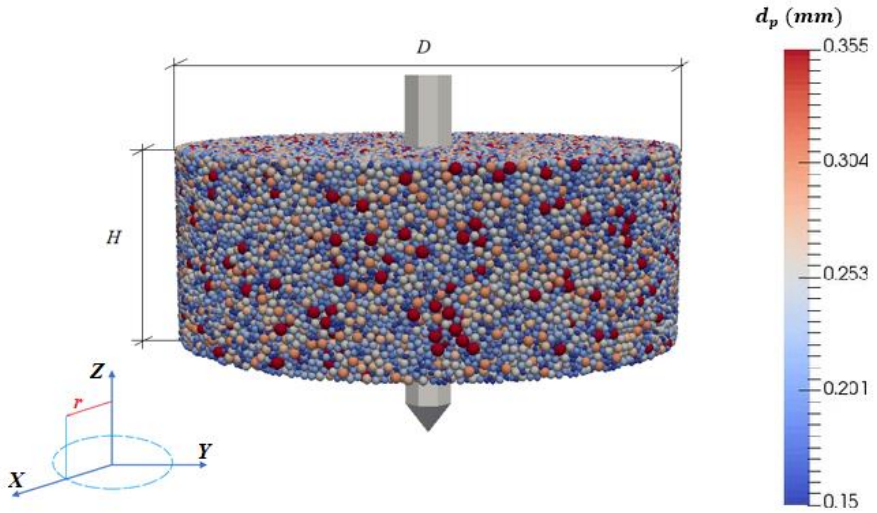


Figure 6.1. The sample perforation

6.2.1 Sand production simulation

The perforated sample was used for sand production simulation where different fluids were injected radially from the external boundary towards the vertical perforation tunnel to mimic the condition in the experiments on a similar sandstone (Kozhagulova et al. 2020). The sand production simulation was conducted using the CFDEM program (Christoph Goniva et al., 2012). To study sand production in different reservoirs in Kazakhstan, the properties of the injection fluids were varied to simulate light oil and heavy oil. The values of the fluid density and viscosity that enter Eqs. 3.14 and 3.18 are given in Table 6-2.

Table 6-2. The reservoir fluid properties of Uzen, Kenkiyak oilfields in Kazakhstan(Bealessio et al., 2020)

	Reservoir fluid	**Kinematic viscosity,** m^2/s	**Density,** m^3/kg	**Oil field**
HO	heavy oil	$1.62 \cdot 10^{-4}$	927.09	Kenkiyak
LO	light oil	$4.12 \cdot 10^{-6}$	849.16	Uzen

For the CFD-DEM simulation of sand production, the fluid phase is represented in terms of a CFD mesh that has the same dimension as the perforated sample of the solid phase in Figure 6.1. The CFD grid was constructed using the HELYS-OS graphical user interface (ENGYS Ltd, n.d.). Figure 6.2 shows the computational domain of the fluid phase that is comprised of 2344 cells, the grid size of which was determined following the correlation in Eq. 3.22. In this case, the particle size was taken as the mean particle diameter ($D_{50} = 0.227\ mm$), which resulted in the value Δx_{cfd} should be greater than $3.44 \cdot 10^{-4}\ m$. In this Section, Δx_{cfd} equal to $3.98 \cdot 10^{-4}\ m$ was used. The numerical models of the solid phase in Figure 6.1 and of the fluid phase in Figure 6.2 were used for two separate computations of each phase, respectively. The exchanged information between the two computations is given in term of the interaction force, the pressure force, and the viscous force in Eqs. 3.3 and 3.4, which have been explained in the Section 3.1.

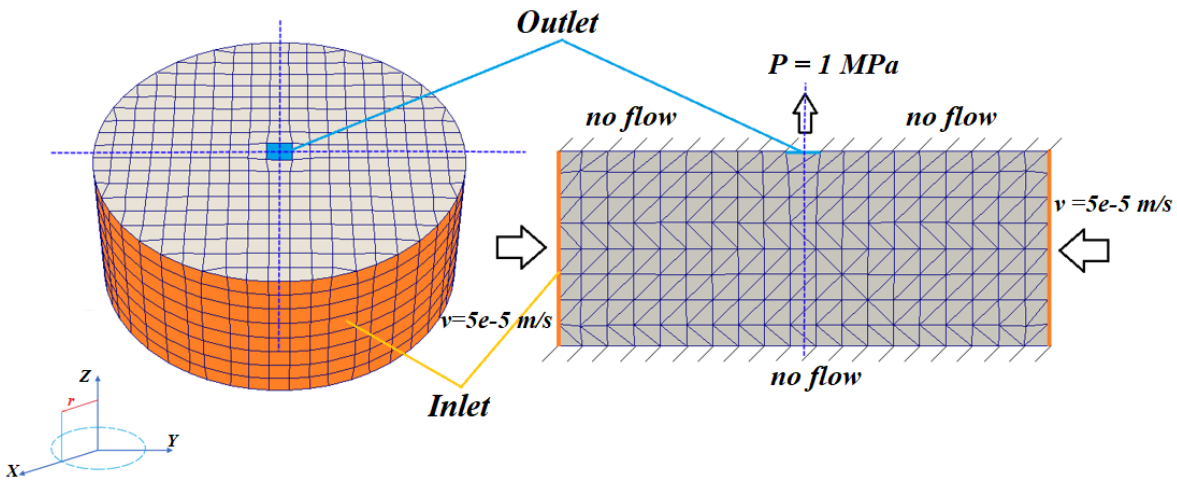


Figure 6.2. CFD simulation domain and boundary conditions

The boundary conditions are described in Figure 6.2 (Table 4-1, set ***II***). The top and the bottom walls were set as impermeable walls with no-slip boundary condition. There was a central opening on the top surface to serve as the fluid outlet. The light and heavy oils were injected from the whole

circumferential wall of the cylinder at a constant velocity of $\mathbf{5 \cdot 10^{-5}\ m/s}$. The outlet pressure was maintained at a constant $\boldsymbol{P = 1\ MPa}$ for all cases. The fluids in the simulation are considered incompressible. To check the computational convergence of the constructed CFD mesh, two additional meshes (fine and coarse) were tested, of which except the mesh size all other conditions are the same as the setup of the final selected mesh in Figure 6.2.

Three different mesh sizes are subjected to the same boundary conditions in Figure 6.2. The radial distributions of the fluid velocity along a horizontal plane at $\mathbf{4.8\ mm}$ from the bottom of the sample are shown in Figure 6.3. The maximum value of the fluid velocity at the outlet and the mean velocity at the final time step were calculated, the difference in these values between different mesh sizes were calculated as follows:

$$\Delta u_f^{mean}(\%) = \frac{\left|u_f^{mean} - u_{f\_b}^{mean}\right|}{u_{f\_b}^{mean}} * 100\% \tag{6.1}$$

$$\Delta u_f^{max}(\%) = \frac{\left|u_f^{max} - u_{f\_b}^{max}\right|}{u_{f\_b}^{max}} * 100\% \tag{6.2}$$

where $\boldsymbol{u_{f\_b}^{mean}}$, $\boldsymbol{u_{f\_b}^{max}}$ – are mean and maximum values of fluid velocity of the final selected mesh size in Figure 6.2.

Here, the discretized time step in the CFD simulation was set as $\mathbf{10^{-6}\ s}$, while the total simulation time was equal to $\mathbf{3 \cdot 10^{-4}\ s}$. The CFD time step was calculated using the Courant–Friedrichs–Lewy condition (Courant et al., 1986; Laney, 1998):

$$C = \frac{u_f \Delta t}{\Delta x_{cfd}} \leq const \tag{6.3}$$

where Δt is the CFD time step. The value of $const$ is usually less than 1 and can change with the method used to solve the discretised equation, especially depending on whether the method is explicit or implicit. If $C < 1$, the fluid will move from one cell to another within one time step. While for $C > 1$, the fluid travels through two or more cells in each time step and this can affect negatively to the numerical convergence. The C^{mean} and C^{max} in Table 6-3 are the mean and the maximum values of the Courant number C at the end of the simulation.

Table 6-3. Mesh convergence analysis results

Parameters	Symbol	Fine	Medium	Coarse
CFD cell length (m)	Δx_{cfd}	$\mathbf{3.29 \cdot 10^{-4}}$	$\mathbf{3.98 \cdot 10^{-4}}$	$\mathbf{5.04 \cdot 10^{-4}}$
Mean courant number	C^{mean}	$\mathbf{2.78 \cdot 10^{-7}}$	$\mathbf{2.33 \cdot 10^{-7}}$	$\mathbf{1.8 \cdot 10^{-7}}$
Maximum courant number	C^{max}	$\mathbf{2.61 \cdot 10^{-5}}$	$\mathbf{2.36 \cdot 10^{-5}}$	$\mathbf{1.56 \cdot 10^{-5}}$
Difference in the mean velocity (%)	Δu_f^{mean}	2.29	0	15.33
Difference in the maximum velocity (%)	Δu_f^{max}	4.72	**0**	81.1

The radial distribution of the fine and the selected medium mesh sizes are almost identical in Figure 6.3, which reflects a mesh convergence condition. The coarse mesh size, on the other hand, showed significant discrepancy that could be as large as 81% as compared to the convergence results of the other two mesh sizes. The medium mesh size was selected as the final size for further simulation as it achieved both the convergence condition and the CFD-DEM condition in Eq.16.

The interactions between the fluid and the particle phases were simulated by a four-way coupling method (Norouzi et al., 2016). The motion of the particles affects the motion of the fluid, and vice versa. The coupling information was given in terms of the pressure force, the viscous force and drag force. The time steps for the DEM and CFD simulations were $2 \cdot 10^{-8}\, s$ and $10^{-6}\, s$, respectively. The CFD simulation was computed for every 50 (coupling number) of the DEM simulation and the coupling information was exchanged accordingly. This setting in sufficient accuracy yet allows for higher computational efficiency. The effect of coupling number to CFD-DEM calculations have discussed in Section 7. It took 19 days to complete the simulation time of 0.0606 s for a coupled simulation case in this study using a highly powerful computer cluster of 8 cores and 500 GB of random-access memory. It was found that the raising of number threads more than 8 does not significantly increase the calculation speed.

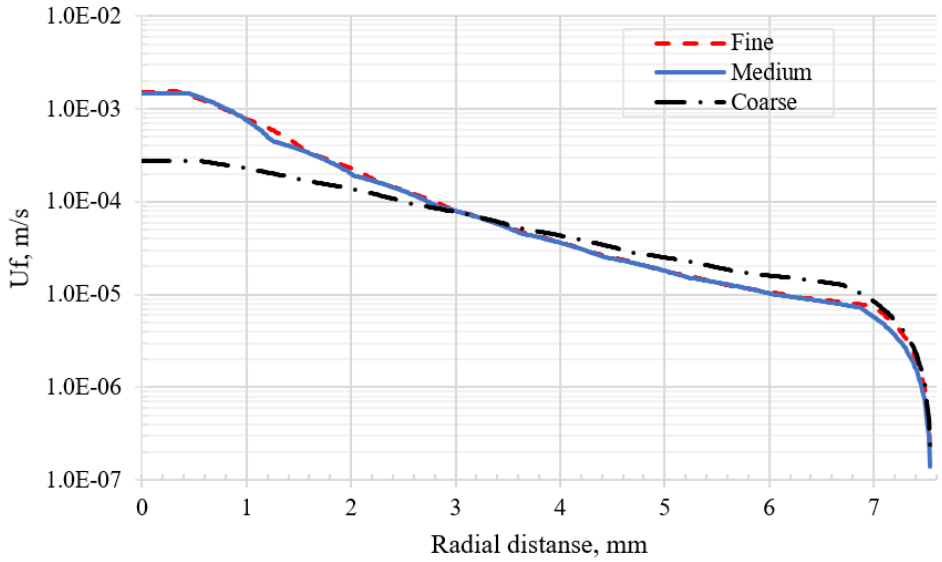


Figure 6.3. The fluid velocity distribution along the line at $z = 4.8\ mm$ from the bottom of the sample

6.3 Simulation results

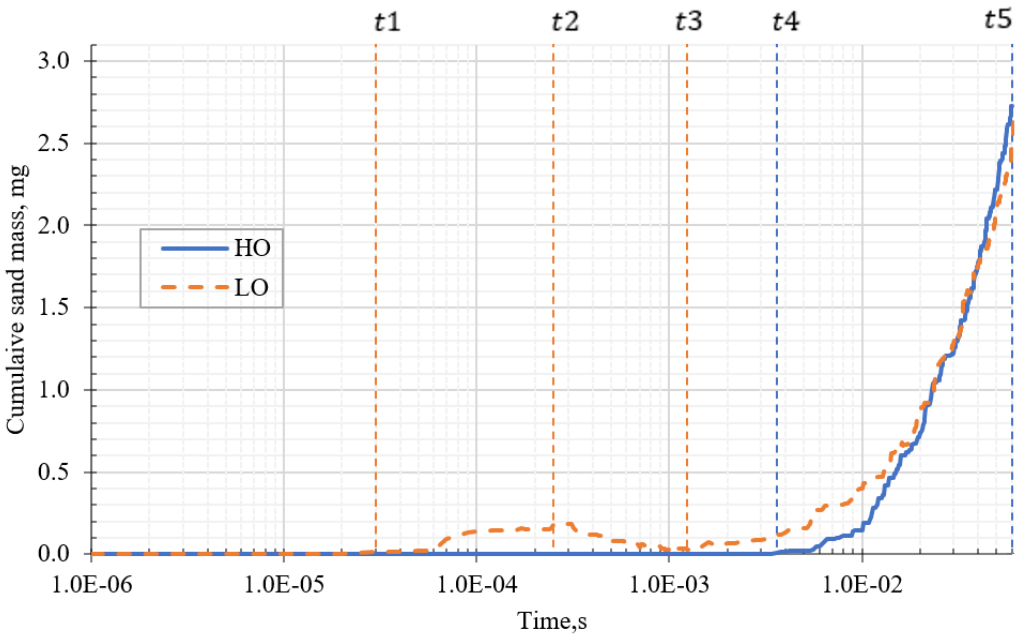


Figure 6.4. Cumulative and production for different reservoir fluids

Figure 6.4 shows the results of the cumulative produced sand mass for the heavy oil (HO) and the light oil (LO) over the same total simulation time of $0.0606\ s$. Sand production started earlier for LO at the marked time step $t1$ as part of a transient sanding behavior that reached a peak sand mass at $t2$ and then decreased to zero over time. The transient behavior was followed by a continuous sand production that started from the time step $t3$ and was on-going at the end of the simulation. For the

HO case, there was no transient sanding behavior but only the continuous sand production from the time step $t4$. Although sand production was started later, the HO case produced more sand at the end of the simulation, that is the time step $t5$.

The produced particles were counted as they crossed the level of about one particle diameter above the top surface of the sample. If the produced particles moved upward to an even higher level of about four particle diameters above the top cap, they were deleted from the simulation as it was assumed that the particles had successfully escaped the sample. The transient cumulative sand mass of light oil decreased to zero from $t2$ to $t3$ is due to the produced particles were not able to completely leave the sample but rather return to deposit on the top surface. The escape fluid velocity at the exit point is not sufficient to lift the particles up to the upper elevation to be completely removed from the specimen.

Figure 6.5 shows the results of the average particle velocity (HO_p, LO_p) and the fluid velocity at outlet (HO_f, LO_f) for HO and LO cases. The results were normalized by the final value of the average particle velocity at $t5$ in the LO case as it is the smallest velocity among the four final results. The normalization would allow comparison of the flow dynamics between different phases and cases. As the flow began, the fluid velocities fluctuated until a steady state of a constant velocity was achieved for both cases. The fluctuation is much more significant for heavy oil due to a higher inertia in this case. The particle velocity is higher than the fluid velocity for LO until $t2$ when the difference becomes zero, and this is associated with the transient sanding behavior. The continuous sand production occurred in the steady state flow condition when the fluid velocity is nearly a hundred times higher than the average particle velocity, such that the hydrodynamical forces can continuously transport particles out of the sample. The particle velocity of LO decreased more quickly in the steady state and became less than the HO particle velocity at the final time step, this would indicate a more severe on-going sand production for the heavy oil as it was discussed in the literature (e.g., Al-Awad et al., 1999).

Before the beginning of the continuous sand production, the fluid velocity is higher than the particle velocity, but it is not sufficient to transport the particles out of the sample. The fluid velocity at the continuous sanding onset can be compared to the terminal settling velocity, that is dependent on the particle size and can be determined from the Stokes' Law as follows (Coker, 2007):

$$\boldsymbol{Uset} = \frac{\boldsymbol{g d_p^2 (\rho_f - \rho_p)}}{\boldsymbol{18 \mu_f \rho_f}} \tag{6.4}$$

where ρ_p is the particle density; the other parameters have been defined in the previous equations. The sample consists of eight different particle sizes (Table 6-1), and $Uset\_min$ and $Uset\_max$ are the minimum and the maximum settling velocities of the smallest, and of the largest particles, respectively. In Figure 6.5, the outlet fluid velocity of heavy oil is greater than $Uset\_max$, whereas for the LO case the outlet fluid velocity is capped between the two extreme settling velocities. The outlet velocities are similar for heavy and light oils, but the settling velocities are different due to different fluid densities and viscosities. The transport capability of HO is higher than LO that would lead to more severe sand production.

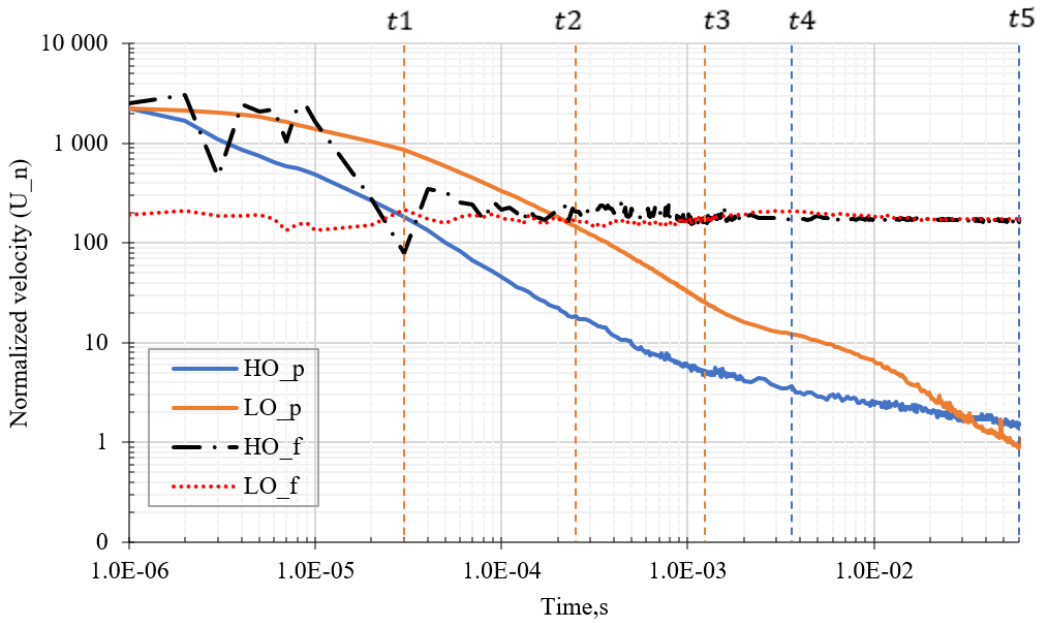


Figure 6.5. Average particle velocity and fluid velocity at outlet. The velocities are normalized by the final average particle velocity of light oil, that is $U_p^{t5} = 1.34e-2\ m/s$

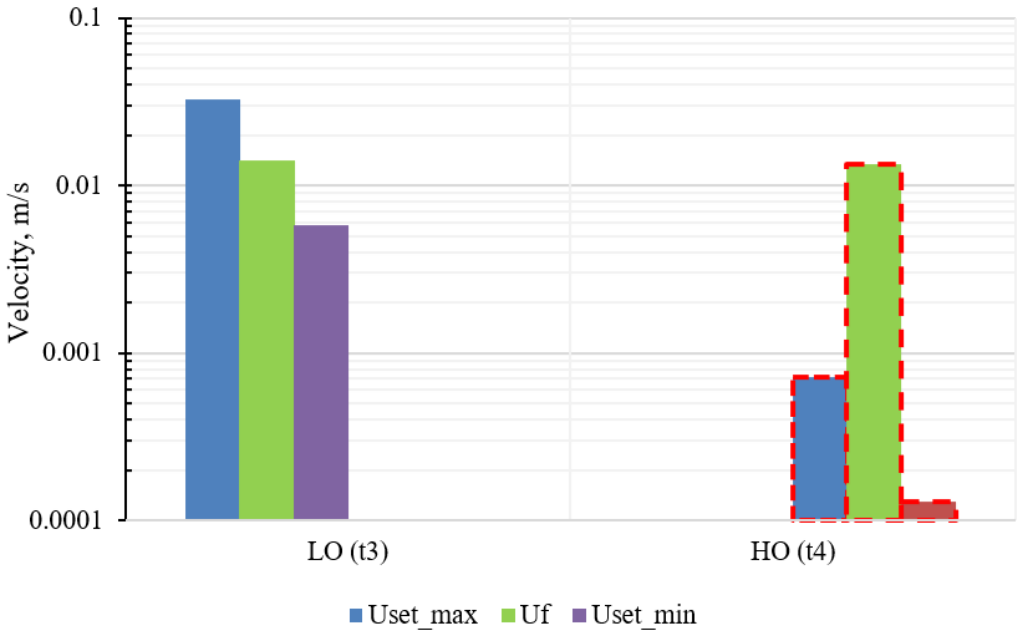


Figure 6.6. Settling velocity and outlet velocity at the onset of continuous sand production

The detailed information of the produced particles is given in Table 6-4 . The transient sanding behavior of light oil was mostly associated with small particle sizes, among which the particles around the D_{50} size occupy half of the produced particles. The continuous sand production, on the other hand, produced particles of all sizes for both HO and LO. The particle size distributions of the produced sand at $t5$ in Figure 6.7 are slightly above the distribution of the intact sample as there are more fine particles in the produced sands. The produced sand mass in the HO case is greater than the combined sand masses of both the transient and the continuous sand productions in the LO case.

Table 6-4. The cumulative sand production values and PSD

	Total mass, mg	**Diameter (mm)**	**0.15**	**0.18**	**0.2**	**0.22**	**0.25**	**0.275**	**0.3**	**0.355**
$LO[t1-t3]$	0.179	Quantity	3	1	2	2	4	1	0	0
		$\Delta m_i(\%)$	7.4	4.3	11.7	15.6	45.8	15.2	0.0	0.0
$LO[t3-t5]$	2.516	Quantity	28	30	23	18	33	12	9	6
		$\Delta m_i(\%)$	4.9	9.1	9.6	10.0	26.8	13.0	12.6	14.0
$HO[t4-t5]$	2.726	Quantity	33	33	25	19	35	17	11	4
		$\Delta m_i(\%)$	5.3	9.2	9.6	9.7	26.3	17.0	14.3	8.6

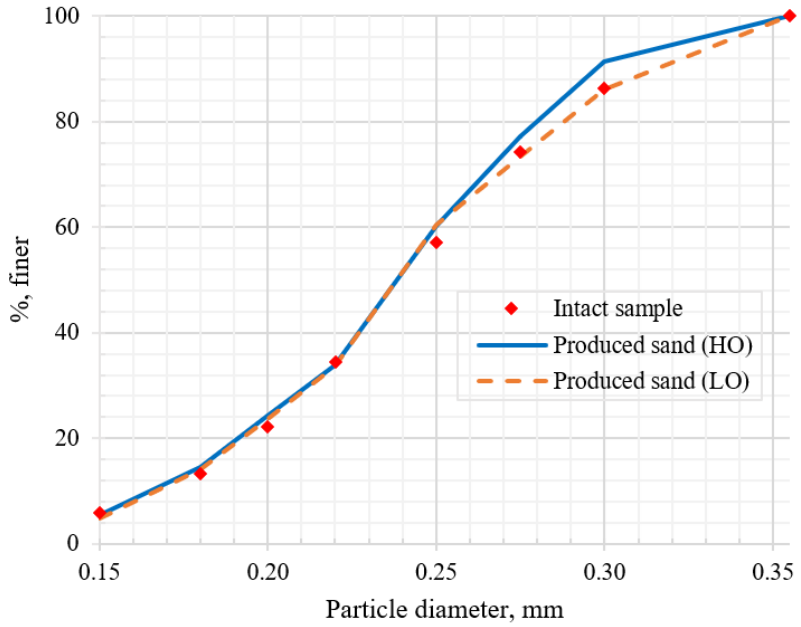


Figure 6.7. Particle size distributions of the final produced sands and of the intact sample

The particles of the sandstone sample can have multiple contacts with the surrounding neighbours that are consisted of both the bonded and unbonded contacts. All the bonded contacts were only formed once in the intact state using the modified JKR model. The mechanical impacts in the following perforation and sand production stages could break the bonds and form new unbonded contacts that were modelled using the Hertz contact model. The algorithm goes through the list of all particles in the sample and count the numbers of the modified JKR contacts (bonded) and of the Hertz contacts (unbonded) for each particle to provide the total bond number and the total unbonded contact number. Furthermore, the coordination number (Z) can be calculated as the average total contact number as follows:

$$Z = \frac{\sum_{j=1}^{k} N_t^j}{k} \tag{6.5}$$

where N_t^j is the total contact number of the j^{th} particle that includes both the bonded and the unbonded contacts.

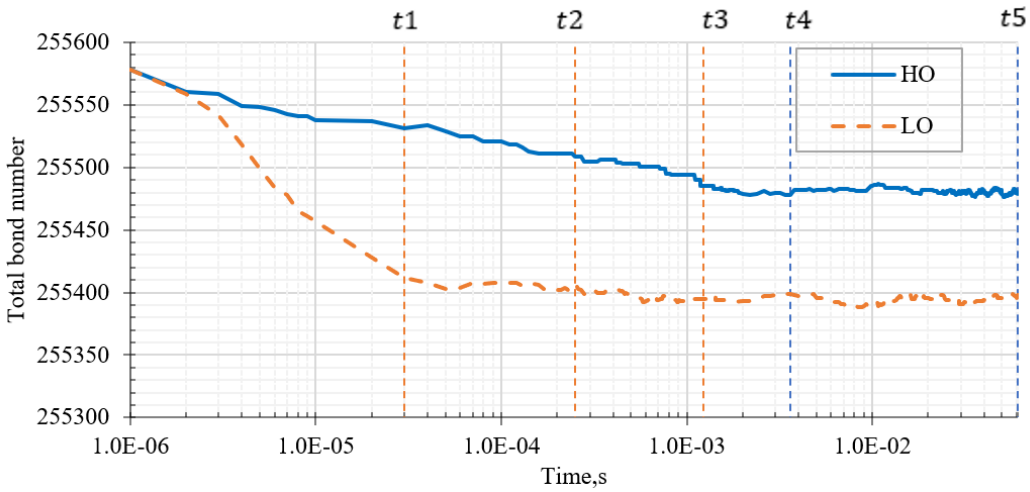


Figure 6.8. Variation of the total bonded contacts in the system

Figure 6.8 shows the variation of the total bonded contacts in the sample during the sand production process. The sanding onset conditions at $t1$ and $t4$ for the LO and HO cases, respectively, occurred after the fluid injection broke a certain number of bonds and a relatively constant bond number was maintained during the sand production process until the end of the simulation. The fluid flow first created damage to the bond network and then removed the loosen particles from the sample. It is noted that the produced sand masses only account for less than 0.2% the total sample, and the effect should hence be limited in these simulation cases.

Figure 6.9 shows the variation of the coordination number. The initial value of 3.5 is less than 4, or the requirement for mechanical stability of equal size frictional spheres (Song et al., 2010). The perforation created damages to the sample in terms of a region of unbonded particles around the central axis of the sample (Section 5) that are potentially prone to sand production. The fluid flow tends to move all particles toward the outlet point on the top surface that increases the coordination number until the end of the simulations. The sanding onset is however marked with a temporary drop in the coordination number that indicates local instability. For LO, the decrease in the coordination number is more extended from $t1$ to $t3$ with a minimum value at $t2$ when there is a maximum sand mass during the transient sand production.

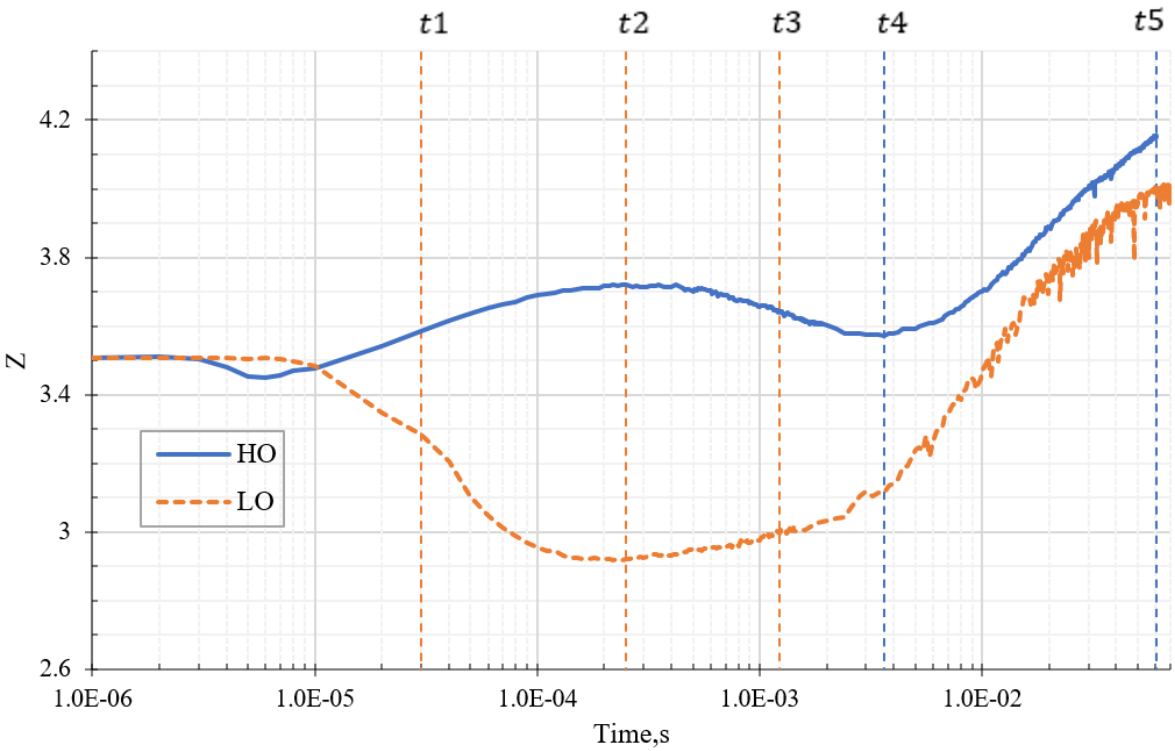


Figure 6.9. Variation of the coordination number

6.3.1 Interaction forces

The influence of fluid on the particles immersed into the fluid in the CFD-DEM system is expressed by drag, pressure, and viscous forces. The theoretical expressions of these forces have presented in Section 3.1-3.2. Figures 6.10-6.12 show the results of average drag, viscous and viscous forces for HO and LO cases. Similar to Figure 6.5, the results were normalized by the values at $t5$ in the LO case.

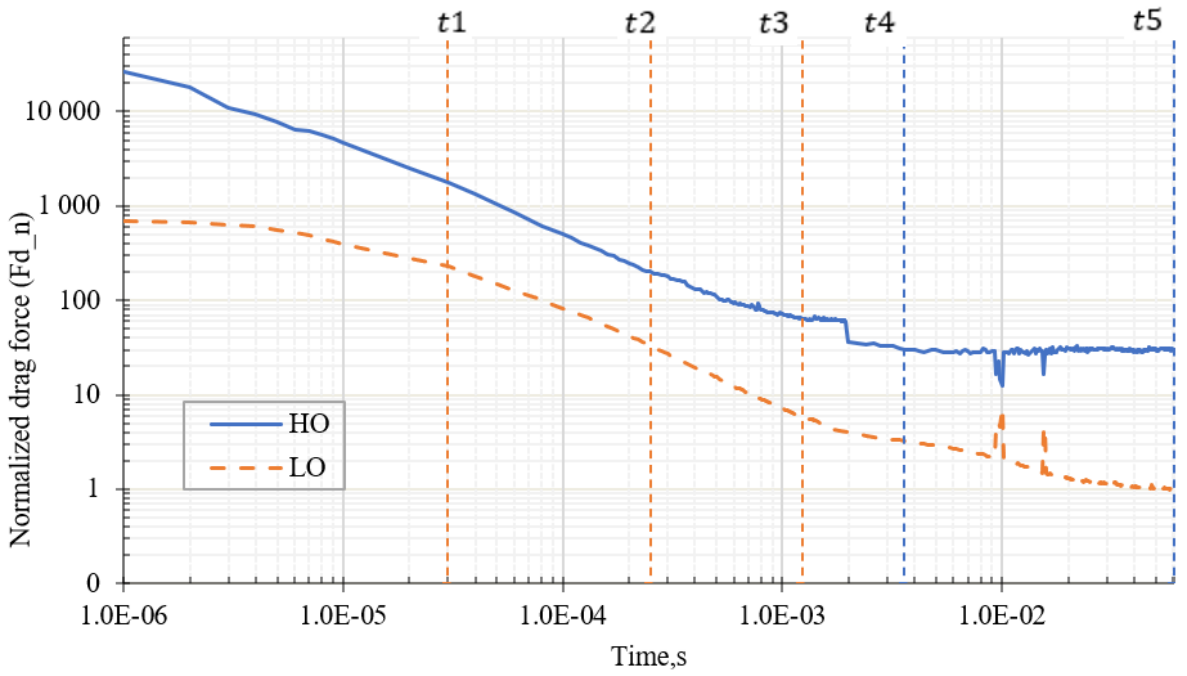


Figure 6.10. Average drag force behavior. The forces are normalized by the final average drag force of light oil, that is $F_d^{t5} = 1.95 \cdot 10^{-8}\ N$

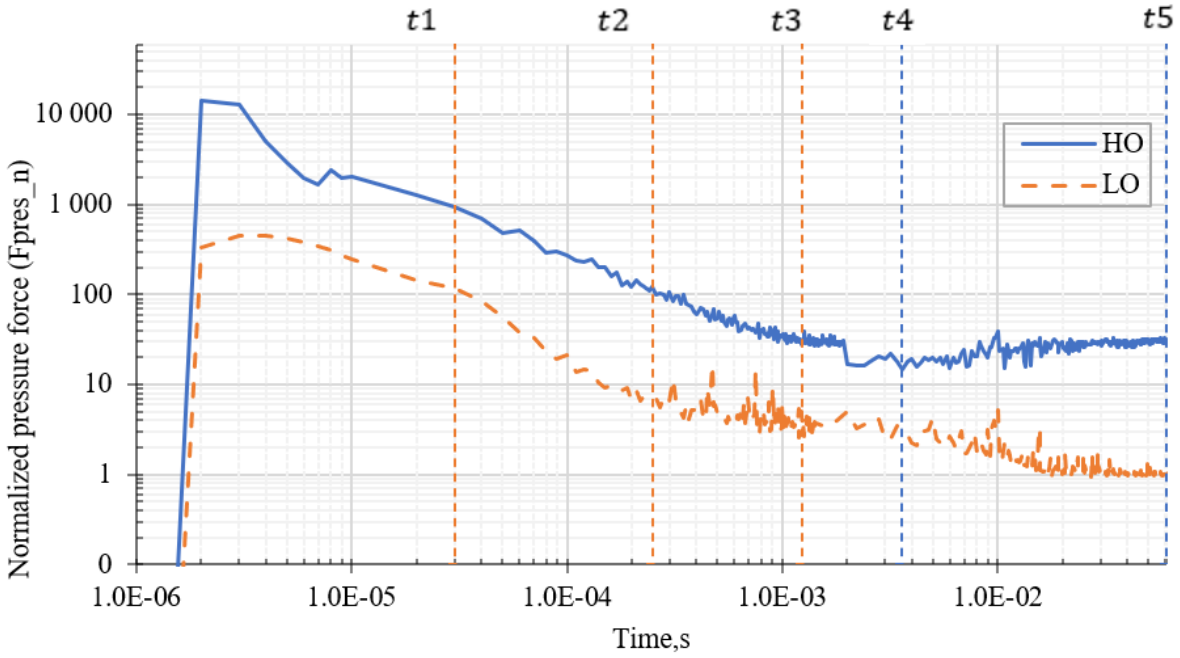


Figure 6.11. Average pressure forces. The forces are normalized by the final pressure force of light oil, that is $F_{Pres}^{t5} = 2.28 \cdot 10^{-8}\ N$

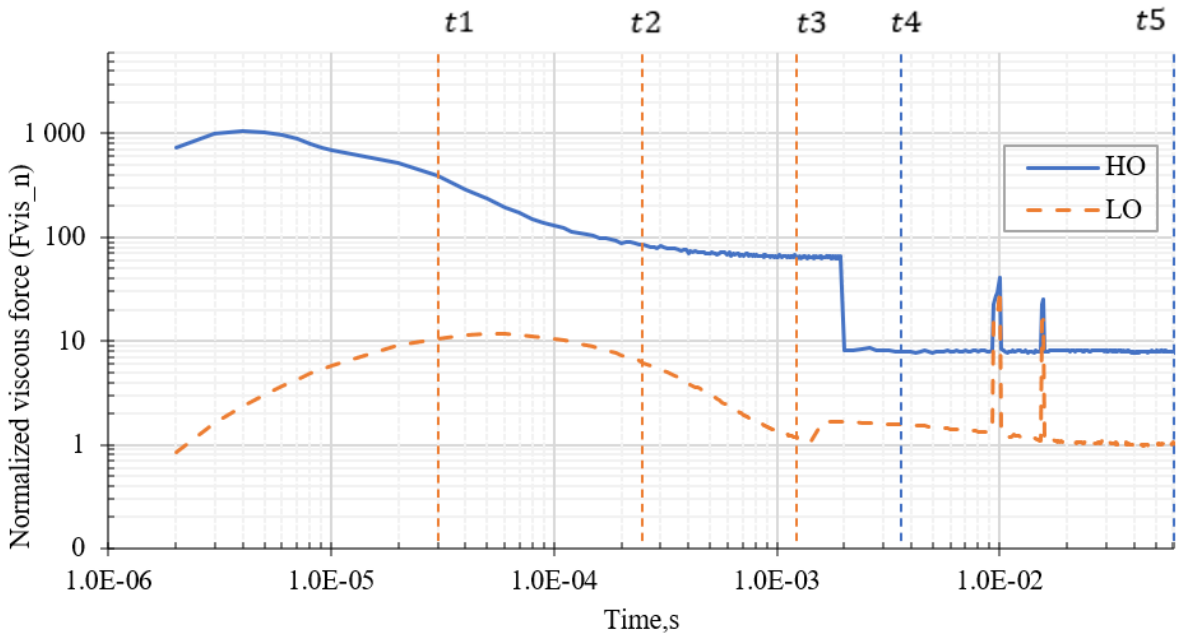


Figure 6.12. Average viscous force behavior. The forces are normalized by the final viscous force of light oil, that is $F_{vis}^{t5} = 1.2 \cdot 10^{-10}\ N$

Generally, all forces in the case of HO are much bigger than LO. The ratios between drag, pressure, and viscous forces for HO and LO cases correspondingly equal to $29.83, 29.17$ and 7.74. at the time $t5$ of simulations. The average value of viscous force in both HO and LO cases is significantly less than drag and pressure forces. In order to assess the contribution of viscous to the overall system, let us calculate the change in the fraction of viscous force (F_{vis}) in total force as:

$$\Delta Frac_{vis}(\%) = \frac{|F_{vis}|}{|F_{vis}| + |F_d| + |F_{Pres}|} * 100\% \tag{6.6}$$

The mean and max values of $\Delta Frac_{vis}$ of the LO simulation cases are correspondingly equal to 0.23% and 1.41%. The mean and max values of $\Delta Frac_{vis}$ of the HO simulation case are correspondingly equal 0.09% and 0.43%. Based on these results, the contribution of viscous force to the total force in both HO and LO cases are small, and this force can be neglected further.

6.3.2 Microscopic analysis of the sand production

The microstructure of the specimen can be analysed in more detail as the simulation domain is divided into five cylindrical rings as shown in Figure 6.13. The ring thickness is equal to the diameter of the penetrometer, which is larger than in the previous simulation (Section 5) and allows for better statistical representation. The rings were counted as zones 1 to 5 from the central axis outwards, where the outermost ring was not taken in the analysis as it was affected by the external boundary condition. These zones were only used in the post-simulation analysis to capture the change in the microscopic properties during the sand production process, but do not affect the DEM and the CFD computations. The programming code of post processing shown in the Appendix.

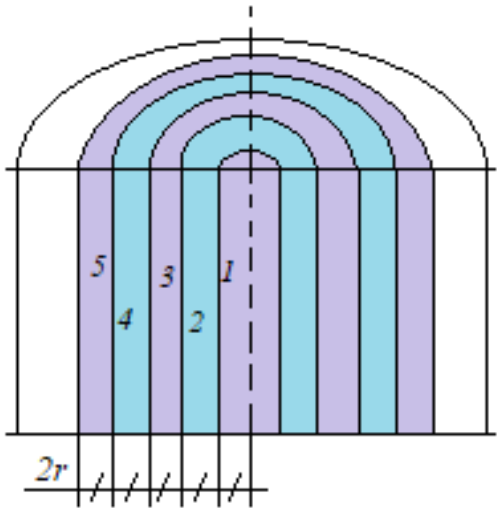


Figure 6.13. Cylindrical rings of the simulation domain for microscopic analysis

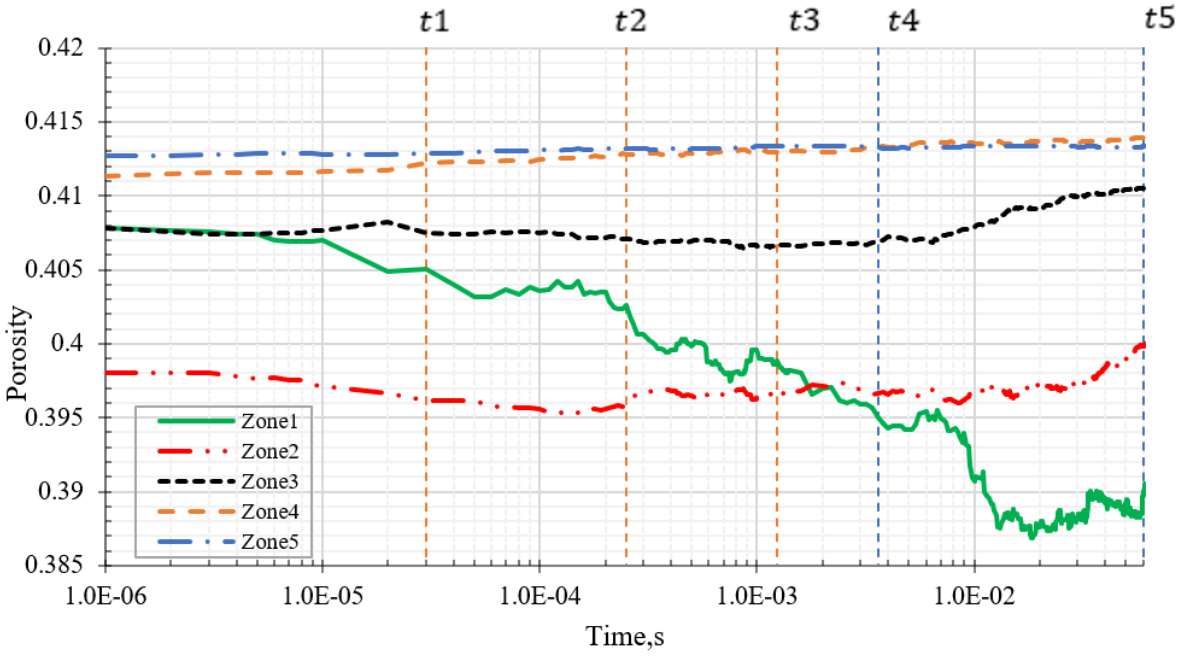


Figure 6.14. Porosity behaviour during sand production with Light Oil

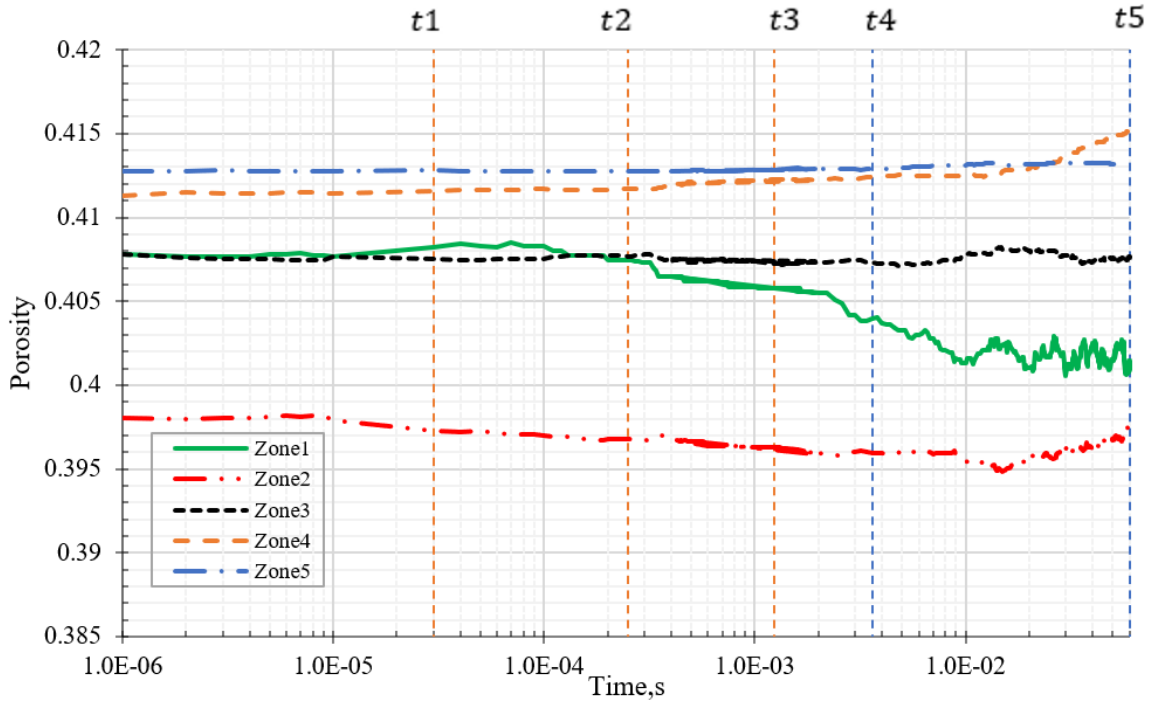


Figure 6.15. Porosity behaviour during sand production with Heavy Oil

Porosity of the granular specimen can be calculated as the ratio of the void volume over the total volume. The volume of the particles, V_p^i, located within each zone can be calculated as:

$$V_p^i = \sum_{j=1}^{m_i} V_j \tag{22}$$

where V_j is the volume of the individual particle; m_i-total number of particles in the i^{th} zone, $i = 1,2,\dots,5$. The porosity in each zone (ϕ_i) can be determined as

$$\phi_i = \frac{V_i - V_p^i}{V_i} \tag{23}$$

where V_i is the volume of the i^{th} zone. Porosity is the property of the solid particle phase, which can be approximated as the void fraction in the fluid phase that is different for each fluid cell.

Figure 6.14 and Figure 6.15 show the porosity results for light oil and heavy oil, respectively. The initial porosity of zone 2 was smallest as it formed a compacted zone of unbonded particles due to perforation damage (see Chapter 5). The porosity of zone 5 did not change significantly as it was located farthest from the perforation tunnel. The particles moved from the outer zones toward the central perforation tunnel under fluid flow effect, and this decreased the porosity of zone 1 while increased the porosity of zones 2, 3 and 4. The particle movement occurred earlier for light oil, which was associated with an earlier transient sand production. The continuous sand production however

leads to the most significant particle movement and porosity change. There were fluctuations in the porosity values in this period due to particles escape from the specimen. The porosity change is less significant for heavy oil than for light oil, which would indicate less particle movement and microstructural change for the case of heavy oil.

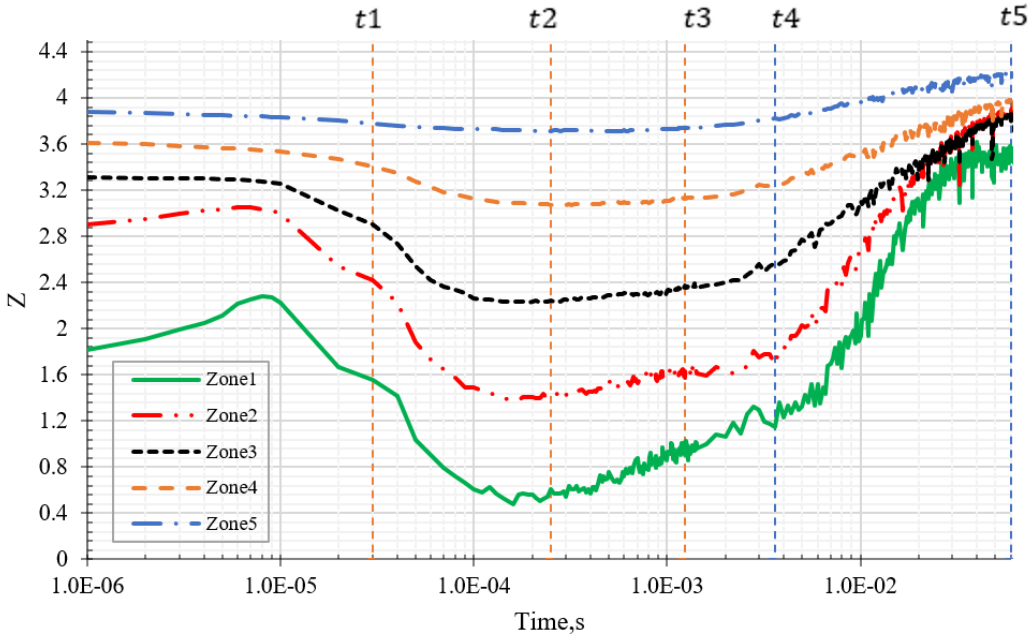
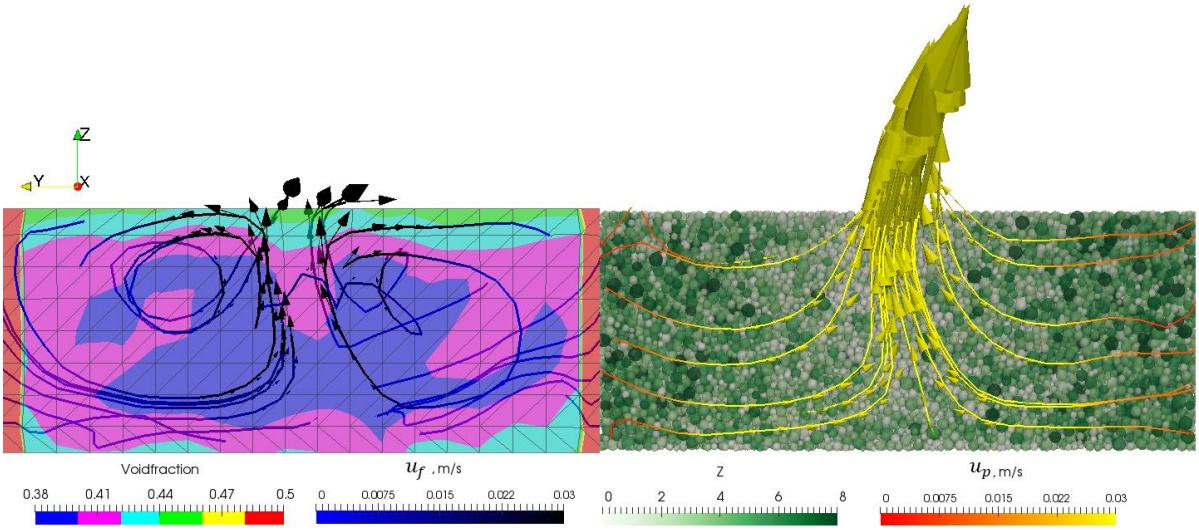
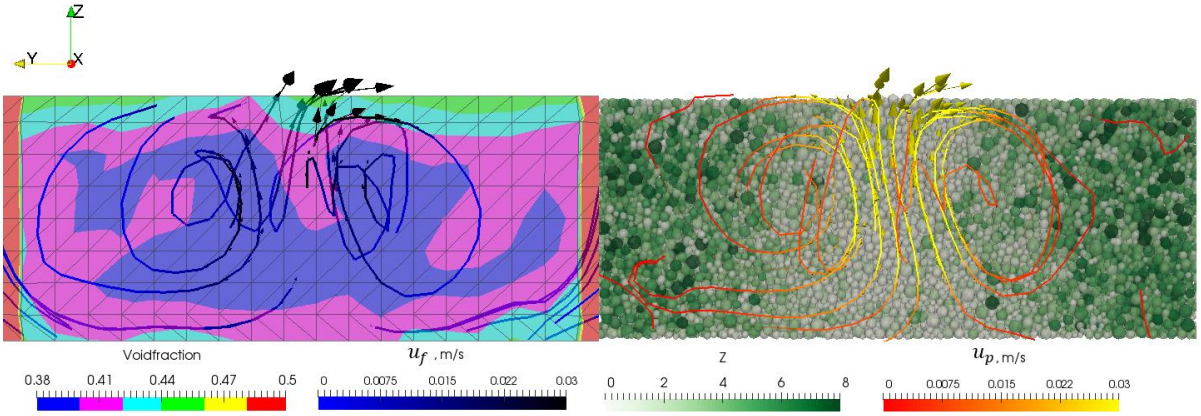


Figure 6.16. Variation of the coordination number for Light Oil

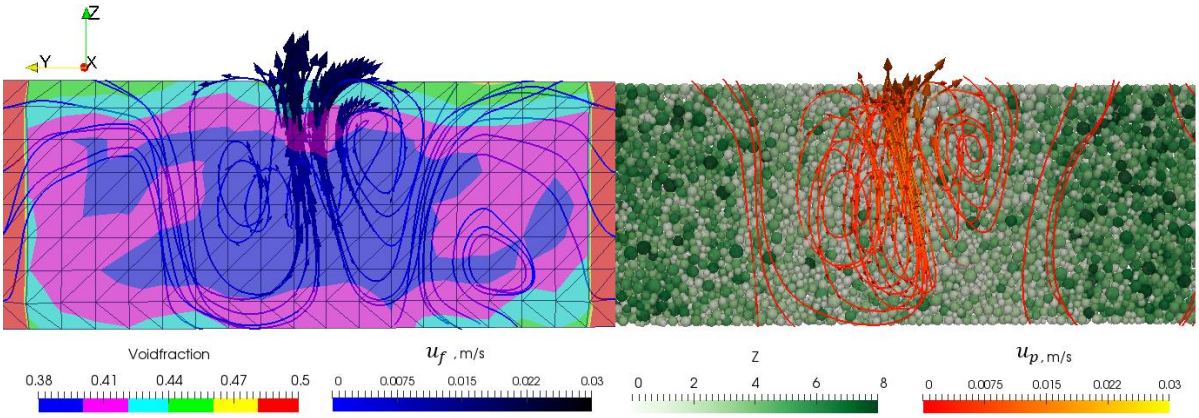
Figure 6.16 shows the variation of the coordination number calculated separately for each cylindrical ring for light oil. The initial value of the coordination number was smallest in zone 1 and increased toward the boundary until zone 5, which reflects different levels of damage away from the central perforation axis. As the transient sand production started at $t1$, the coordination number in zone 1 dropped sharply to a minimum value around $t2$, where there was also a maximum transient produced sand mass. This contact reduction is associated with a decrease in porosity, or volume reduction, in zone 1 in Figure 6.14. As the granular materials in zone 1 lost its volume and contact at the same time, it would indicate a volume collapse during the transient sand production. The coordination number results in other zones followed a similar pattern with the behaviour in zone 1 but at a lesser extent. Contact decrease occurred at nearly zero volume change for zones 2, 3 and 4. The coordination number increased for all zones during the continuous sand production starting from $t3$. The formation of new contacts would be the results of new microstructural development, although different zones underwent different positive and negative volume changes during this period.



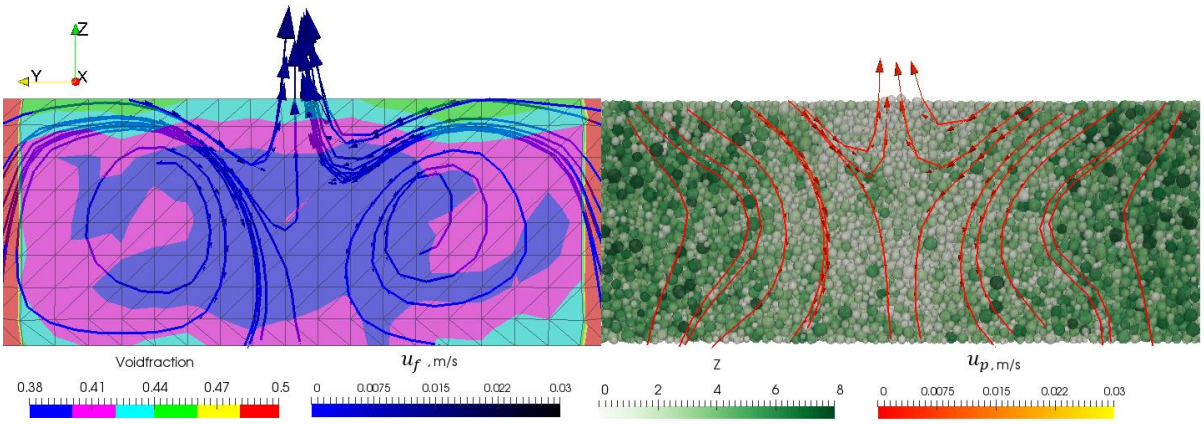
a) $time = t1$, vector scale factor = 0.015



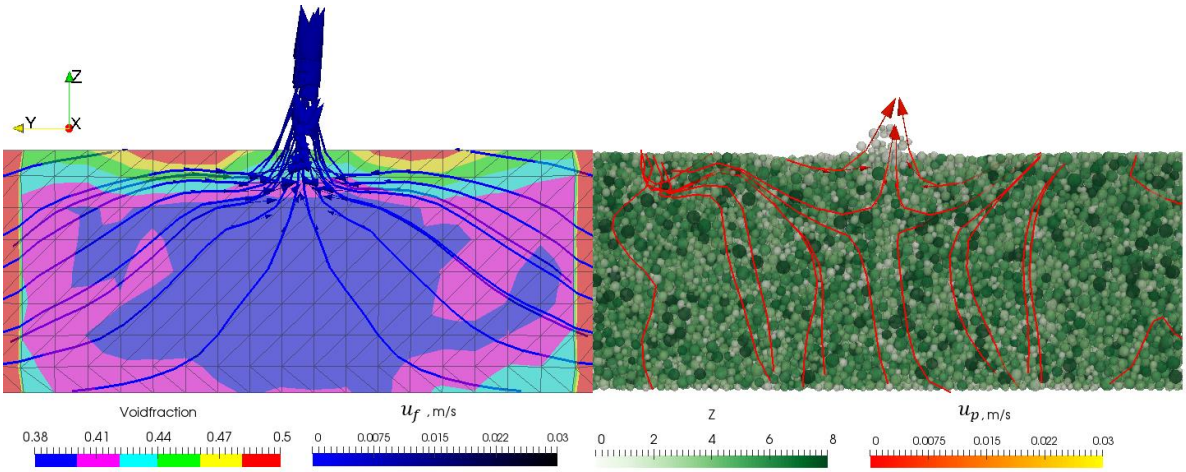
b) $time = t2$, vector scale factor = 0.015



c) $time = t3$, vector scale factor = 0.1



d) $time = t4$, vector scale factor = 0.2



e) $time = t5$, vector scale factor = 0.5

Figure 6.17. Fluid streamlines (over void fraction background, left), and particle trajectories (over particle coordination number background, right) for Light Oil

Figure 6.17 shows the fluid streamlines and the particle trajectories for light oil. The background of the fluid streamlines on the left shows the distribution of void fraction computed from the CFD simulation, which can be considered equivalent to the porosity of the solid phase. The background on the right, on the other hand, represents the particles that are colour-coded according to the coordination number values of the particles. The particle trajectories were obtained from the resultant velocity vectors of all the particles in each fluid cell, the colours here indicate the local average velocity magnitude of the particles. The void fraction distribution results show more porous layers along the boundaries due to the flat wall surfaces. The upper part of the perforation tunnel is associated with higher void fraction and parallel upward streamlines were formed along the vertical sides of the tunnel in Figure 6.17a. As the streamlines reached the top outlet, the fluid leaved the sample, but parts of the flow were diverted to the sides of the outlet and formed two recirculating flow patterns that resemble each other on the opposite sides of the perforation.

The higher void fraction zone in the upper part of the perforation seems to grow larger in Figure 6.17b, which also explains the contact loss in Figure 6.16 as a zone of light shaded particles in the central part of the sample. The particle velocity in this zone was higher than in the outer zones as the results of more significant particle movements. The higher void fraction zone disappeared in Figure 6.17c that indicates a volume collapse in the upper part of the perforation. New contacts were formed in the central zones and the particle velocities became more homogeneous at $t3$. It should be noted that there was no significant change in the void fraction distribution in the outer zones in Figure 6.17c.

The streamlines became regular in the final state in Figure 6.17e, there were two zones of high porosity near the outlet on the top surface. The particle trajectories showed nearly parallel columns of dark particles of higher coordination number inside the sample that were almost perpendicular to the fluid streamlines. The produced particles, on the other hand, came from the loose zones following the more horizontal trajectories in the region just below the top surface. It should be expected that more severe continuous sand production would continue only if the sand loss from the upper layer would trigger more disruption to the internal microstructure of the sample Figure 6.17e.

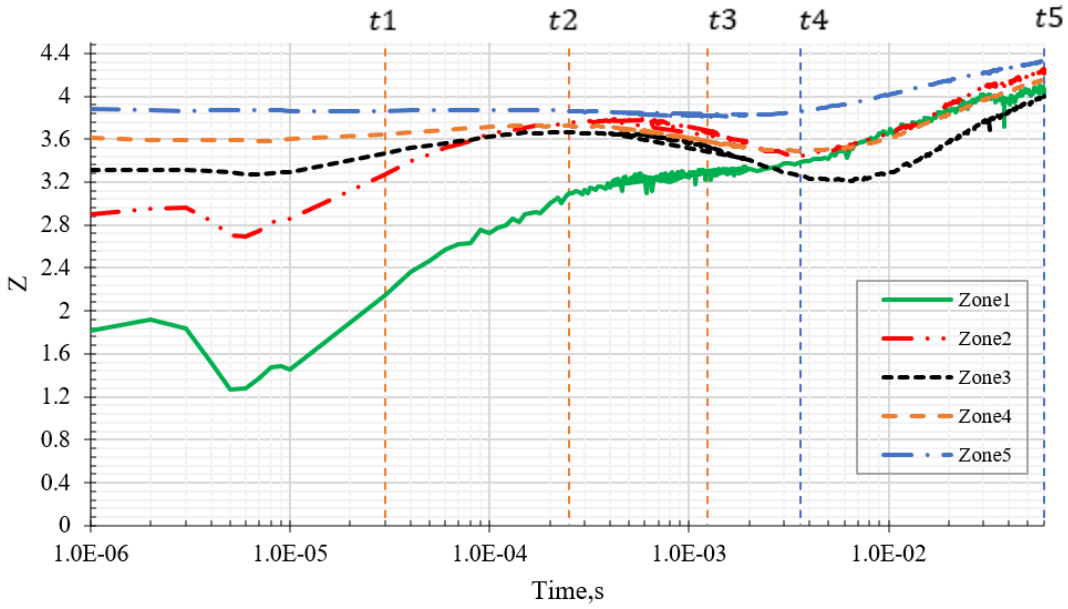
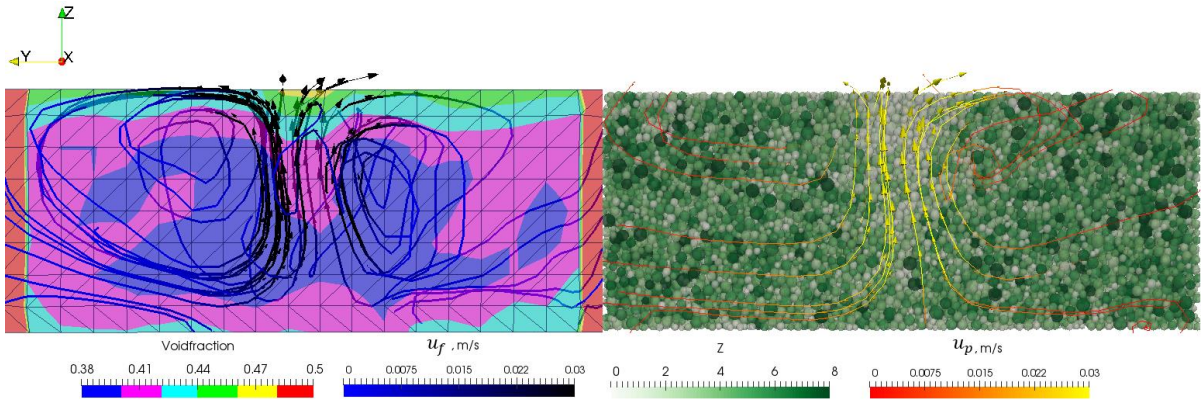


Figure 6.18. Variation of the coordination number for Heavy Oil

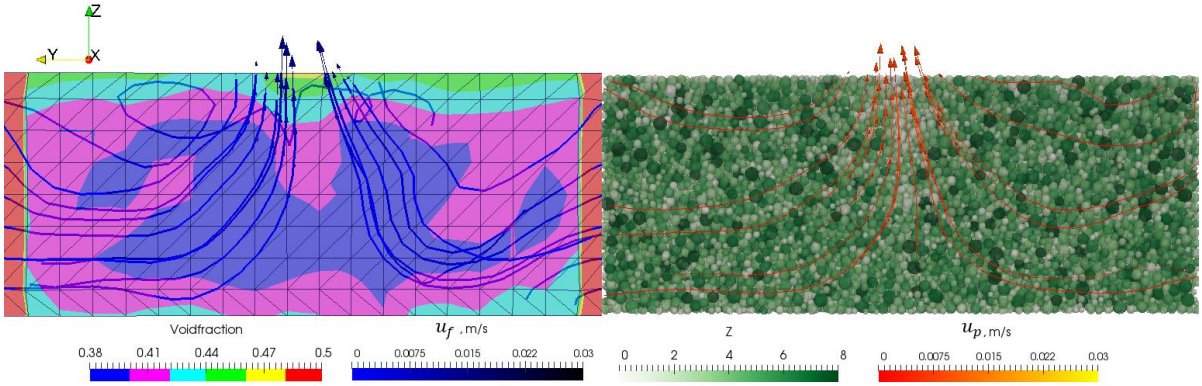
The coordination number values varied in a narrower range for the heavy oil in Figure 6.18. The patterns showed an overall increase in contact numbers as the primary trend for all zones. A small contact loss occurred close to $t4$ at the onset of the continuous sand production with the most significant loss in zone 3 and the smallest loss in zone 1, which is opposite to the result of light oil in Figure 6.16.

The fluid streamlines and the particle velocity trajectories for heavy oil are given in Figure 6.19. The void fraction and the flow patterns at $t1$ in Figure 6.19a were similar to the results of light oil at $t2$ in Figure 6.19d. A zone of higher porosity and no contact was formed in the upper part of the perforation. There were high velocity movements of both the fluid and the particles in the central zone of the sample. The streamlines and particle trajectories already became regular at the onset of the continuous sand production for heavy oil in Figure 6.19d, which was not the case for light oil. The dark particles of higher coordination number values formed discontinuous clusters around but not perfectly aligned along the trajectory curves. The zone of higher porosity in the upper perforation finally disappeared in Figure 6.19e, which would cause the contact increase in zone 1 and contact loss in zone 3 between $t4$ and $t5$ in Figure 6.18. The void fraction distribution in Figure 6.19e however showed that the void fraction decreased in the central zone was due to a volume expansion (or void fraction increase) in the outer zones, which agrees with the largest contact loss in zone 3 in Figure 6.18.

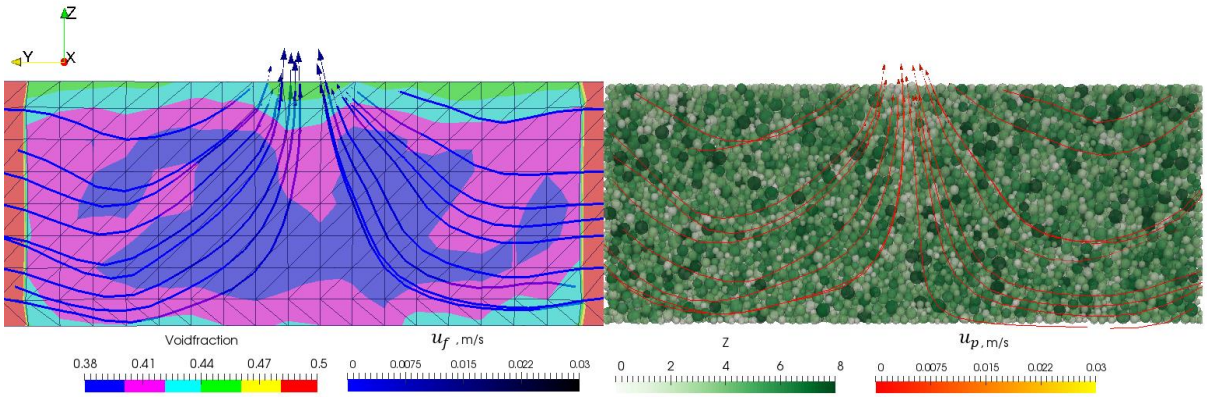
The dark particles in Figure 6.19e formed continuous chains along the trajectory curves that were oriented toward the outlet. Although the cumulative sand masses in the final state were similar for heavy oil and light oil in Figure 6.4, they were the results of different microstructures as shown in Figure 6.17e and Figure 6.19e. The continuous sand production for heavy oil mobilized a much larger proportion of the sample's particles and the particle movements were aligned much better with the fluid flow. This would cause more severe sand production by heavy oil as it was observed in other researches in the literature (Ahmad & Miskimins, 2019; Al-Awad et al., 1999; B Wu et al., 2016).



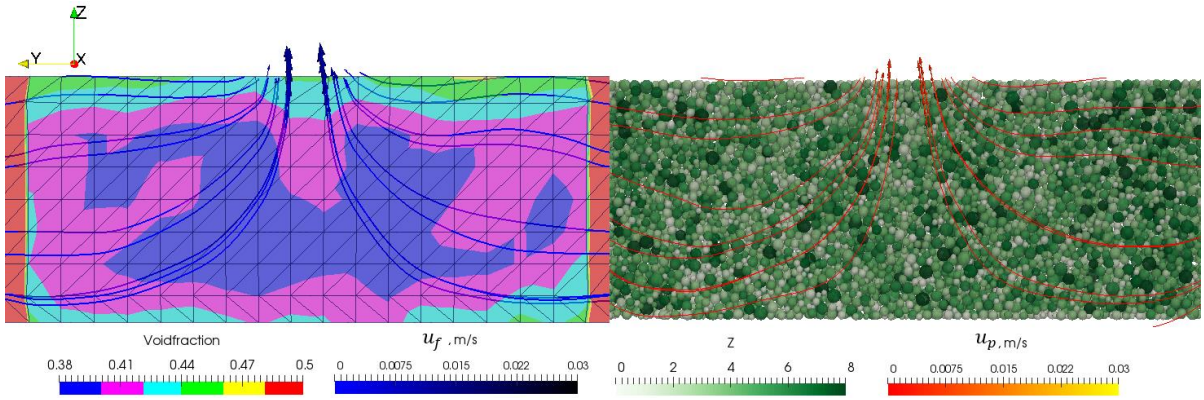
a) $time = t1$, vector scale factor = 0.015



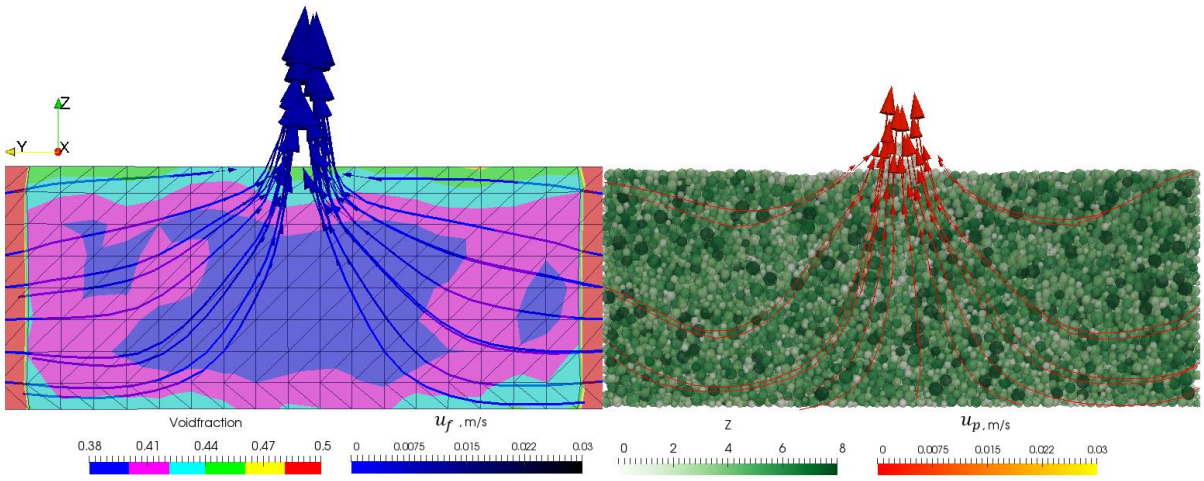
b) $time = t2$, vector scale factor = 0.09



c) $time = t3$, vector scale factor = 0.1



d) $time = t4$, vector scale factor = 0.1



e) $time = t5$, vector scale factor = 0.5

Figure 6.19. Fluid streamlines (over void fraction background, left), and particle trajectories (over particle coordination number background, right) for Heavy Oil

6.4 Conclusion

Sand production is a common problem that is associated with the production of hydrocarbons from weak sandstone reservoirs in the oil-gas industry. While the problem can be minimized by controlling the flow rate for light oil, the production of heavy oil causes more severe sand production that may need the installation of a sand control system and thus increases the operational cost. Th sanding mechanisms were investigated for different reservoir fluids using a coupled simulation with the Discrete Element Method for the solid sandstone particles and the Computational Fluid Dynamics for the modelling of hydrocarbons flow. It was found that for a short simulation time both heavy and light oils produced similar sand masses, but they were due to very different microstructural changes inside the sandstone samples. There were more bond breakages for light oil that facilitated more significant particle movement and porosity change within the damage zone around the central perforation tunnel. Sand production occurred earlier for light oil in two stages of a first transient sand production and followed by a second continuous sand production. In the final time step, sands were being produced from two loose zones in an upper layer near the outlet and hence should be limited in scope. There was only a continuous sand production for heavy oil that occurred later than for light oil. Bond breakage, particle movement and porosity change all happened to a lesser extent. The particle velocity trajectories became regular more quickly that followed the fluid streamlines toward the outlet point. The heavy oil outflow mobilized a larger proportion of the sample's particles and would lead to more severe sand production. The research on the impact of different reservoir fluid flows on sand production is being extended by investigating the effect of multiphase flow of different reservoir fluids on sand production, the results of which will be shown in the next Section.

CHAPTER 7 - MICRO AND MACROSCOPIC BEHAVIOR OF SAND PRODUCTION WITH MULTIPHASE FLUID FLOW

In this Chapter, the two-phase fluid CFD-DEM model was developed to investigate the sanding mechanisms, where the DEM model was adopted from Chapter 5. Two different DEM samples with overburden stress $3\ MPa$ and $5.9\ MPa$ were used to mimic laboratory experiments and reservoir development. The necessity of using multiphase fluid models for more truthful modeling is presented. Heavy oil, water, and two types of carbon dioxide were used injection and saturating fluids. The influence of coupling frequency N on the accuracy and efficiency of the CFD-DEM model was also discussed.

7.1 Introduction

This Section presents results from the numerical sand production simulations on weak sandstones with the different two-phase fluid flows: gas injection into the water-saturated sample; gas and water injection in the oil-saturated sample. The fingering affect for various injection fluids is also shown.

7.1.1 Sand production in Kazakhstan

The hydrocarbon production oil fields in Kazakhstan, such as Karazhambas, North Buzachi, Kalamkas Zhalgiztobe, and Kenkiyak, are accompanied by sand production ("Kazakhstan Upstream Oil and Gas Technology and R&d Roadmap," 2013). The local oil field (let us call it NN field) with significant sand production was chosen to detect reservoir conditions in which sand production has occurred.

The NN field has weekly consolidated sandstone reservoirs. These production reservoirs are grouped into three "Objects" (1,2,3) by increasing depth and age. The NN permeability is relatively low and varies from 0.175 to $0.500\ Darcy$. The reservoir oil is classified as heavy oil with viscosities ranging from 375 to $550\ cP$. Up to date, the oil field is being actively developed using secondary (water flooding) and enhanced oil recovery (hot water and steam flooding) techniques.

The oil and gas historical production data of NN field over the 36 years were analyzed in order to determine the major horizons with high value of sand production. All wells were sorted by cumulative sand value during the whole well life. Top 100 wells depend on the horizon shown in Figure 7.1.

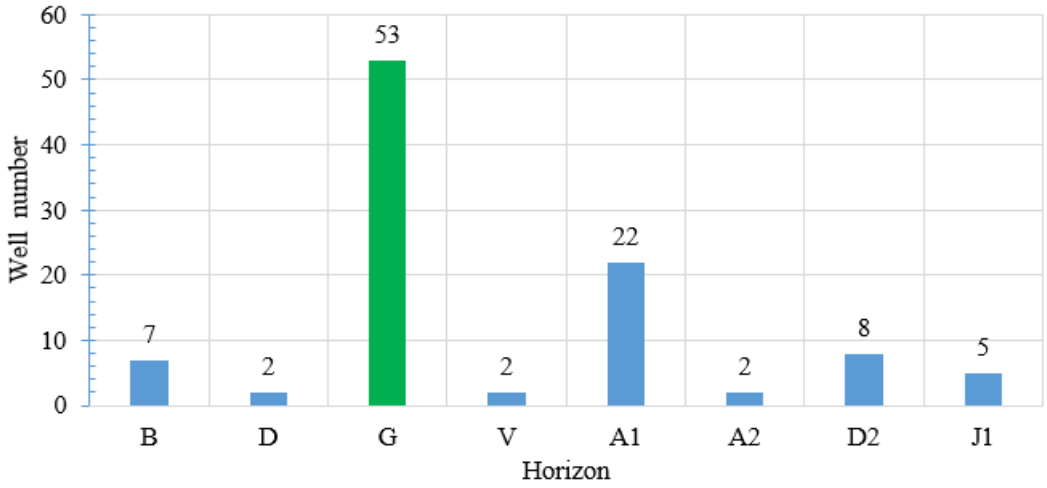


Figure 7.1. Top 100 wells distributed by horizons

Figure 7.1 shows the distribution of 100 wells with high cumulative sand production by horizons. The horizons G and $A1$ contain the 75% wells from the whole list. The horizon G with the largest number of wells will be used for further analysis.

Kazakhstan oil fields with weakly encoded terrigenous reservoirs usually consist of highly permeable and highly porous sandstones with active water. Strata are formed within highly permeable and highly porous sandstones with initial flooding (usually above 20%), low depth, viscous oil, etc.. Newly drilled wells have an initial watercut that increases during the well operation period(Abishev et al., 2018; Collins et al., 2008; Nie et al., 2010).

The next step of analysis is to determine the initial fluid conditions in production horizons. The parameter watercut can be expressed in the following way:

$$WC = V_{water}/V_{total} \tag{7.1}$$

where V_{water} is volume of water and V_{total} is total volume of produced liquid.

The 53 wells of the G horizon were distributed by initial watercut (water saturation). The results of the distribution are shown in Figure 7.2. Most of newly drilled wells (36) have initial watercut less than 20% and only six wells produced pure oil. It means that 88,68% newly drilling wells have no zero initial watercut, and produced fluid is multiphase. The distribution of sand rates of 53 wells also was considered (Figure 7.3). The 46 wells with non-zero watercut produce about 97% of total sand mass and only 3% of sand mass comes from the pure oil wells. Obtained analysis results are fully consistent with the literature review that the sand production phenomenon in Kazakhstan occurs in a multiphase fluid flow conditions (Collins et al., 2008).

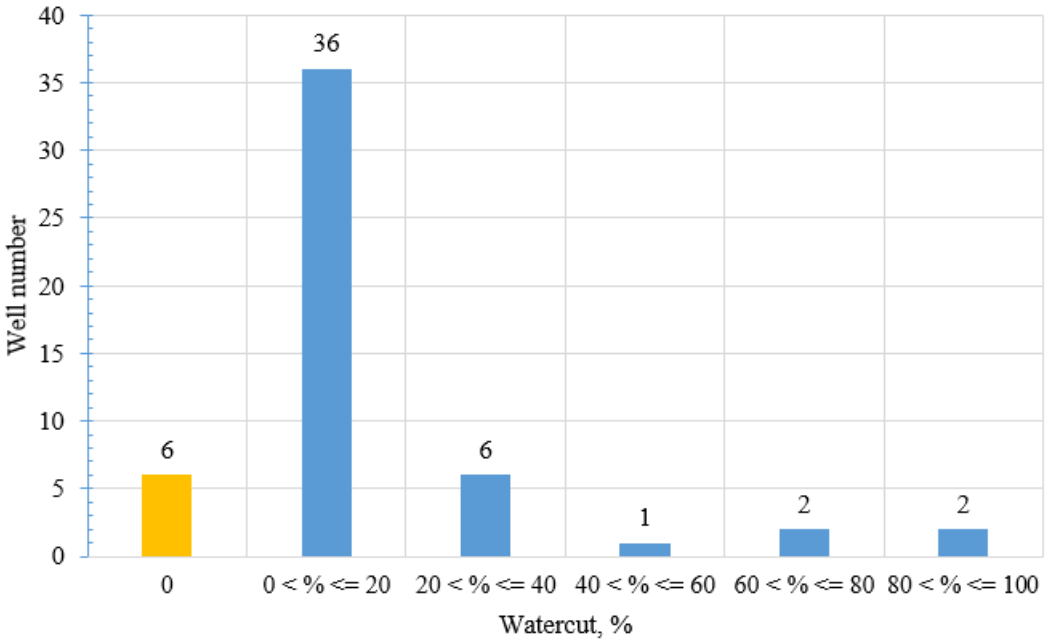


Figure 7.2. The distribution of G horizon wells depend on the WC

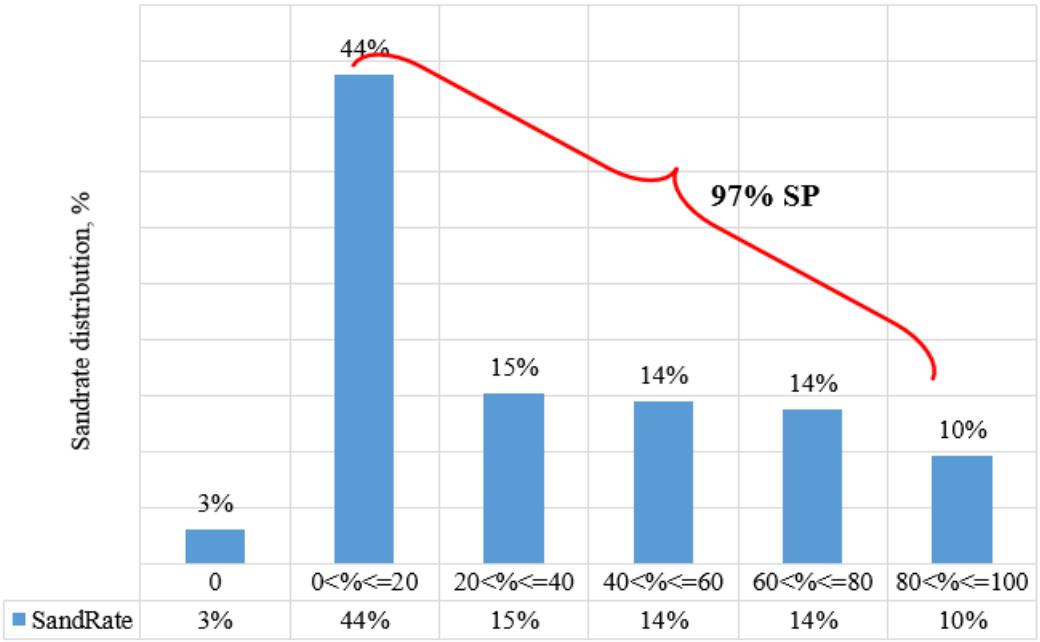


Figure 7.3. The sand production distribution depends on WC

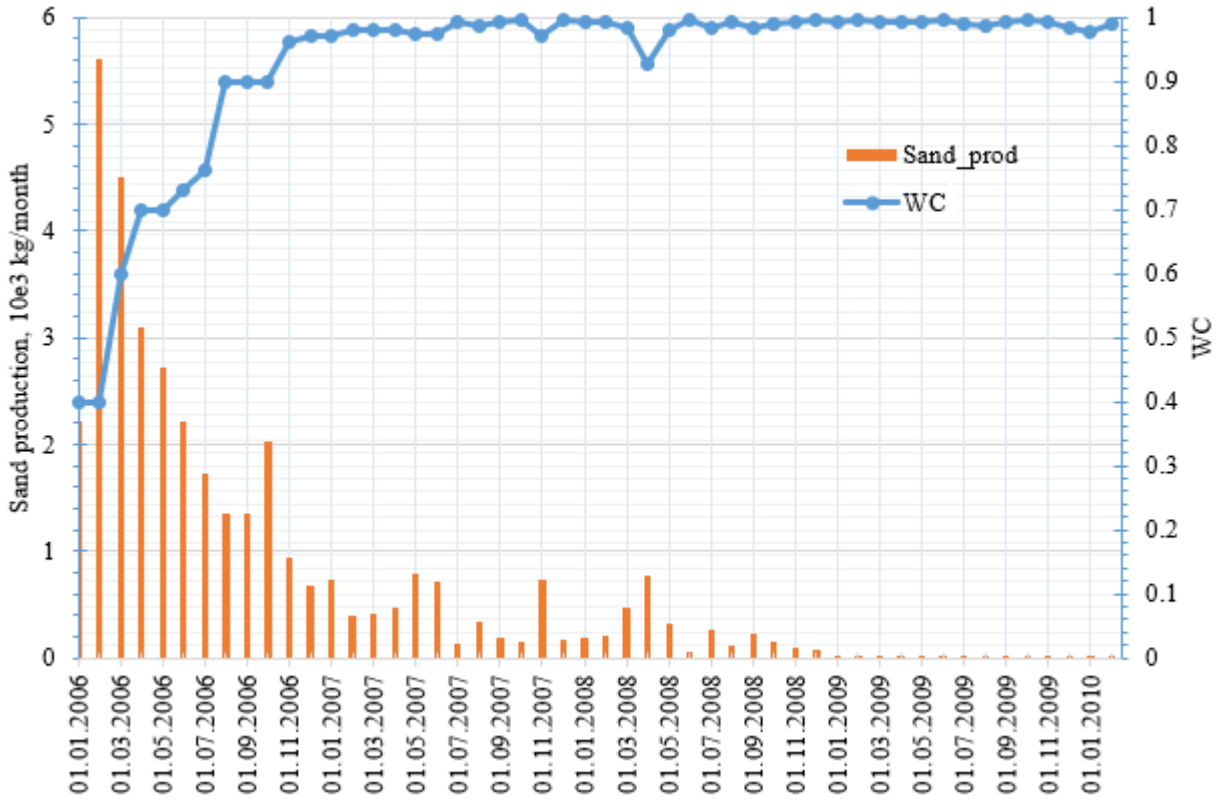


Figure 7.4. The sand production history of $Nwell$ in NN field

The typical sand production behavior of field NN can be found in Figure 7.4. The sand rate achieves a maximum value at the beginning of liquid production and is gradually decreasing after. The constant flowrate of produced liquid conditions is maintained. For all wells, the watercut increases during the well operation period. This decreasing of sand production behaviour is characterized as transient sand production. Another template of sand production can be found in the study Kozhagulova et al. (2020). The sand rate can sufficiently increase during the shutdown operations.

7.2 Model building

This Section continues to follow the CFD-DEM formulations of so-called *model A* (Norouzi et al., 2016; Zhou et al., 2010). The governing equations that describe the motion of a multiphase fluid ($phase1$ and $phase2$) in the presence of a secondary particulate phase can be given by a modified set of Navier-Stokes's equations and by the well-known interface tracking VOF method (Enwald et al., 1996; Hirt & Nichols, 1981; Nichols & Hirt, 1975). The governing equations are described in Section 3.3.

7.2.1 CFD-DEM simulations

The numerical simulations of dry samples in this study fully repeat sample preparation in Section 5.1 which is which in turn followed the laboratory steps of consolidation, cementation, perforation, and sand production (Kozhagulova et al., 2020a, 2020b). The numerical particles were generated randomly inside a cylindrical wall and let fall under gravity to the bottom wall of the space to simulate a pluvial deposition process. The number and the size of the particles were specified to follow Section 5.2. The samples slowly compressed to achieve the vertical stress are equal to $\mathbf{3}; \mathbf{5.9}\ \boldsymbol{MPa}$. The diameters (*D*) of the compressed samples are equal to $\mathbf{15.12}\ \boldsymbol{mm}$, and the heights (*H*) correspondingly are equal to $\mathbf{5.826}; \mathbf{5.727}\ \boldsymbol{mm}$ (Figure 5.2). The intact sample was then subjected to perforation and sand production that created damages to its initial structure, and this process was captured in the microscopic simulation.

The numerical simulations in this study were executed on a High-Performance Computer System, which was equipped with $\mathbf{26}$ multi-threading Intel® Xeon(R) Gold 6230R CPU with a clock speed of $\mathbf{2.10}$ GHz and $\mathbf{500}$ gigabytes of memory. A standard sand production simulation was carried out for about $\mathbf{9}$ days to complete $\mathbf{0.1}\ \boldsymbol{s}$ physical time even with the parallel computation that utilized as $\mathbf{8}$ threads using the MPI protocol. It was found that the raising of number threads more than $\mathbf{8}$ does not significant increase the calculation speed.

The perforated dry sample (Figure 7.5a) was initially saturated by different fluids (Table 7-2) and further used for sand production simulation where different fluids were injected radially from the external boundary towards the vertical perforation tunnel at the center as similar to the experimental conditions (Kozhagulova et al., 2021). The properties of fluids used in these numerical experiments are described in Table 7-1. The heavy oil and water in this study were designed as incompressible fluid with invariable properties. At the same time, carbon dioxide gas was considered in two different states. The first state $\boldsymbol{CO2\_1MPa}$ corresponds to $\boldsymbol{CO2}$ gas at surface conditions (temperature is $\mathbf{20}$℃, under the pressure $\mathbf{1}\ \boldsymbol{Atm}$). The second – to the reservoir conditions of Object $\mathbf{2}$ of Karazhanbas oil field (Collins et al., 2008).

Table 7-1. The fluids properties (Bealessio et al., 2020; Collins et al., 2008; Wischnewski, n.d.)

Reservoir fluid	**Kinematic viscosity, m^2/s**	**Density, m^3/kg**
Heavy oil (HO)	$161.796 \cdot 10^{-6}$	927.09
Water	$1 \cdot 10^{-6}$	1000
CO2_1MPa	$8.36 \cdot 10^{-6}$	1.836
CO2_4.5MPa	$0.147 \cdot 10^{-6}$	110.529

In this part of the study, the different numerical simulations were considered in order to mimic laboratory experiments (Shabdirova Ainash et al., 2021) and recovery techniques (primary and secondary) in reservoir developments (Alvarado & Manrique, 2010; Fanchi, 2002). Three numerical simulations with overburden stress $\mathbf{3}\ \boldsymbol{MPa}$ were conducted to repeat laboratory experiments. Other cases reproduced the implementation of recovery technics in the poorly consolidated reservoir. The reservoir properties of Object **2** of the Karazhanbas oil field had chosen for numerical simulations. The numerical simulations details are shown in Table 7-2. The sand production simulations were conducted using the CFDEM®coupling program (Christoph Goniva et al., 2012). All cases described in Table 7-2 used a two-phase fluid model.

Table 7-2. Simulation's information

№	**Case_name**	**Injection fluid**	**Reservoir fluid**	**Overburden stress, MPa**	**Reservoir pressure, MPa**
1	$\boldsymbol{CO2-W}$	CO2_1MPa	Water	3	1
2	$\boldsymbol{Water\_1MPa}$	Water	Water	3	1
3	$\boldsymbol{CO2\_1MPa}$	CO2_1MPa	CO2_1MPa	3	1
4	$\boldsymbol{CO2-HO}$	CO2_4.5MPa	Heavy oil	5.9	4.5
5	$\boldsymbol{W-HO}$	Water	Heavy oil	5.9	4.5
6	$\boldsymbol{CO2\_4.5MPa}$	CO2_4.5MPa	CO2_4.5MPa	5.9	4.5
7	$\boldsymbol{Water\_4.5MPa}$	Water	Water	5.9	4.5

The CFD grid and boundary conditions. The CFD grids have been built using HELYS-OS graphical user interface (ENGYS Ltd, n.d.). Similar to Section 6.2.1, the constructed mesh was tested and satisfied with the stability conditions. The computational fluid domains in Figure 7.5b have $\mathbf{2344}$ cells that form the same size as the initial sample in Figure 7.5a. Although the numerical samples height is different for $\mathbf{3}$ and $\mathbf{5.9}\ \boldsymbol{MPa}$ overburdens stress this is not affecting the final number of CFD cells.

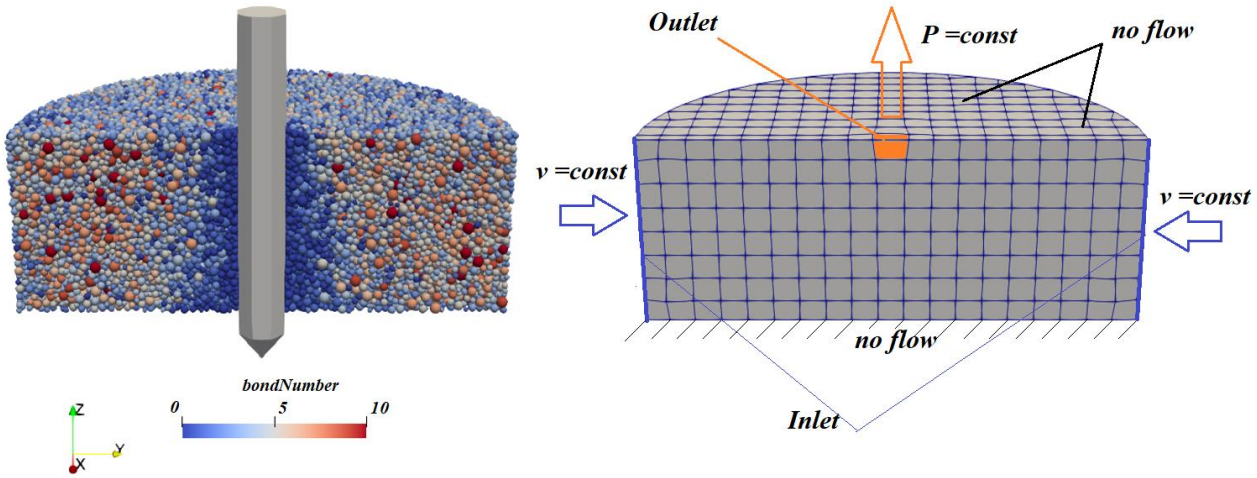


Figure 7.5. a) The bond number distribution during the dry sample perforation. b) CFD simulation domain and boundary conditions

The CFD-DEM system boundary conditions are illustrated in Figure 7.5b (Table 4-1, sets $\boldsymbol{III-IV}$). The top and the bottom walls were set as impermeable walls with no-slip boundary conditions. There is an outlet in the centre of the top wall. The different fluids are also injected from the whole side wall at a constant velocity $\boldsymbol{1.39 \cdot 10^{-2}; 2.83 \cdot 10^{-2}\ m/s}$. The outlet pressure is maintained at a constant $\boldsymbol{P = 1; 4.5\ MPa}$. The fluids in simulations are considered incompressible.

The interactions between the fluid and the particle phases were simulated by a four-way coupling method (Norouzi et al., 2016). The motion of particles affects the motion of fluid and vice versa. The average solid volume fraction in this numerical study is about $\boldsymbol{0.58}$. The coupling information was given in terms of the pressure force, the viscous force and the Di Felice (Di Felice, 1994) drag force (Di Felice, 1994; Zhou et al., 2010). The discretization time steps for DEM and CFD are correspondingly $\boldsymbol{2 \cdot 10^{-8}}$ and $\boldsymbol{2 \cdot 10^{-7} s}$. A CFD simulations were run for every $\boldsymbol{10}$ time steps of the DEM simulation and the coupling data were exchanged accordingly. The choosing of coupling number $\boldsymbol{10}$ is ensuring sufficient accuracy and significant efficiency of commutation.

7.2.2 Optimal coupling number for CFD-DEM system

Usually, the DEM timestep is substantially smaller than CFD (Christoph Goniva et al., 2010, 2012; Zhao & Shan, 2013). For every N (integer value) DEM timestep, CFD can perform 1 calculation. It can save the computational time i.e., the CFD simulation run for every N ($N > 1$) timesteps of the DEM simulation and the coupling data exchanged accordingly. Of course, it still possible to run CFD solver at each DEM timestep, but in practice it will sufficiently increase computation time and only slightly improve the accuracy of CFD calculations.

Goniva et al. (2012, 2010) investigated the granular materials properties using $N = 10{,}100$ for CFD-DEM system (Christoph Goniva et al., 2010, 2012). Zhao et al. (2013) presents a CFD–DEM approach to simulate the behavior of fluid–particle interaction for applications relevant to mining and geotechnical engineering. The authors' numerical experience shows that for each CFD computing step, exchanging information after $N = 100$ steps of DEM calculation will ensure sufficient accuracy and efficiency. If the time steps for DEM and CFD are sufficiently small, more steps for DEM are also acceptable (Zhao & Shan, 2013).

Climent et al. (2014) have built the $3D$ numerical model based CFD-DEM to simulate sand production around perforations. The basic idea is that the coupling interval should be small enough to resolve the desired coupling behavior. Coupling information should be exchanged several times as a particle moves across a fluid element. The authors have used the following condition (Climent et al., 2014):

$$C^{crit} = \frac{\Delta x_{cfd}}{|u_p| t_c} > 3 \tag{7.2}$$

where t_c is the coupling time and u_p the particle velocity.

In this study in order to find optimal value of $\boldsymbol{N}$, the CFD-DEM simulations were run for every $\boldsymbol{N = 1, 10, 50, 100}$ time steps of the DEM simulation and the coupling data were exchanged accordingly. The results of simulations were compared from to accuracy and computation speed points of view.

The main assumption is that $N = 1$ case is the sand production simulation test with maximum accuracy and biggest simulation time, and all another simulation results for $N > 1$ will be faster but less accurate. The optimal value N should be satisfying the minimum simulation time where the relative errors of parameters (average particle velocity, drag, pressure and viscous forces) is still in valid intervals.

The CFD time step in the model described in Section 7.2.1 with boundary conditions (Table 4-1, set *III*) have been set as $2 \cdot 10^{-8}, 2 \cdot 10^{-7}, 10^{-6}, 2 \cdot 10^{-6}\ s,$ which are correspondingly equal to $N = 1{,}10{,}50{,}100$. The sample was initially saturated by oil. The injection fluid is also oil. The total physical time of simulations is $0.0002\ s$. The interactions between the fluid and the particle phases were simulated by a four-way coupling method (Norouzi et al., 2016). The simulations results are tabulated in Table 7-3.

Table 7-3. Comparison results

Coupling number(N)	Simulation time, day	Speed up	Relative error, %			
			Particle velocity	Drag force	Viscous force	Pressure force
1	0.2386	0%	0	0	0	0
10	**0.1620**	**47%**	**0.31**	**0.48**	**0.29**	**1.62**
50	0.1584	51%	1.66	1.91	1.57	6.50
100	0.1343	78%	2.37	3.47	3.45	10.70

The coupling number $N = 10$ increases the speed of calculations in 47% and at the same time the relative error in assessment parameters does not exceed the value 1.62%. The other cases can give valuable speed up but the error increase too. The choice of coupling number $N = 10$ is ensuring sufficient accuracy and significant efficiency of commutation.

7.3 Simulation results

7.3.1 Sample overburden stress is $3\ MPa$

The dry sample (Figure 7.5a) was saturated by water and displaced by $CO2$ gas. The numerical simulations were executed in order to mimic laboratory experiments (Shabdirova Ainash et al., 2021). Three numerical simulations with overburden stress $3\ MPa$ were conducted to repeat laboratory experiments. The boundary conditions for CFD-DEM systems are established in the following way as set III (Table 4-1). The physical simulation time is about $0.074\ s$. The simulations results of $Water\_1MPa$, $CO2\_1MPa$ and $CO2\_W$ cases (Table 7-2) are shown in Figure 7.6 - Figure 7.9.

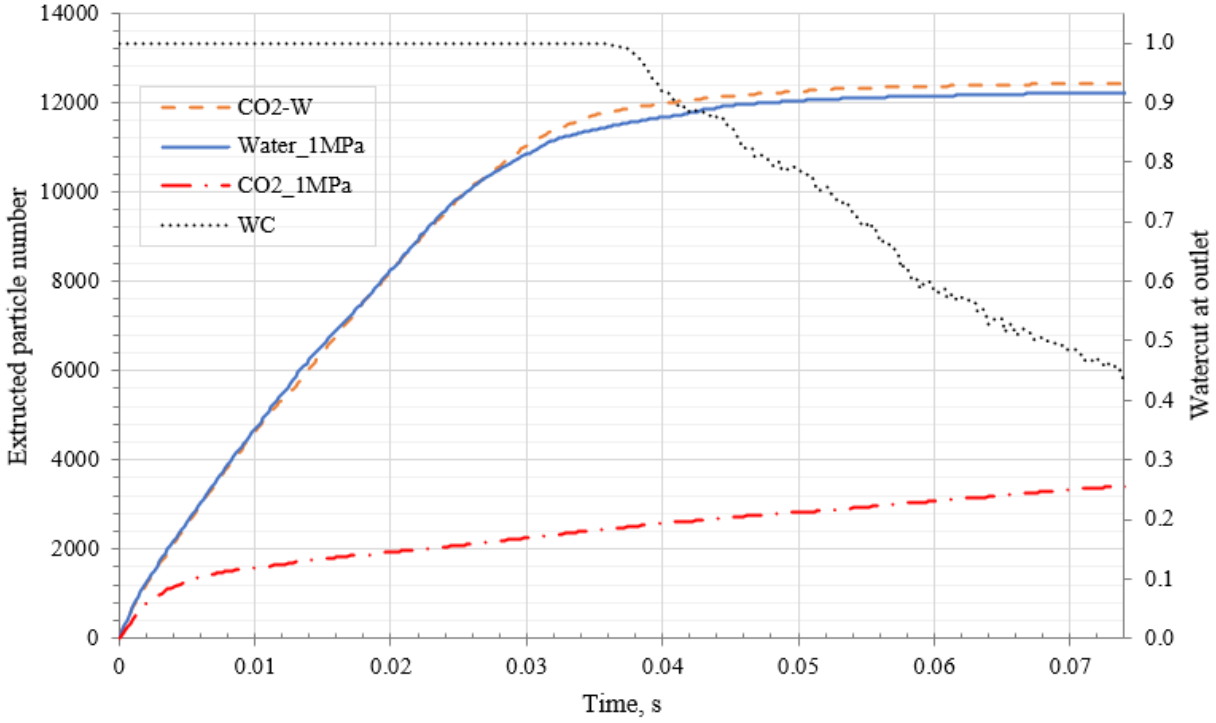


Figure 7.6. Extracted particles for different cases. Overburden stress 3 MPa

The highest extracted particles value observed in case of $CO2\_W$. First gas breakthrough in outlet cell has occurred at $t = 0.0156$ and increased during the simulation. At the end of simulation of $CO2\_W$ the water cut decreased up to 0.4346. The difference between $CO2\_W$ and $Water\_1MPa$ in extracted particles increases with the increase of gas concentration. It should be noted that for all numerical simulations in this study, the constant injection flowrate is used for the inlet (Figure 7.5b). In contrast, in the experimental setup, constant pressure depletion was used. The deviation in extracted particles between $CO2\_W$ and $Water\_1MPa$ will only increase with changing of boundary conditions to constant pressure depletion. In general, the sand production behavior for $Water\_1MPa$, $CO2\_1MPa$ and $CO2\_W$ cases correspond to the transient sand production pattern (Veeken et al., 1991).

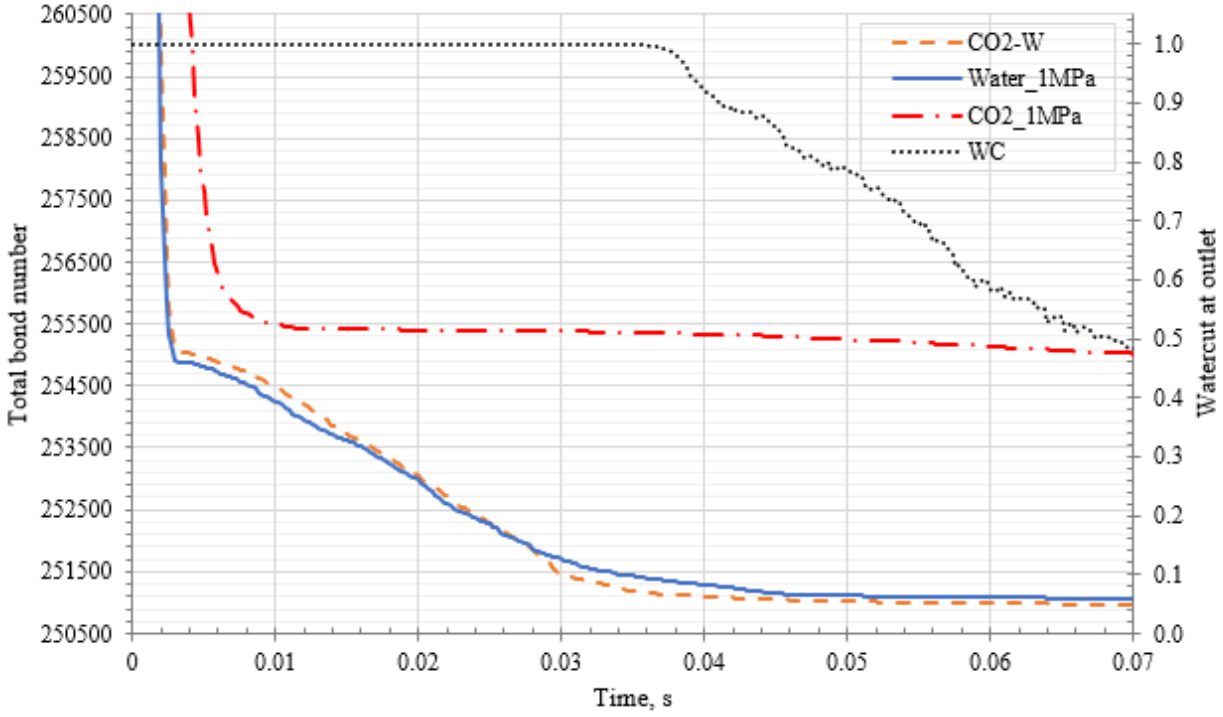


Figure 7.7. The total bond numbers. Overburden stress 3 MPa

Figure 7.7 shows the bond numbers behavior versus time and watercut. The bond number characterized the total number of cemented contacts of the sample. A constant bond number value in the sample means that the sample contacts no longer break, and the system cannot form new free particles, which can be further extracted from the sample due to fluid influence. In cases $CO2\_W$ and $Water\_1MPa$ this parameter quickly decreases up to the time $t < 0.03\ s$ and becomes more constant after. The total number of cemented (bond) contacts in $CO2\_1MPa$ cases are greater than others, what causes a greater sand production (Figure 7.6).

The voidfraction distributions of $Water\_1MPa$, $CO2\_1MPa$ and $CO2\_W$ are shown in Figure 7.8. The zones are marked in green color indicate fluid cells without particles. The red zones correspond to the initial state of the sample with porosity 0.37. The particles extraction of $Water\_1MPa$ and $CO2\_W$ samples have a similar $3D$ pattern, and the particles were removed mainly from the plastic zone, formed under perforation processes. The elastic zones remained unchanged (see Section 5). In case of $CO2\_1MPa$ particles were extracted only from the top of the sample.

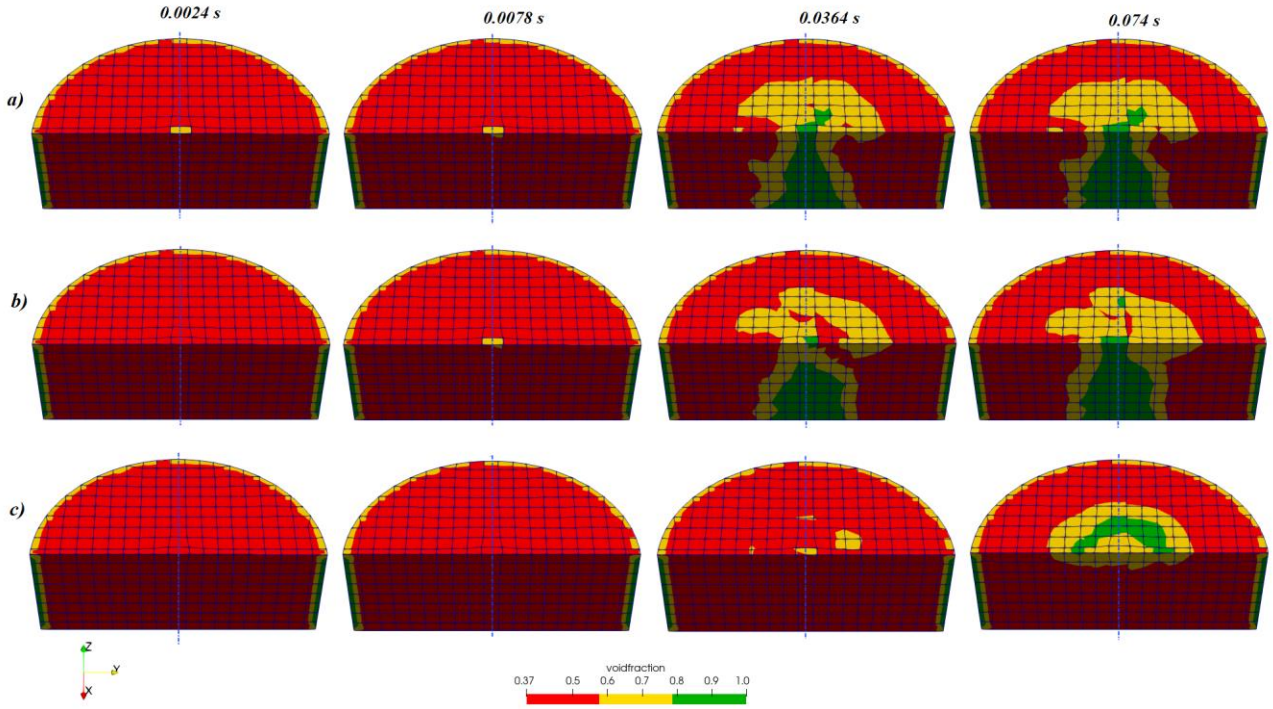


Figure 7.8. The $3D$ voidfraction distribution vs time. a) $CO2\_W$; b) $Water\_1MPa$; c) $CO2\_1MPa$

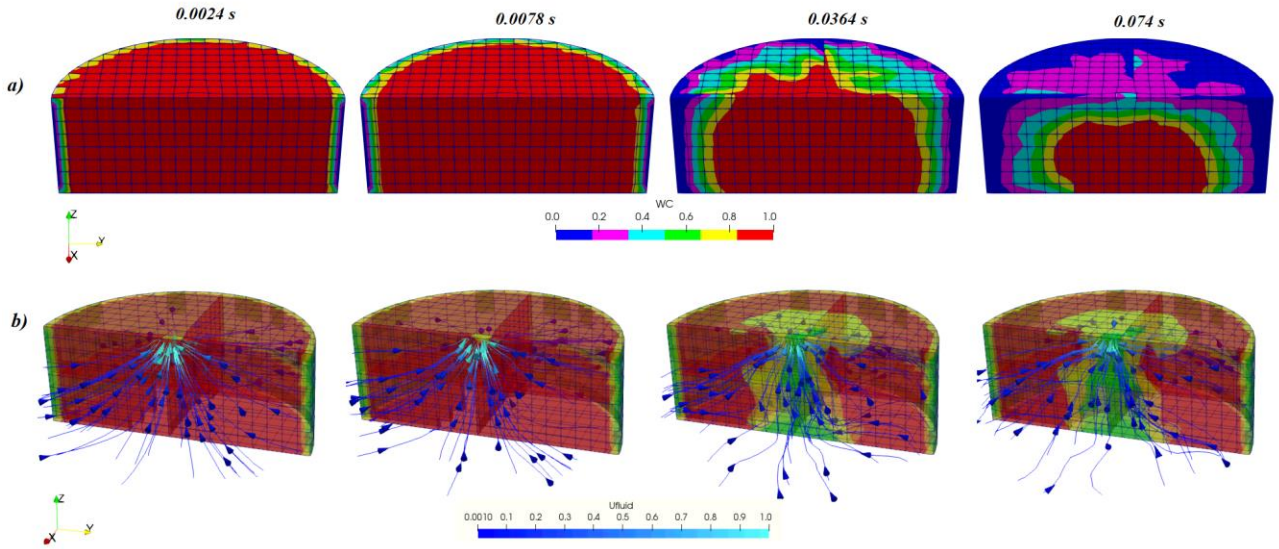


Figure 7.9. $CO2\_W$ simulation results. a) Watercut; b) Fluid streamlines (over voidfraction background)

The watercut distribution and fluid streamlines of the $CO2\_W$ case are illustrated in Figure 7.9. The carbon dioxide gas was injected into the sample, which was initially saturated by water. Dark blue zones in the watercut screens indicate the gas, while the red zone is water (Figure 7.9a). The

more viscous gas uniformly displaced the water without any fingering effect (Table 7-1). The dimensionless parameter M is a ratio between displaced and injected fluids, in this case, equal to 0.12 (Doorwar & Mohanty, 2017; Kargozarfard et al., 2019; Lenormand et al., 1988; Moortgat, 2016).

Validation of model with laboratory experimental data

The validation of the two-phase fluid CFD-DEM model with laboratory experiment was executed by Shabdirova et al. (2021) using $CO2\_W$ simulations results**.** By analogy with Section 5.3.2, the numerical and experimental results were compared in dimensionless form. The dimensionless cumulative sand productions versus dimensionless time are shown in Figure 7.10. In this part of the validation study, there is no comparison watercut behavior due to the absence of these data in laboratory experiments.

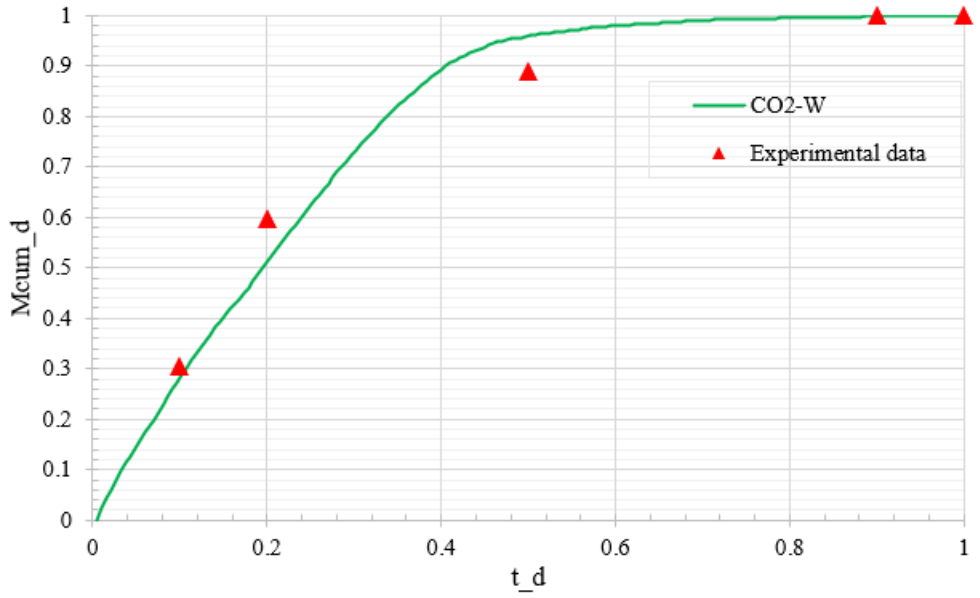


Figure 7.10. Numerical and experimental results comparison. Dimensionless cumulative sand production

Figure 7.10 shows cumulative sand production profiles of numerical and experimental results. A similar pattern of decreasing sand production increments of transient sand production is observed. There is a good matching between experimental data and numerical results. It should be noted that numerical and experimental studies have different fluid boundary conditions, geometrical scale, simulation time and comparison only qualitatively. In future works, in order to conduct quantitative analysis, the development of a new CFD-DEM model with many particles and complex geometry is planned.

7.3.2 Sample overburden stress is $5.9\ MPa$

In this Section, the implementation of secondary recovery methods: gas and water injections in the poorly consolidated reservoir, which is initially saturated by heavy oil, were represented (Alvarado & Manrique, 2010; Fanchi, 2002). The Object 2 of the Karazhanbas oil field's reservoir properties was chosen for initial and boundary conditions of CFD-DEM simulations. The Object 2 is at a depth $290\ m$ with initial reservoir pressure $4.5\ MPa$ and temperature 27℃. The average calculated stress for this Object is found as $5.9\ MPa$ (Collins et al., 2008). The fluid properties and simulations details are tabulated in Table 7-1 and Table 7-2. The simulation results of the CFD-DEM model were not verified due to the lack of laboratory experiment data or other data in the literature.

The oilcut is the fraction of oil in CFD cell (Figure 7.5b) can be calculated by:

$$OC = V_{oil}/V_{total} \tag{7.3}$$

where V_{oil} is volume of oil and V_{total} is total volume of CFD cell.

The cumulative sand production behaviors of $Water\_4.5MPa, CO2\_4.5MPa$, $CO2\_HO$ and W_HO cases are shown in Figure 7.11. The modeling time $t = 0.05$ s corresponds to the event of sand production has stopped for all simulation cases.

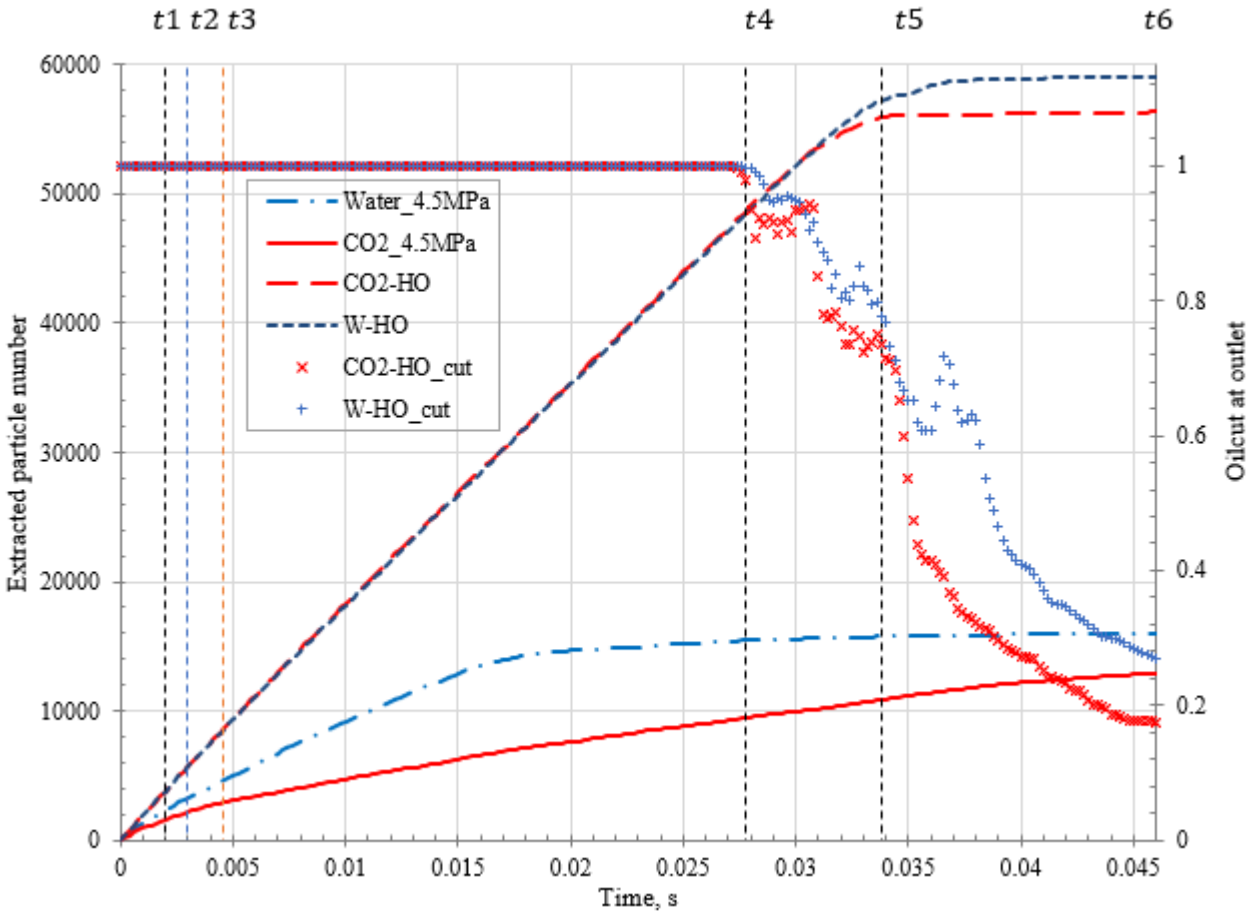


Figure 7.11. Extracted particles and oilcut vs time. Overburden stress 5.9 MPa

The sand production of $CO2\_HO$ and W_HO profiles have a similar pattern up to time $t \leq 0.03\ s$, and this behavior can be characterized as before, which is related to the heavy oil (see Section 6). Both cases start to differ when gas/water has a breakthrough in the outlet cell. Sand production profiles can be classified as transient production. This behavior correlated with field sand production as observed in the field NN. The sand production decreases with a decrease of oil fraction in produced liquid (see Figure 7.4). The $Water\_4.5MPa$ and $CO2\_4.5MPa$ cases have a lower sandrate, but sand production profiles can also be classified as transient.

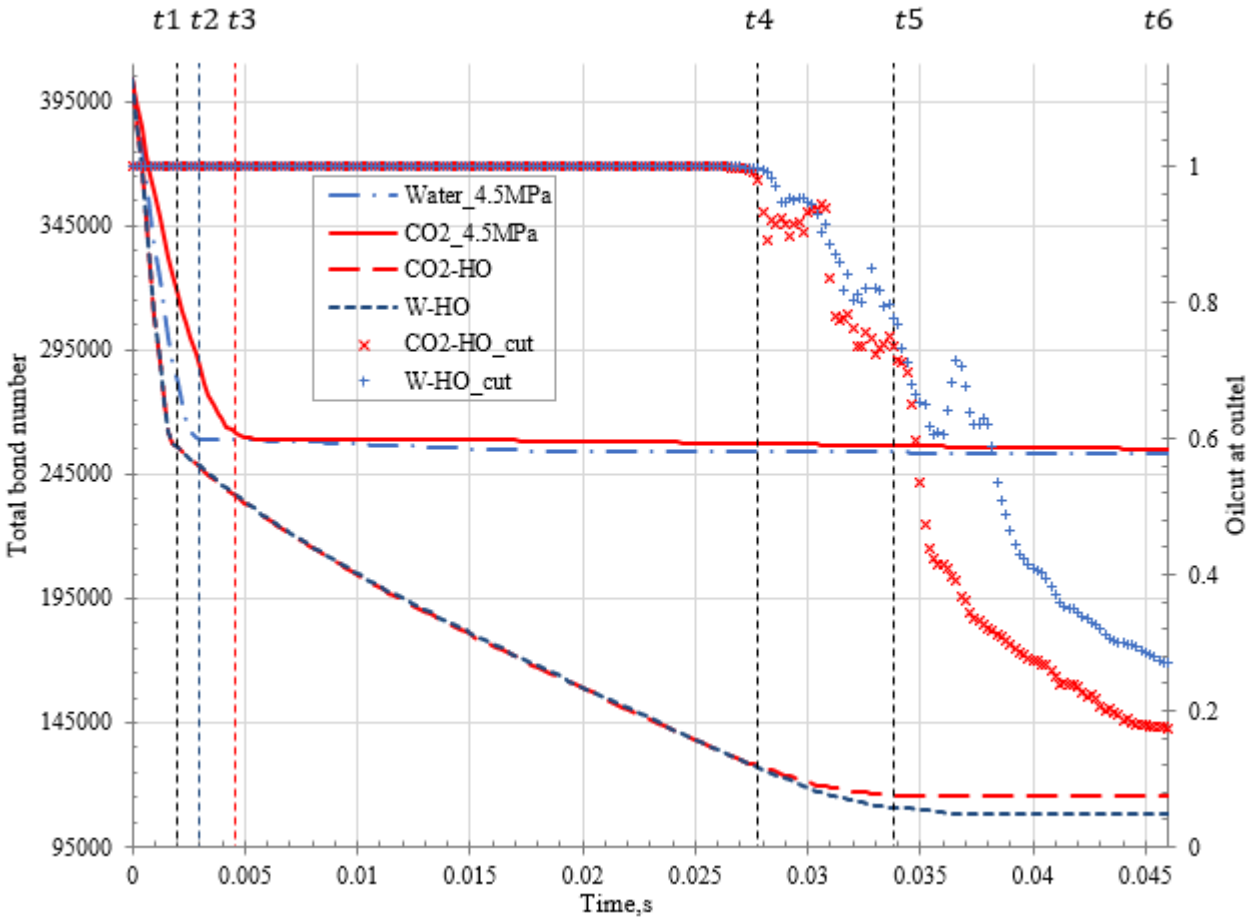


Figure 7.12. Bond number vs time. Overburden stress 5.9 MPa

The bond number behaviors are shown in Figure 7.12. The simulation cases have the same initial DEM conditions. The maximum value of a bond number is 403932 at overburden stress 5.9 MPa. For $Water\_4.5MPa$ and $CO2\_4.5MPa$ a similar pattern of bond breakage is observed. A sharp drop at the beginning and a constant value of bond number at the remaining time. The $t2$ and $t3$ are crucial times for these cases. Another trend of bond breakage behavior is found in the case $CO2\_HO$ and W_HO. There are four stages: 1) sharp drop at the beginning ($0 \div t1$); 2) constant bond breakage rate at the period between the $t1$ and $t4$; 3) damping/cessation of bond breakage ($t4\ to\ t5$) and no breakage stage ($t5\ to\ t6$). The time instant $t4$ corresponds to the first gas/water breakthrough in the outlet cell. The sample continuously breaks down and changes its behavior only after gas/water appearance in the outlet cell. The high bond breakage value of $CO2\_HO$ and W_HO cases promoted the high number of extracted particles during the sand production (Figure 7.11).

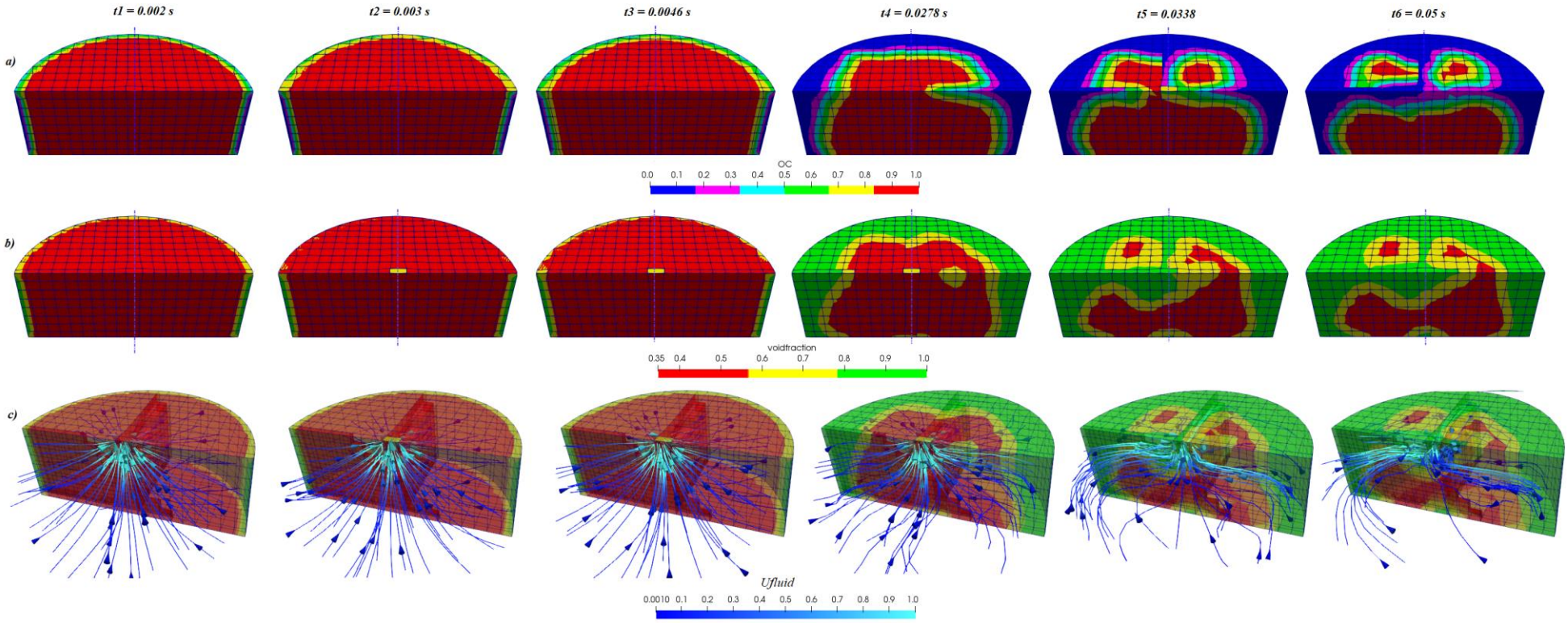


Figure 7.13. $CO2\_HO$ simulation results. a) Oilcut; b) Voidfraction; c) Fluid streamlines (over voidfraction background)

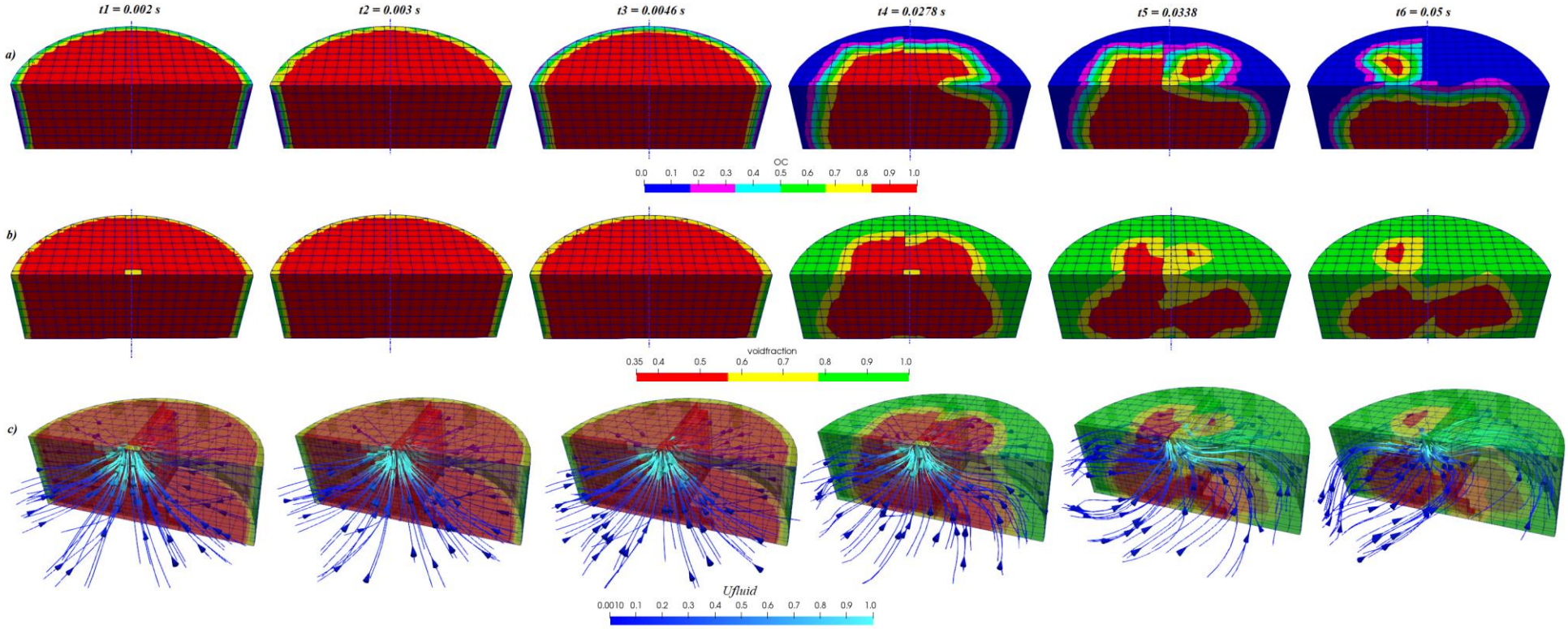


Figure 7.14. W_HO simulation results. a) Oilcut; b) Voidfraction; c) Fluid streamlines (over voidfraction background)

The $3D$ oilcut distribution of $CO2\_HO$ and W_HO cases at different modelling times are shown in Figure 7.13a - Figure 7.14a. Simulation cases try to reproduce heavy oil displacement profiles during the gas/water flooding. Dark blue zones indicate the injected fluid (gas, water), and the red zone is the heavy oil. The less viscous gas appeared in the outlet cell earlier than more viscous water (see $t4$ in Figs). The non-uniform oil displacement fronts are observed in both cases. This can be explained by the fingering effect. The dimensionless parameter M between heavy oil and injected fluids in this study correspondingly equal to 161.8 (W_HO) and 1100 ($CO2\_HO$). The high value of M creates a high probability of fingering effect (Doorwar & Mohanty, 2017; Kargozarfard et al., 2019; Lenormand et al., 1988; Moortgat, 2016).

The voidfraction distribution of $CO2\_HO$ and W_HO (Figure 7.13b and Figure 7.14b) before the first injection fluid breakthrough shows the stable conical shape of the remaining particles in the sample. Particles in this period were extracted with the same sandrate (see Figure 7.11). After the appearance of injected fluid in the outlet cell ($t \geq t4$), the voidfraction profiles are changed. The sand rate is decreased before and stopped after a time $t5$.

Figure 7.13c and Figure 7.14c show the fluid streamlines behavior in $CO2\_HO$ and W_HO cases. At initial timesteps, the flow is moving radially from inlet to outlet has a uniform distribution. After the time $t4$, the flow pattern is changed. The less viscous injected fluid due to the fingering effect creates the zone with higher permeability (channels) due to active washing out the sample particles. The flow velocities in these channels increase flow pattern is redistributed through the formed channels. The sand production and bond breakage in the reservoir with heavy oil did not stop until the channels formed. Another flow behaviour is observed in the case $CO2\_W$ with overburden stress is $3\ MPa$, flow distribution was not changed after gas breakthrough and channels were not created (see Figure 7.9).

Sand production simulations were carried out for about 5 days to complete $0.05\ s$. The injection velocity in these simulations with $5.9\ MPa$ overburden stress is sufficiently higher than all previous ones (see Table 4-1). The injection velocity is chosen in such a way that there is a breakthrough in the outlet cell.

A high production flowrate with sand particles used in the CFD-DEM model mimics the CHOPS technology, which is an economic success at Karazhanbas. CHOPS implies that intensive liquid production in producing wells creates high permeability channels (wormholes) for heavy oil flow. The created channels artificially expand the drainage zone and increase the oil recovery factor (Collins et al., 2008; Haddad & Gates, 2015; Rangriz Shokri & Babadagli, 2012).

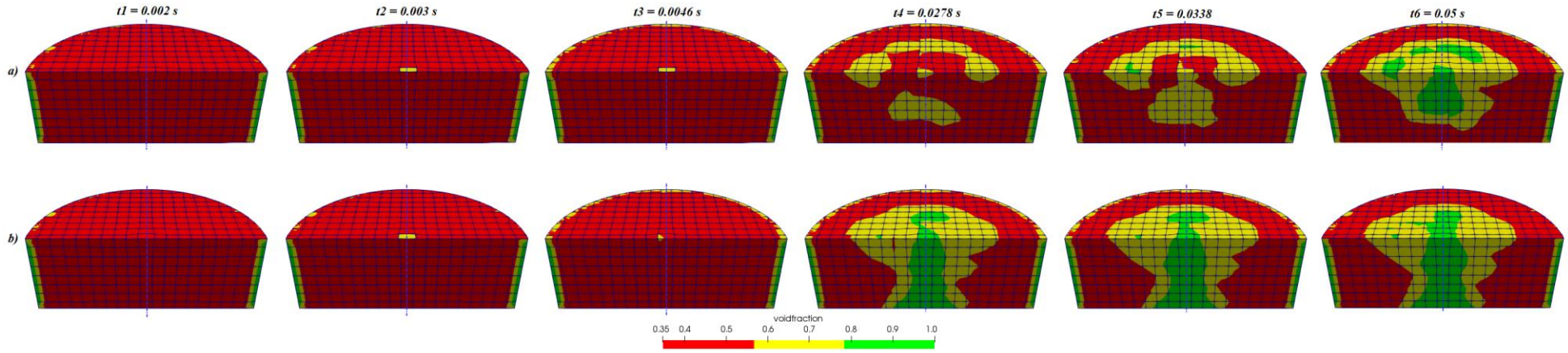


Figure 7.15. Voidfraction distribution for pure water and gas cases. a) $CO2\_4.5MPa$; b) $Water\_4.5MPa$

The voidfraction distribution of cases $CO2\_4.5MPa$ and $Water\_4.5MPa$ shown in Figure 7.15. The sand particles have been extracted only from the top, and the centre of the sample, which was initially perforated. The red zones the near inlet of the cylinder have a complete structure and do not change during simulation. Similar results can be found in Figure 7.8.

7.4 Conclusion

Sand production is a common problem associated with the production of hydrocarbons from weak sandstone reservoirs in the Kazakhstan oil-gas industry. It can dramatically affect the well performance, damage downhole and surface equipment. On the other hand, when sand is extracted from the reservoir, the porosity close to the well increases. An increased permeability zone is formed, which expands, providing a greater liquid flow into the wellbore.

The sanding mechanisms were investigated for different reservoir conditions and fluid properties using a coupled simulation with the two-phase fluid CFD-DEM model. There are two types of overburden stress $3\ MPa$ and $5.9\ MPa$ DEM simulations to mimic laboratory experiments and Karazhanbas reservoir conditions.

In the first case, the CFD-DEM numerical model was verified by experimental data. The two-phase gas-water flow breaks more bonded contacts and can produce more sand particles through an initially water-saturated sample than the one-phase water.

The reservoir properties of Object 2 of the Karazhanbas oil field were used for the second case. It was found that sand production with heavy oil provides a high bond breakage rate, and this behaviour is changed only after gas/water breakthrough. The bond breakage behaviour can be grouped into four different stages. The fingering effect and formation of new channels due to injection of less viscous fluid were observed.

The multiphase properties of fluid flow at the microscale affect the bond breakage and the distribution of fluid flow in the sample, which finally affects sand production. The transient sand production was observed in all cases in this Chapter.

CHAPTER 8 - CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

In this study, coupled CFD-DEM numerical models were used to investigate sand production phenomena. The new contact model for weak cement sandstone was successfully implemented in CFD-DEM models with one, two-phase fluid flow with particles and verified with laboratory experiments. The scientific visualization technics using cylindrical rings were also developed to provide greater insight into the complex sanding processes.

During all studies, the following operations: numerical sandstone preparation, perforation, and radial fluid flowing were repeated in a series of different simulation conditions.

In the first part of the research, the perforation process in the dry compressed sample was investigated. Overburden stress of compressed sample was $1\ MPa$. The perforation process created a damage zone of about $3r$ to $4r$ thickness of the unbonded particles with nearly complete bond breakage in this zone, where r is the radius of the perforation tunnel. A compacted lower porosity zone $2r$ to $4r$ (minimal porosity is 30.5%) was embedded inside the perforation damage zone. Under fluid flow conditions, it expanded quickly to attain a greater porosity than the intact porosity as the particles migrated towards the perforation tunnel during sand production. The simulation described a transient sand production phenomenon when small particles were produced first in a series of rapid events followed by larger particles in less frequent sand bursts. Particle movements inside the sample ceased once there was no longer sand production.

In the second part of the research, the sanding mechanisms were considered for different reservoir fluids: light (kinematic viscosity $1.62 \cdot 10^{-4}\ m^2/s$) and heavy ($4.12 \cdot 10^{-6}\ m^2/s$) oils. It was found that very different microstructural changes were inside the sandstone samples. More bond breakages for a light oil facilitated more significant particle movement and porosity change within the damage zone around the central perforation tunnel. Sand production occurred earlier for light oil in two stages of first transient sand production ($0.179\ mg$) and continuous sand production ($2.516\ mg$). In the final time step, sands were produced from two loose zones in an upper layer near the outlet and hence should be limited in scope. Only a continuous sand production ($2.726\ mg$) for heavy oil occurred later than for light oil. The heavy oil outflow mobilized a more significant proportion of the sample's particles and would lead to more severe sand production.

In the third part of the research, the two-phase gas-water flow at overburden stress $3MPa$ produced more sand particles through a water-saturated sample than the single-phase water flow. In the case of CFD-DEM simulations with Karazhanbas oil field properties (reservoir pressure $4.5\ MPa$ and overburden stress $5.9\ MPa$, it was found that sand production with heavy oil provides a high

bond breakage rate, and this behaviour is changed only after gas/water breakthrough. The fingering effect and formation of new channels due to injection of more mobile fluid were observed.

Limitation of the study:

- In all numerical simulations, the particles size distribution similar to natural sandstone from the Ustyurt-Buzachi Sedimentary Basin were used. The total number of particles in simulations equals to 10^5, while the reproduction of laboratory experiments (Kozhagulova et al., 2021) requires the use of $1.31 * 10^9$ particles.
- The numerical and experimental data verifications were conducted only qualitatively using dimensionless parameters. The quantitative analysis was impossible due to different scales.
- The sand material is considered as a conglomerate of ideal spherical particles, while in nature, sand particles have a more complex shape.
- The contact model for weak cement sandstone relies on of contact elasticity assumption and is not tested for a big range of overburden stresses. For the high pressure, the elastic model is less truthful. Makse et al. (2004) performed a test of the applicability of elasticity theory to granular materials and defined a dependency of applicability with pressure.
- The CFD-DEM model with the multiphase fluid flow (gas-oil, water-oil) at Karazhanbas oil field reservoir conditions is not verified by laboratory experiments, and all results should be considered preliminary.

In future work, as a continuation of this study are planned:

- Develop a new CFD-DEM multi-scale/multi-physics model with a PINN-based hybrid physics model with a sufficiently big number of particles ($10^6 \div 10^9$) to conduct a qualitative comparison of experimental and numerical results.
- Develop a new elastic-plastic contact model suitable for big-range overburden stresses.
- Develop the CFD-DEM model with compressible fluid to provide more insight into the natural flow physics and fully repeat laboratory sand production experiments.

BIBLIOGRAPHY

Abishev, A., Tokarev, V., & Sagyndikov, M. (2018). Evaluation of In-Situ Combustion Efficiency in Karazhanbas Oilfield, Western Kazakhstan. *Day 2 Thu, November 01, 2018*, 1–10. https://doi.org/10.2118/192553-MS

Acock, A., ORourk, A., Shirmboh, D., Alexander, J., Andersen, G., & López-de-Cárdenas, J. (2004). Practical Approaches to Sand Management. *Oil Field Review*, *16*, 10–27. http://www.slb.com/~/media/Files/resources/oilfield_review/ors04/spr04/02_sand_management.pdf

Agrawal, V., Shinde, Y., Shah, M. T., Utikar, R. P., Pareek, V. K., & Joshi, J. B. (2018). Effect of drag models on CFD–DEM predictions of bubbling fluidized beds with Geldart D particles. *Advanced Powder Technology*, *29*(11), 2658–2669. https://doi.org/10.1016/j.apt.2018.07.014

Ahmad, F. A., & Miskimins, J. L. (2019, January 29). Proppant Transport and Behavior in Horizontal Wellbores Using Low Viscosity Fluids. *Day 2 Wed, February 06, 2019*. https://doi.org/10.2118/194379-MS

Al-Awad, M. N. J., El-Sayed, A.-A. H., & Desouky, S. E.-D. M. (1999). Factors Affecting Sand Production from Unconsolidated Sandstone Saudi Oil and Gas Reservoir. *Journal of King Saud University - Engineering Sciences*, *11*(1), 151–172. https://doi.org/10.1016/S1018-3639(18)30995-4

Alsayed, M. I. (2002). Utilising the Hoek triaxial cell for multiaxial testing of hollow rock cylinders. *International Journal of Rock Mechanics and Mining Sciences*, *39*(3), 355–366. https://doi.org/10.1016/S1365-1609(02)00030-8

Alvarado, V., & Manrique, E. (2010). EOR's Current Status. In *Enhanced Oil Recovery* (pp. 133–156). Elsevier. https://doi.org/10.1016/B978-1-85617-855-6.00014-0

Anderson, T. B., & Jackson, R. (1967). Fluid Mechanical Description of Fluidized Beds. Equations of Motion. *Industrial & Engineering Chemistry Fundamentals*, *6*(4), 527–539. https://doi.org/10.1021/i160024a007

B. Dusseault, M., & El-Sayed, S. (1999). CHOP - Cold Heavy Oil Production. *IOR 1999 - 10th European Symposium on Improved Oil Recovery*. https://doi.org/10.3997/2214-4609.201406351

Batchelor, G. K. (1967). *An Introduction to Fluid Dynamics*. Cambridge University Press.

Bealessio, B. A., Blánquez Alonso, N. A., Mendes, N. J., Sande, A. V., & Hascakir, B. (2020). A review of enhanced oil recovery (EOR) methods applied in Kazakhstan. *Petroleum*, *November 2019*, 1–9. https://doi.org/10.1016/j.petlm.2020.03.003

Ben Mahmud, H., Leong, V. H., & Lestariono, Y. (2020). Sand production: A smart control framework for risk mitigation. *Petroleum*, *6*(1), 1–13. https://doi.org/10.1016/j.petlm.2019.04.002

Berger, R., Kloss, C., Kohlmeyer, A., & Pirker, S. (2015). Hybrid parallelization of the LIGGGHTS open-source DEM code. *Powder Technology*, *278*, 234–247. https://doi.org/10.1016/j.powtec.2015.03.019

Blunt, M. J., Bijeljic, B., Dong, H., Gharbi, O., Iglauer, S., Mostaghimi, P., Paluszny, A., & Pentland, C. (2013). Pore-scale imaging and modelling. *Advances in Water Resources*, *51*, 197–216. https://doi.org/10.1016/j.advwatres.2012.03.003

Bratli, R. K., & Risnes, R. (1981). Stability and Failure of Sand Arches. *Society of Petroleum Engineers Journal*, *21*(02), 236–248. https://doi.org/10.2118/8427-PA

Brignoli, M., Santarelli, F. J., & Papamichos, E. (1995). Capillary effects in sedimentary rocks: Application to reservoir water-flooding. *35th U.S. Symposium on Rock Mechanics, USRMS 1995*, 619–625.

Carman, P. C. (1939). Permeability of saturated sands, soils and clays. *The Journal of Agricultural Science*, *29*(2), 262–273. https://doi.org/10.1017/S0021859600051789

Cerasi, P., Bemtsen, A. N., Walle, L. E., & Papamichos, E. (2015). Sand production delay in gas flow experiments. *49th US Rock Mechanics / Geomechanics Symposium 2015*, *4*, 3078–3083.

Cerasi, Pierre, Papamichos, E., & Stenebraten, J. (2005). Quantitative Sand-Production Prediction: Friction-Dominated Flow Model. *Proceedings of SPE Latin American and Caribbean Petroleum Engineering Conference*, *2005-June*. https://doi.org/10.2523/94791-MS

Chen, G., Xiong, Q., Morris, P. J., Paterson, E. G., Sergeev, A., & Wang, Y.-C. (2014). OpenFOAM for Computational Fluid Dynamics. *Notices of the American Mathematical Society*, *61*(4), 354. https://doi.org/10.1090/noti1095

Chen, X., & Wang, J. (2014). A comparison of two-fluid model, dense discrete particle model and CFD-DEM method for modeling impinging gas–solid flows. *Powder Technology*, *254*, 94–102. https://doi.org/10.1016/j.powtec.2013.12.056

Chen, Z., Xie, C., Chen, Y., & Wang, M. (2016). Bonding Strength Effects in Hydro-Mechanical Coupling Transport in Granular Porous Media by Pore-Scale Modeling. *Computation*, *4*(1), 15. https://doi.org/10.3390/computation4010015

Cheng, J., Dou, Y., Zhang, N., Li, Z., & Wang, Z. (2018). A New Method for Predicting Erosion Damage of Suddenly Contracted Pipe Impacted by Particle Cluster via CFD-DEM. *Materials*, *11*(10), 1858. https://doi.org/10.3390/ma11101858

Cheung, L. Y. G. (2010). *Micromechanics of Sand Production in Oil Wells*. Imperial College of London. https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.528300

Climent, N. (2016). *A Coupled CFD-DEM Model for Sand Production in Oil Wells*. *September*. http://hdl.handle.net/2117/105810

Climent, N., Arroyo, M., O'Sullivan, C., & Gens, A. (2014). Sand production simulation coupling DEM with CFD. *European Journal of Environmental and Civil Engineering*, *18*(9), 983–1008. https://doi.org/10.1080/19648189.2014.920280

Coetzee, C. J. (2017). Review: Calibration of the discrete element method. *Powder Technology*, *310*, 104–142. https://doi.org/10.1016/j.powtec.2017.01.015

Coker, A. K. (2007). MECHANICAL SEPARATIONS. In *Ludwig's Applied Process Design for Chemical and Petrochemical Plants* (pp. 371–443). Elsevier. https://doi.org/10.1016/B978-075067766-0/50013-0

Collins, P. M., Dusseault, M. B., Dorscher, D., & Kueber, E. (2008). Implementing CHOPS in the Karazhanbas Heavy Oil Field, Kazakhstan. *World Heavy Oil Congress*. http://www.petroleumgeomechanics.com/papers/Collins et al. WHOC 2008-500.pdf

Courant, R., Friedrichs, K., & Lewy, H. (1986). Über die partiellen Differenzengleichungen der mathematischen Physik. In *Kurt Otto Friedrichs* (pp. 53–95). Birkhäuser Boston. https://doi.org/10.1007/978-1-4612-5385-3_7

Crank, J., & Nicolson, P. (1947). A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Mathematical Proceedings of the Cambridge Philosophical Society*, *43*(1), 50–67. https://doi.org/10.1017/S0305004100023197

Cromer, A. (1981). Stable solutions using the Euler approximation. *American Journal of Physics*, *49*(5), 455–459. https://doi.org/10.1119/1.12478

Crowe, C. T., Schwarzkopf, J. D., Sommerfeld, M., & Tsuji, Y. (2011). *Multiphase Flows with Droplets and Particles* (2nd Editio). CRC Press. https://doi.org/10.1201/b11103

Cundall, P. A., & Strack, O. D. L. (1979). A discrete numerical model for granular assemblies. *Géotechnique*, *29*(1), 47–65. https://doi.org/10.1680/geot.1979.29.1.47

Daigle, H., Rasromani, E., & Gray, K. E. (2017). Near-wellbore permeability alteration in depleted, anisotropic reservoirs. *Journal of Petroleum Science and Engineering*, *157*(July), 302–311. https://doi.org/10.1016/j.petrol.2017.07.046

Damián., S. M. (2013). *An Extended Mixture Model for the Simultaneous Treatment of Short and Long Scale Interfaces* [Universidad Nacional Del Litoral]. https://bibliotecavirtual.unl.edu.ar:8443/handle/11185/489

Damián, Santiago Márquez. (2009). Description and utilization of interFoam multiphase solver. *Report*, 1–64. http://infofich.unl.edu.ar/upload/3be0e16065026527477b4b948c4caa7523c8ea52.pdf

Dauyeshova, B., Rojas-Solórzano, L. R., & Monaco, E. (2018). Numerical simulation of diffusion process in T-shaped micromixer using Shan-Chen Lattice Boltzmann Method. *Computers & Fluids*, *167*, 229–240. https://doi.org/10.1016/j.compfluid.2018.03.029

DCS Computing GmbH. (n.d.). *CFDEM®coupling Documentation*. https://www.cfdem.com/media/CFDEM/docu/CFDEMcoupling_Manual.html#

Derjaguin, B. ., Muller, V. ., & Toporov, Y. . (1975). Effect of contact deformations on the adhesion of particles. *Journal of Colloid and Interface Science*, *53*(2), 314–326. https://doi.org/10.1016/0021-9797(75)90018-1

Derksen, J., Eskin, D., Luo, L.-S., & Krafczyk, M. (2013). Mesoscopic methods in engineering and science. *Computers & Mathematics with Applications*, *65*(2), 127–128. https://doi.org/10.1016/j.camwa.2012.12.010

Deshpande, S. S., Anumolu, L., & Trujillo, M. F. (2012). Evaluating the performance of the two-phase flow solver interFoam. *Computational Science & Discovery*, *5*(1), 014016. https://doi.org/10.1088/1749-4699/5/1/014016

Di Felice, R. (1994). The voidage function for fluid-particle interaction systems. *International Journal of Multiphase Flow*, *20*(1), 153–159. https://doi.org/10.1016/0301-9322(94)90011-6

Ding, J., & Gidaspow, D. (1990). A bubbling fluidization model using kinetic theory of granular flow. *AIChE Journal*, *36*(4), 523–538. https://doi.org/10.1002/aic.690360404

Doorwar, S., & Mohanty, K. K. (2017). Viscous-Fingering Function for Unstable Immiscible Flows. *SPE Journal*, *22*(01), 019–031. https://doi.org/10.2118/173290-PA

ENGYS Ltd. (n.d.). *HELYX-OS.Open-source Graphical User Interface developed by ENGYS to operate and control natively the standard CFD utilities and solvers provided with the OpenFOAM libraries*. http://engys.github.io/HELYX-OS/

Enwald, H., Peirano, E., & Almstedt, A.-E. (1996). Eulerian two-phase flow theory applied to fluidization. *International Journal of Multiphase Flow*, *22*(SUPPL. 1), 21–66. https://doi.org/10.1016/S0301-9322(96)90004-X

Ergun, S. (1952). Fluid flow through packed columns. In *Chemical enfineering progress* (pp. 89–94).

Fanchi, J. R. (2002). Improved Recovery. In *Shared Earth Modeling* (pp. 272–281). Elsevier. https://doi.org/10.1016/B978-075067522-2/50016-1

Fromm, J. E. (1968). A method for reducing dispersion in convective difference schemes. *Journal of Computational Physics*, *3*(2), 176–189. https://doi.org/10.1016/0021-9991(68)90015-6

Geilikman, M. B., & Dusseault, M. B. (1997). Fluid rate enhancement from massive sand production in heavy-oil reservoirs. *Journal of Petroleum Science and Engineering*, *17*(1–2), 5–18. https://doi.org/10.1016/S0920-4105(96)00052-6

Geldart, D. (1973). Elsevier Sequoia SA, Lausanne-Printed in the Netheriands Types of Gas Fhidization. *Powder Technology*, *7*, 285–292.

Gidaspow, D., Bezburuah, R., & Ding, J. (1992). Hydrodynamics of circulating fluidized beds: Kinetic theory approach. *7th Fluidization Conference*, 75–82. http://www.osti.gov/scitech/servlets/purl/5896246

Goniva, C., Blais, B., Radl, S., & Kloss, C. (2015). Open source cfd-dem modelling for particle-based processes. *11th International Conference on CFD in the Minerals and Process Industries, December*, 1–7. https://www.cfd.com.au/cfd_conf15/PDFs/140GON.pdf

Goniva, Christoph, Kloss, C., Deen, N. G., Kuipers, J. A. M., & Pirker, S. (2012). Influence of rolling

friction on single spout fluidized bed simulation. *Particuology*, *10*(5), 582–591. https://doi.org/10.1016/j.partic.2012.05.002

Goniva, Christoph, Kloss, C., Hager, A., & Pirker, S. (2010). *An Open Source CFD-DEM Perspective*. *February 2015*.

Goshtasbi, K., Elyasi, A., & Naeimipour, A. (2013). Numerical assessment of the mechanical stability in vertical, directional and horizontal wellbores. *International Journal of Mining Science and Technology*, *23*(6), 937–942. https://doi.org/10.1016/j.ijmst.2013.11.010

Gueyffier, D., Li, J., Nadim, A., Scardovelli, R., & Zaleski, S. (1999). Volume-of-Fluid Interface Tracking with Smoothed Surface Stress Methods for Three-Dimensional Flows. *Journal of Computational Physics*, *152*(2), 423–456. https://doi.org/10.1006/jcph.1998.6168

Guo, Y., Wu, C. Y., & Thornton, C. (2013). Modeling gas-particle two-phase flows with complex and moving boundaries using DEM-CFD with an immersed boundary method. *AIChE Journal*. https://doi.org/10.1002/aic.13900

Haddad, A. S., & Gates, I. (2015). Modelling of Cold Heavy Oil Production with Sand (CHOPS) using a fluidized sand algorithm. *Fuel*, *158*, 937–947. https://doi.org/10.1016/j.fuel.2015.06.032

Hall, C. D., & Harrisberger, W. H. (1970). Stability of Sand Arches: A Key to Sand Control. *Journal of Petroleum Technology*, *22*(07), 821–829. https://doi.org/10.2118/2399-PA

Han, G., Dusseault, M. B., & Cook, J. (2002). Quantifying Rock Capillary Strength Behavior in Unconsolidated Sandstones. *Proceedings of the SPE/ISRM Rock Mechanics in Petroleum Engineering Conference*, 171–180.

Han, Y., & Cundall, P. (2017). Verification of two-dimensional LBM-DEM coupling approach and its application in modeling episodic sand production in borehole. *Petroleum*, *3*(2), 179–189. https://doi.org/10.1016/j.petlm.2016.07.001

Hanley, K. J., O'Sullivan, C., Oliveira, J. C., Cronin, K., & Byrne, E. P. (2011). Application of Taguchi methods to DEM calibration of bonded agglomerates. *Powder Technology*, *210*(3), 230–240. https://doi.org/10.1016/j.powtec.2011.03.023

Hardy, J., Pomeau, Y., & de Pazzis, O. (1973). Time evolution of a two-dimensional model system. I. Invariant states and time correlation functions. *Journal of Mathematical Physics*, *14*(12), 1746–1759. https://doi.org/10.1063/1.1666248

Harris, P. C., Morgan, R. G., & Heath, S. J. (2005). Measurement of Proppant Transport of Frac Fluids. *All Days*, 281–292. https://doi.org/10.2118/95287-MS

Hertz, H. (1881). Über die Berührung fester elastischer Körper. *Journal Für Die Reine Und Angewandte Mathematik*, *171*, 156–171.

Hirt, C. ., & Nichols, B. . (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, *39*(1), 201–225. https://doi.org/10.1016/0021-9991(81)90145-5

Huang, H., Wang, L., & Lu, X. Y. (2011). Evaluation of three lattice Boltzmann models for multiphase flows in porous media. *Computers and Mathematics with Applications*. https://doi.org/10.1016/j.camwa.2010.06.034

Hughes, K., Santos, N., Arias, R. E., & Nadezhdin, S. V. (2008). New viscoelastic surfactant fracturing fluids now compatible with CO 2 drasticaly improve gas production in rockies. *Proceedings - SPE International Symposium on Formation Damage Control*, *1*, 17–21. https://doi.org/10.2118/111431-ms

Irmay, S. (1958). On the theoretical derivation of Darcy and Forchheimer formulas. *Eos, Transactions American Geophysical Union*, *39*(4), 702–707. https://doi.org/10.1029/TR039i004p00702

Isehunwa, S., & Farotade, A. (2010). Sand Failure Mechanism and Sanding Parameters in Niger Delta Oil Reservoirs. *International Journal of Engineering Science and Technology*, *2*(5), 777–782. https://www.researchgate.net/publication/50273851_Sand_Failure_Mechanism_and_Sanding_Parameters_in_Niger_Delta_Oil_Reservoirs/fulltext/0e5f78d8f0c4c08779007455/Sand-

Failure-Mechanism-and-Sanding-Parameters-in-Niger-Delta-Oil-Reservoirs.pdf
Issa, R. . (1986). Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, *62*(1), 40–65. https://doi.org/10.1016/0021-9991(86)90099-9
Jing, L., Kwok, C. Y., Leung, Y. F., & Sobral, Y. D. (2016). Extended CFD-DEM for free-surface flow with multi-size granules. *International Journal for Numerical and Analytical Methods in Geomechanics*, *40*(1), 62–79. https://doi.org/10.1002/nag.2387
K.L. Johnson. (1976). Adhesion at the contact of solids. *W. Koiter (Ed.), Proc. XIV Int. Congress Theoret. Appl. Mech., North Holland, New York*, 133–143.
Kafui, D. K., Johnson, S., Thornton, C., & Seville, J. P. K. (2011). Parallelization of a Lagrangian–Eulerian DEM/CFD code for application to fluidized beds. *Powder Technology*, *207*(1–3), 270–278. https://doi.org/10.1016/j.powtec.2010.11.008
Kafui, K. D., Thornton, C., & Adams, M. J. (2002). Discrete particle-continuum fluid modelling of gas–solid fluidised beds. *Chemical Engineering Science*, *57*(13), 2395–2410. https://doi.org/10.1016/S0009-2509(02)00140-9
Kallemov, B., Bhalla, A., Griffith, B., & Donev, A. (2016). An immersed boundary method for rigid bodies. *Communications in Applied Mathematics and Computational Science*, *11*(1), 79–141. https://doi.org/10.2140/camcos.2016.11.79
Kamrin, K. (2010). Nonlinear elasto-plastic model for dense granular flow. *International Journal of Plasticity*, *26*(2), 167–188. https://doi.org/10.1016/j.ijplas.2009.06.007
Kargozarfard, Z., Riazi, M., & Ayatollahi, S. (2019). Viscous fingering and its effect on areal sweep efficiency during waterflooding: an experimental study. *Petroleum Science*, *16*(1), 105–116. https://doi.org/10.1007/s12182-018-0258-6
Kazakhstan Upstream oil and gas technology and R&d Roadmap. (2013). *Institute for Manufacturing, University of Cambridge*. https://www.ifm.eng.cam.ac.uk/uploads/Roadmapping/Kaz_RM_book_English.pdf
Kenneth Langstreth Johnson, K. K. and A. D. R. (1971). Surface energy and the contact of elastic solids. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, *324*(1558), 301–313. https://doi.org/10.1098/rspa.1971.0141
Khamitov, F., Minh, N. H., Zhao, Y., & Fok, S. C. (2019). *Implementation of jkr contact model in liggghts*. 9. https://mercurylab.co.uk/dem8/wp-content/uploads/sites/4/2019/07/103.pdf
Kim, A. S. (2012). Sand Control Mechanism and its Impact on Mature Fields - Halliburton. In *Halliburtonblog.Gom* (p. 3). http://halliburtonblog.com/sand-control-mechanism-and-its-impact-on-mature-fields/
Kloss, C., & Goniva, C. (2011). LIGGGHTS - Open Source Discrete Element Simulations of Granular Materials Based on Lammps. In *Supplemental Proceedings* (pp. 781–788). John Wiley & Sons, Inc. https://doi.org/10.1002/9781118062142.ch94
Kloss, C., Goniva, C., Hager, A., Amberger, S., & Pirker, S. (2012). Models, algorithms and validation for opensource DEM and CFD-DEM. *Progress in Computational Fluid Dynamics, An International Journal*, *12*(2/3), 140. https://doi.org/10.1504/PCFD.2012.047457
Koch, D. L., & Hill, R. J. (2001). Inertial effects in suspension and porous—media flow. *Annual Review of Fluid Mechanics*, *33*(1), 619–647. https://doi.org/10.1146/annurev.fluid.33.1.619
Kozeny, J. (1927). Uber kapillare Leitung der Wasser in Boden. *Sitzungsber. Akad. Wiss. Wien*, 271–306.
Kozhagulova, A., Minh, N. H., Zhao, Y., & Fok, S. C. (2020a). A study on bond breakage behavior of weak Cretaceous Kazakhstani reservoir sandstone analogue. *Geomechanics for Energy and the Environment*, *21*, 100159. https://doi.org/10.1016/j.gete.2019.100159
Kozhagulova, A., Minh, N. H., Zhao, Y., & Fok, S. C. (2020b). Experimental and Analytical Investigation of Sand Production in Weak formations for Multiple Well Shut-Ins. *Journal of*

Petroleum Science and Engineering, *195*(August 2019), 107628. https://doi.org/10.1016/j.petrol.2020.107628

Kozhagulova, A., Shabdirova, A., Minh, N. H., & Zhao, Y. (2021). An integrated laboratory experiment of realistic diagenesis, perforation and sand production using a large artificial sandstone specimen. *Journal of Rock Mechanics and Geotechnical Engineering*, *13*(1), 154–166. https://doi.org/10.1016/j.jrmge.2020.09.004

Kozicki, J., & Donzé, F. V. (2009). YADE-OPEN DEM: an open-source software using a discrete element method to simulate granular material. *Engineering Computations*, *26*(7), 786–805. https://doi.org/10.1108/02644400910985170

Labuz, J. F., & Zang, A. (2012). Mohr–Coulomb Failure Criterion. *Rock Mechanics and Rock Engineering*, *45*(6), 975–979. https://doi.org/10.1007/s00603-012-0281-7

Laney, C. B. (1998). The CFL Condition. In *Computational Gasdynamics* (pp. 214–221). Cambridge University Press. https://doi.org/10.1017/CBO9780511605604.016

Lenormand, R., Touboul, E., & Zarcone, C. (1988). Numerical models and experiments on immiscible displacements in porous media. *Journal of Fluid Mechanics*, *189*(1988), 165–187. https://doi.org/10.1017/S0022112088000953

Leonard, B. P. (1979). A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, *19*(1), 59–98. https://doi.org/10.1016/0045-7825(79)90034-3

Li, Yanjie, Xu, Y., & Thornton, C. (2005). A comparison of discrete element simulations and experiments for "sandpiles" composed of spherical particles. *Powder Technology*, *160*(3), 219–228. https://doi.org/10.1016/j.powtec.2005.09.002

Li, Yong, Zhang, J., & Fan, L.-S. (1999). Numerical simulation of gas–liquid–solid fluidization systems using a combined CFD-VOF-DPM method: bubble wake behavior. *Chemical Engineering Science*, *54*(21), 5101–5107. https://doi.org/10.1016/S0009-2509(99)00263-8

Lian, G., Thornton, C., & Icdfui, D. (1998). *Trubal A 3-D Computer Program for Modelling Particle Assemblies*.

Liu, Y., Ersson, M., Liu, H., Jönsson, P., & Gan, Y. (2019). Comparison of Euler-Euler Approach and Euler–Lagrange Approach to Model Gas Injection in a Ladle. *Steel Research International*, *90*(5), 1–13. https://doi.org/10.1002/srin.201800494

Lohner, R., Corrigan, A. T., Wichmann, K.-R., & Wall, W. (2014). Comparison of Lattice-Boltzmann and Finite Difference Solvers. *52nd Aerospace Sciences Meeting*, *January*, 1–13. https://doi.org/10.2514/6.2014-1439

Makse, H. A., Gland, N., Johnson, D. L., & Schwartz, L. (2004). Granular packings: Nonlinear elasticity, sound propagation, and collective relaxation dynamics. *Physical Review E*, *70*(6), 061302. https://doi.org/10.1103/PhysRevE.70.061302

Marchelli, F., Hou, Q., Bosio, B., Arato, E., & Yu, A. (2020). Comparison of different drag models in CFD-DEM simulations of spouted beds. *Powder Technology*, *360*, 1253–1270. https://doi.org/10.1016/j.powtec.2019.10.058

Margenau, H. (1939). Van der waals forces. *Reviews of Modern Physics*, *11*(1), 1–35. https://doi.org/10.1103/RevModPhys.11.1

Menon, Sankar; Nilsson, H. (2016). *CFD with OpenSource software Coupled Level-Set with VOF interFoam*. 22. http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2015/SankarMenon/Report_SankarMenon.pdf

Mesarovic, S. D. J., & Fleck, N. A. (1999). Spherical indentation of elastic–plastic solids. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, *455*(1987), 2707–2728. https://doi.org/10.1098/rspa.1999.0423

Mindlin, R. D., & Deresiewicz, H. (1954). Thickness-Shear and Flexural Vibrations of a Circular

Disk. *Journal of Applied Physics*, *25*(10), 1329–1332. https://doi.org/10.1063/1.1721554
Moortgat, J. (2016). Viscous and gravitational fingering in multiphase compositional and compressible flow. *Advances in Water Resources*, *89*, 53–66. https://doi.org/10.1016/j.advwatres.2016.01.002
Moukalled, F., Mangani, L., & Darwish, M. (2015). *The Finite Volume Method in Computational Fluid Dynamics* (p. 817). Sprinter. https://link.springer.com/content/pdf/10.1007/978-3-319-16874-6.pdf
Müller, P., & Tomas, J. (2012). Compression Behavior of Moist Spherical Zeolite 4A Granules. *Chemical Engineering & Technology*, *35*(9), 1677–1684. https://doi.org/10.1002/ceat.201200245
Narsilio, G. A., Buzzi, O., Fityus, S., Yun, T. S., & Smith, D. W. (2009). Upscaling of Navier–Stokes equations in porous media: Theoretical, numerical and experimental approach. *Computers and Geotechnics*, *36*(7), 1200–1206. https://doi.org/10.1016/j.compgeo.2009.05.006
Nguyen, V.-T., & Park, W.-G. (2017). A volume-of-fluid (VOF) interface-sharpening method for two-phase incompressible flows. *Computers & Fluids*, *152*, 104–119. https://doi.org/10.1016/j.compfluid.2017.04.018
Nichols, B. D., & Hirt, C. W. (1975). *Methods for calculating multidimensional, transient free surface flows past bodies*. https://ui.adsabs.harvard.edu/abs/1975STIN...7625526N/abstract
Nie, Z., Zheng, X., & Xie, C. (2010, April 4). New Cementing Technologies Successfully Solved the Problems in Shallow Gas, Low Temperature and Easy Leakage Formations of North Buzachi Oilfield. *International Oil and Gas Conference and Exhibition in China*. https://doi.org/10.2118/131810-MS
Noh, W. P., & Woodward, P. (1976). *SLIC (simple line interface calculation)*. 1–1. https://doi.org/10.1002/9783527678679.dg12095
Norouzi, H. R., Zarghami, R., Sotudeh-Gharebagh, R., & Mostoufi, N. (2016). *Coupled CFD-DEM Modeling*. John Wiley & Sons, Ltd. https://doi.org/10.1002/9781119005315
Nouri, A., Vaziri, H., & Kuru, E. (2006). Numerical investigation of sand production under realistic reservoir/well flow conditions. *Canadian International Petroleum Conference 2006, CIPC 2006*. https://doi.org/10.2118/2006-093
O'Connor, R. M., Torczynski, J. R., Preece, D. S., Klosek, J. T., & Williams, J. R. (1997). Discrete element modeling of sand production. *International Journal of Rock Mechanics and Mining Sciences*, *34*(3–4), 231.e1-231.e15. https://doi.org/10.1016/S1365-1609(97)00198-6
O'Sullivan, C. (2011). *Particulate Discrete Element Modelling: A Geomechanics Perspective* (1st ed.). Spon Press/Taylor & Francis. https://link.springer.com/article/10.1007/s00024-013-0736-0
OpenCFD. (2021). *OpenFoam v2106. Programmer's Guide* (Issue June). https://www.openfoam.com/documentation/overview
Oryem, M., Arroyo, M., Butlanska, J., & Gens, A. (2016). DEM modelling of cone penetration tests in a double-porosity crushable granular material. *COMPUTERS AND GEOTECHNICS*, *73*, 109–127. https://doi.org/10.1016/j.compgeo.2015.12.001
Osher, S., & Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, *79*(1), 12–49. https://doi.org/10.1016/0021-9991(88)90002-2
Papamichos, E., Vardoulakis, I., Tronvoll, J., & Skjaerstein, A. (2001). Volumetric sand production model and experiment. *International Journal for Numerical and Analytical Methods in Geomechanics*, *25*(8), 789–808. https://doi.org/10.1002/nag.154
Papamichos, E, Cerasi, P., & Stenebråten, J. (2010). Sand production rate under multiphase flow and water breakthrough. *44th US Rock Mechanics Symposium*. https://onepetro.org/ARMAUSRMS/proceedings-abstract/ARMA10/All-ARMA10/ARMA-

10-340/118011
Papamichos, Euripides. (2006). Sand production. *Revue Européenne de Génie Civil*, *10*(6–7), 803–816. https://doi.org/10.1080/17747120.2006.9692856
Parmigiani, A., Huber, C., Bachmann, O., & Chopard, B. (2011). Pore-scale mass and reactant transport in multiphase porous media flows. *Journal of Fluid Mechanics*, *686*, 40–76. https://doi.org/10.1017/jfm.2011.268
Patankar, N. A., & Joseph, D. D. (2001). Modeling and numerical simulation of particulate flows by the Eulerian–Lagrangian approach. *International Journal of Multiphase Flow*, *27*(10), 1659–1684. https://doi.org/10.1016/S0301-9322(01)00021-0
Patankar, S. (1980). Numerical heat transfer and fluid flow: Computational methods in mechanics and thermal science. In *Hemisphere Publication Corporation,Washington, DC* (pp. 1–197). http://www.crcpress.com/product/isbn/9780891165224
Peng, Z., Doroodchi, E., & Moghtaderi, B. (2020). Heat transfer modelling in Discrete Element Method (DEM)-based simulations of thermal processes: Theory and model development. *Progress in Energy and Combustion Science*, *79*, 100847. https://doi.org/10.1016/j.pecs.2020.100847
Peskin, C. S. (2002). The immersed boundary method. *Acta Numerica*, *11*, 479–517. https://doi.org/10.1017/S0962492902000077
Polillo, A., & Graves, R. M. (1994). *Simulation of Sand Arching Mechanics Using an Elasto-Plastic Finite Element Formulation This paper presents a method to simulate sand production mechanics using an George D . Vassilellis , SPE , Richardson Operating Company James W . Crafton , SPE © 1994*. https://doi.org/https://doi.org/10.2118/23728-PA
Potyondy, D. O., & Cundall, P. A. (2004). A bonded-particle model for rock. *International Journal of Rock Mechanics and Mining Sciences*, *41*(8 SPEC.ISS.), 1329–1364. https://doi.org/10.1016/j.ijrmms.2004.09.011
Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). *NUMERICAL RECIPES,Third Edition*. Cambridge University Press.
Rahmati, H., Jafarpour, M., Azadbakht, S., Nouri, A., Vaziri, H., Chan, D., & Xiao, Y. (2013). Review of Sand Production Prediction Models. *Journal of Petroleum Engineering*, *2013*, 1–16. https://doi.org/10.1155/2013/864981
Rakhimzhanova, A., Thornton, C., Amanbek, Y., & Zhao, Y. (2019). *Numerical simulations of cone penetration tests in cemented sandstone*. 1–15. https://doi.org/https://doi.org/10.31224/osf.io/uhy36
Rakhimzhanova, A K, Khamitov, F. A., Minh, N. H., & Thornton, C. (2018). 3D DEM simulations of triaxial compression tests of cemented sandstone. In *Proceedings of IS Atlanta 2018 symposium on geomechanics from micro to macro in research and practice*.
Rakhimzhanova, Aigerim K., Thornton, C., Minh, N. H., Fok, S. C., & Zhao, Y. (2019). Numerical simulations of triaxial compression tests of cemented sandstone. *Computers and Geotechnics*, *113*(May), 103068. https://doi.org/10.1016/j.compgeo.2019.04.013
Rakhimzhanova, Aigerim K, Thornton, C., Amanbek, Y., & Zhao, Y. (2020). *Numerical simulations of sand production in oil wells using the CFD-DEM-IBM approach*. 1–24. https://doi.org/https://doi.org/10.31224/osf.io/pkteu
Rangriz Shokri, A., & Babadagli, T. (2012). An approach to model CHOPS (cold heavy oil production with sand) and post-CHOPS applications. *Proceedings - SPE Annual Technical Conference and Exhibition*, *3*(1), 2305–2316. https://doi.org/10.2118/159437-ms
Ranjith, P. G., Perera, M. S. A., Perera, W. K. G., Wu, B., & Choi, S. K. (2013). Effective parameters for sand production in unconsolidated formations: An experimental study. *Journal of Petroleum Science and Engineering*, *105*, 34–42. https://doi.org/10.1016/j.petrol.2013.03.023
Rathbone, D., Marigo, M., Dini, D., & van Wachem, B. (2015). An accurate force-displacement law

for the modelling of elastic-plastic contacts in discrete element simulations. *Powder Technology*, *282*, 2–9. https://doi.org/10.1016/j.powtec.2014.12.055

Ray, P., Rijken, M., Cameron, J., Jones, C., Energy, C., & Company, T. (2014). *Estimating Sand Production Volume in Oil and Gas Reservoir: Vol. i* (Issue October, pp. 27–29).

Ringwood, J. V. (2015). Implementation of an OpenFOAM Numerical Wave Tank for Wave Energy Experiments. *Proceedings of the 11th European Wave and Tidal Energy Conference, September*, 1–10.

Risnes, R., Bratli, R. K., & Horsrud, P. (1982a). Sand arching - A case study. *SPE*. https://doi.org/https://doi.org/10.2118/12948-MS

Risnes, R., Bratli, R. K., & Horsrud, P. (1982b). Sand Stresses Around a Wellbore. *Society of Petroleum Engineers Journal*, *22*(06), 883–898. https://doi.org/10.2118/9650-PA

Rong, L. W., Dong, K. J., & Yu, A. B. (2013). Lattice-Boltzmann simulation of fluid flow through packed beds of uniform spheres: Effect of porosity. *Chemical Engineering Science*, *99*, 44–58. https://doi.org/10.1016/j.ces.2013.05.036

Saad, Y. (2003). Iterative Methods for Sparse Linear Systems, Second Edition. In *Methods*. Society for Industrial and Applied Mathematics. http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&amp

Seil, P. (2016). *LBDEMcoupling: Implementation, Validation, and Applications of a Coupled Open-Source Solver for Fluid-Particle Systems* [Johannes Kepler University]. http://epub.jku.at/obvulihs/download/pdf/1484085?originalFilename=true

Shabdirova, A. D., Bissekenova, Z., Minh, N. H., & Kim, J. R. (2016). Sample preparation method of clay-rich sandstone analogue of sandstone reservoirs in Kazakhstan. *50th US Rock Mechanics / Geomechanics Symposium 2016*, *2*, 904–910.

Shabdirova, A. D., Khamitov, F., Kozhagulova, A. A., Amanbek, Y., Minh, N. H., & Zhao, Y. (2020). Experimental and Numerical Investigation of the Plastic Zone Permeability. *American Rock Mechanics Association, 54th U.S. Rock Mechanics/Geomechanics Symposium, 28 June - 1 July*. https://www.onepetro.org/conference-paper/ARMA-2020-1626

Shabdirova, A., Minh, N. H., & Zhao, Y. (2019). A sand production prediction model for weak sandstone reservoir in Kazakhstan. *Journal of Rock Mechanics and Geotechnical Engineering*, *11*(4), 760–769. https://doi.org/10.1016/j.jrmge.2018.12.015

Shabdirova, A., Minh, N. H., Zhao, Y., & Fok, S. C. (2020). The Role of Plastic Zone Permeability in Sand Production in Weak Sandstone Reservoirs. *Underground Space. Under Review*.

Shabdirova Ainash, Khamitov Furkhat, Kozhagulova Ashirgul, & Amanbek Yerlan. (2021). SP_2_phase. *Under Review*.

Song, C., Wang, P., Makse, H. A., Goddard, J., Giovine, P., & Jenkins, J. T. (2010). Theory of random packings. *AIP Conference Proceedings*, *1227*(2), 271–279. https://doi.org/10.1063/1.3435397

Spitzenberger, A., Neumann, S., Heinrich, M., & Schwarze, R. (2020). Particle detection in VOF-simulations with OpenFOAM. *SoftwareX*, *11*, 100382. https://doi.org/10.1016/j.softx.2019.100382

Sussman, M., Smereka, P., & Osher, S. (1994). A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow. *Journal of Computational Physics*, *114*(1), 146–159. https://doi.org/10.1006/jcph.1994.1155

Taghilou, M., & Rahimian, M. H. (2014). Investigation of two-phase flow in porous media using lattice Boltzmann method. *Computers & Mathematics with Applications*, *67*(2), 424–436. https://doi.org/10.1016/j.camwa.2013.08.005

Tananykhin, D., Korolev, M., Stecyuk, I., & Grigorev, M. (2021). An investigation into current sand control methodologies taking into account geomechanical, field and laboratory data analysis. *Resources*, *10*(12). https://doi.org/10.3390/resources10120125

Thakur, S. C., Morrissey, J. P., Sun, J., Chen, J. F., & Ooi, J. Y. (2014). Micromechanical analysis of

cohesive granular materials using the discrete element method with an adhesive elasto-plastic contact model. *Granular Matter*, *16*(3), 383–400. https://doi.org/10.1007/s10035-014-0506-4

Thornton, C. (1997). Coefficient of Restitution for Collinear Collisions of Elastic-Perfectly Plastic Spheres. *Journal of Applied Mechanics*, *64*(2), 383–386. https://doi.org/10.1115/1.2787319

Thornton, Colin. (2015). Granular Dynamics, Contact Mechanics and Particle System Simulations. In *Granular Dynamics, Contact Mechanics and Particle System Simulations: A DEM study* (Vol. 24). Springer International Publishing. https://doi.org/10.1007/978-3-319-18711-2

Thornton, Colin, Cummins, S. J., & Cleary, P. W. (2017). On elastic-plastic normal contact force models, with and without adhesion. *Powder Technology*, *315*(March), 339–346. https://doi.org/10.1016/j.powtec.2017.04.008

Thornton, Colin, & Ning, Z. (1998). A theoretical model for the stick/bounce behaviour of adhesive, elastic-plastic spheres. *Powder Technology*, *99*(2), 154–162. https://doi.org/10.1016/S0032-5910(98)00099-0

Tomas, J. (2001). Assessment of Mechanical Properties of Cohesive Particulate Solids. Part 1: Particle Contact Constitutive Model. *Particulate Science and Technology*, *19*(2), 95–110. https://doi.org/10.1080/02726350152772056

Tovar, J. (2007). Integrating Drilling and Geomechanical Damage in Sandstone Reservoirs: Identification, Quantification and Removal. *Proceedings of European Formation Damage Conference*, *1*, 330–337. https://doi.org/10.2523/107611-MS

Tremblay, B., Sedgwick, G., & Forshner, K. (1996, April 5). Modelling of Sand Production from Wells on Primary Recovery. *Annual Technical Meeting*. https://doi.org/10.2118/96-26

Tronvoll, J., Dusseault, M. B., Sanfilippo, F., & Santarelli, F. J. (2001, September 30). The Tools of Sand Management. *All Days*. https://doi.org/10.2118/71673-MS

Tronvoll, J., & Fjær, E. (1994). Experimental study of sand production from perforation cavities. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, *31*(5), 393–410. https://doi.org/10.1016/0148-9062(94)90144-9

Tsuji, Y., Kawaguchi, T., & Tanaka, T. (1993). Discrete particle simulation of two-dimensional fluidized bed. *Powder Technology*, *77*(1), 79–87. https://doi.org/10.1016/0032-5910(93)85010-7

Utili, S., & Nova, R. (2008). DEM analysis of bonded granular geomaterials. *International Journal for Numerical and Analytical Methods in Geomechanics*, *32*(17), 1997–2031. https://doi.org/10.1002/nag.728

Vångö, M. (2019). CFD-DEM modeling of multi- phase fluid-granular systems and its application to blast furnace tapping. In *Dissertation, Johannes Kepler Universität Linz*. https://epub.jku.at/obvulihs/content/titleinfo/3498046

Veeken, C. A. M., Davies, D. R., Kenter, C. J., & Kooijman, A. P. (1991). Sand production prediction review. Developing an integrated approach. *Proceedings - SPE Annual Technical Conference and Exhibition*, *Pi*(pt 1), 335–346. https://doi.org/10.2523/22792-ms

Verlet Loup. (1967). Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Journal of Physics D: Applied Physics*, *9*(2), 183–195. https://doi.org/10.1088/0022-3727/9/2/008

Vu-Quoc, L., & Zhang, X. (1999). An elastoplastic contact force-displacement model in the normal direction: Displacement-driven version. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *455*(1991), 4013–4044. https://doi.org/10.1098/rspa.1999.0488

Wacławczyk, T. (2017). On a relation between the volume of fluid, level-set and phase field interface models. *International Journal of Multiphase Flow*, *97*, 60–77. https://doi.org/10.1016/j.ijmultiphaseflow.2017.08.003

Wan, R. G., Liu, Y., & Wang, J. (2004). A Multiphase Flow Approach to Modelling Sand Production

Using Finite Elements. *Canadian International Petroleum Conference*, 1–9. https://doi.org/10.2118/2004-277

Wang, H., Gala, D. P., & Sharma, M. M. (2019). Effect of Fluid Type and Multiphase Flow on Sand Production in Oil and Gas Wells. *SPE Journal*, *24*(02), 733–743. https://doi.org/10.2118/187117-PA

Wang, Y., & Xue, S. (2002). Coupled Reservoir-Geomechanics Model With Sand Erosion for Sand Rate and Enhanced Production Prediction. *All Days*, 373–383. https://doi.org/10.2118/73738-MS

Wischnewski, B. (n.d.). *peacesoftware*. https://www.peacesoftware.de/einigewerte/calc_co2.php7

Wu, B, Choi, S. K., Denke, R., Barton, T., Viswanathan, C., Lim, S., Zamberi, M., Shaffee, S., Fadhlan, N., Johar, Z., Jadid, M. B., & Madon, B. B. (2016, March 22). A New and Practical Model for Amount and Rate of Sand Production Estimation. *Day 3 Thu, March 24, 2016*. https://doi.org/10.4043/26508-MS

Wu, Bailin, Tan, C. P., & Lu, N. (2005). Effect of Water-Cut on Sand Production - An Experimental Study. *All Days*, *October 2005*, 349–356. https://doi.org/10.2118/92715-MS

Yan, B., & Regueiro, R. A. (2018). Superlinear speedup phenomenon in parallel 3D Discrete Element Method (DEM) simulations of complex-shaped particles. *Parallel Computing*, *75*, 61–87. https://doi.org/10.1016/j.parco.2018.03.007

Yan, Y. Y. (2007). Recent advances in computational simulation of macro-, meso-, and micro-scale biomimetics related fluid flow problems. *Journal of Bionic Engineering*, *4*(2), 97–107. https://doi.org/10.1016/S1672-6529(07)60021-3

YOUNGS, D. L. (1982). *Time-dependent multi-material flow with large fluid distortion*. Academic Press. https://ci.nii.ac.jp/naid/10029507464/

Zhang, X., & Vu-Quoc, L. (2002). Modeling the dependence of the coefficient of restitution on the impact velocity in elasto-plastic collisions. *International Journal of Impact Engineering*, *27*(3), 317–341. https://doi.org/10.1016/S0734-743X(01)00052-5

Zhao, J., & Shan, T. (2013). Coupled CFD–DEM simulation of fluid–particle interaction in geomechanics. *Powder Technology*, *239*, 248–258. https://doi.org/10.1016/j.powtec.2013.02.003

Zhou, Z. Y., Kuang, S. B., Chu, K. W., & Yu, A. B. (2010). Discrete particle simulation of particle–fluid flow: model formulations and their applicability. *Journal of Fluid Mechanics*, *661*, 482–510. https://doi.org/10.1017/S002211201000306X

Zhou, Z. Y., Yu, A. B., & Choi, S. K. (2011). Numerical simulation of the liquid-induced erosion in a weakly bonded sand assembly. *Powder Technology*, *211*(2–3), 237–249. https://doi.org/10.1016/j.powtec.2011.04.029

Zhu, H. P., Zhou, Z. Y., Yang, R. Y., & Yu, A. B. (2007). Discrete particle simulation of particulate systems: Theoretical developments. *Chemical Engineering Science*, *62*(13), 3378–3396. https://doi.org/10.1016/j.ces.2006.12.089

APPENDIX. DATA POST-PROCESSING C# CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;
using System.Globalization;
using System.Diagnostics;
using System.Threading;
namespace MON_W_W
{
class Program
{
public static double[] quartile = new double[3];
public static void Quartiles(double[] massive)
{
double min;
min = massive[0];
for (int i = 0; i < massive.Length; i++)
for (int j = i + 1; j < massive.Length; j++)
if (massive[j] < massive[i])
{
min = massive[j];
double tem = massive[i];
massive[i] = massive[j];
massive[j] = tem;
}
quartile[0] = massive[Convert.ToInt32(massive.Length / 4)];
quartile[1] = massive[Convert.ToInt32(massive.Length / 4) * 2];
quartile[2] = massive[Convert.ToInt32(massive.Length / 4) * 3];
}


static void Main(string[] args)
{
Console.WriteLine("Hello Furkhat DragForce Reader V2!");

string FileNameLong_z2, FileNameLong_z1, FileNameLong_x1, FileNameLong_x2, FileNameLong_y1, FileNameLong_y2;

int counter1 = 0, counter2 = 0, counterFiles = 0;

const int particle_number = 100000;
int start_timestep = 420000;
int end_timestep = 4440000;
double PerfRadius = 0.0007;//0.0007;
double OuterRadius = 0.00756;//0.00756;
int N_xy = (int)(OuterRadius / PerfRadius);
double N_z = 3.0;
double Z_max = 0.006536;
double[] PSD = { 7.50E-05, 9.00E-05, 1.00E-04, 0.00011, 0.000125, 0.0001375, 0.00015, 0.0001775 };
double Z_cut = 0.00672;//but some zmax >0.0068 z+r
int frequence_of_prob = 2000;
int couplingInterval = 10;
double particleDensity = 2500;
int numPerR = 2;
int forceFrequency = 1;
int number_of_cores = 8;

string FileName_z2 = @"E:\Programming\DragForce_MON\SP_W-W\SP_run\post_SP_run_real";
string FileNameShort_Drag = @"E:\Programming\DragForce_MON\particleProbes_W-W\particleProbes\0\DiFeliceDrag.logDat.";
string FileNameShort_Visc = @"E:\Programming\DragForce_MON\particleProbes_W-W\particleProbes\0\viscForce.logDat.";
string FileNameShort_GradP = @"E:\Programming\DragForce_MON\particleProbes_W-W\particleProbes\0\gradPForce.logDat.";
string FileNamePrintAll = @"E:\Programming\DragForce_MON\SP_W-W\ScalarResultsTotal.txt";
string FileNamePrintPart = @"E:\Programming\DragForce_MON\SP_W-W\ScalarResultsPart.txt";
string FileNamePrintVTK = @"E:\Programming\DragForce_MON\SP_W-Wn\SP_run_drag\post_SP_run_real";



double DEM_time_discrete = 2 * Convert.ToDouble(Math.Pow(10, -8));
double CFD_time_discrete = 2 * Convert.ToDouble(Math.Pow(10, -7));

int particleProb_time = Convert.ToInt32(Convert.ToDouble(end_timestep - start_timestep) * DEM_time_discrete / CFD_time_discrete / frequence_of_prob) - 1;
int particleProb_time_2 = particleProb_time;// 2;

double probTimeDiscrete = CFD_time_discrete * (double)frequence_of_prob;


double sampleVolume = Math.PI * OuterRadius * OuterRadius * Z_max;
//Console.WriteLine("probTimeDiscrete ={0}", probTimeDiscrete);

int[] numberParticlesTime = new int[particleProb_time];
string line;

Stopwatch sWatch = new Stopwatch();

//reading from run.VTK
```

```
double[,] particleDataFromVtk = new double[particle_number * particleProb_time, 12];//time, position(3 values), velocities(3 values),velocity magnitude, velocity_norm, r, coorAll,coorBond
bool flag_reading_vtk = true;

if (flag_reading_vtk)
{
sWatch.Start();
bool flag1, flagVelPart, flagXYZ, flagRadius, flagCoorAll, flagCoorBond;
flag1 = flagVelPart = flagXYZ = flagCoorAll = flagCoorBond = flagRadius = false;
int count = 0;
int countTotal = 0;



for (int l = 0; l < particleProb_time; l++)
{
flag1 = false; flagVelPart = false; flagXYZ = false; flagCoorAll = false; flagCoorBond = false; flagRadius = false;

int current_timeStep = start_timestep + (l + 1) * frequence_of_prob * couplingInterval;
string FileNameLong = FileName_z2 + Convert.ToString(current_timeStep) + Convert.ToString(".vtk");

StreamReader file_2 = new StreamReader(FileNameLong);

Console.WriteLine("Reading data from VTK file " + (current_timeStep));
//file.BaseStream.Position = 0;

while ((line = file_2.ReadLine()) != null)
{

string[] split = line.Split(' ');
if (Convert.ToString(split[0]) == "POINTS")
{
flagXYZ = true;
count = 0 + countTotal;
continue;
}

if (Convert.ToString(split[0]) == "VERTICES")
{
flagXYZ = false;
count = 0 + countTotal;
continue;
}
if (Convert.ToString(split[0]) == "v")
{
flagVelPart = true;
continue;
}
if (Convert.ToString(split[0]) == "radius")
{
flagVelPart = false;
flagRadius = true;
count = 0 + countTotal;
continue;
}
if (Convert.ToString(split[0]) == "omega")
{
flagRadius = false;
flagVelPart = false;
continue;
}
if (Convert.ToString(split[0]) == "f")
{
flagVelPart = false;
flagRadius = false;
continue;
}
if (Convert.ToString(split[0]) == "f_dragforce[1-3]")
{
flagVelPart = false;
flagRadius = false;
continue;
}
if (Convert.ToString(split[0]) == "c_coordAll")
{
flagVelPart = false;
flagRadius = false;
flagCoorAll = true;
flagCoorBond = false;
count = 0 + countTotal;
continue;
}
if (Convert.ToString(split[0]) == "c_coordBonded")
{
flagVelPart = false;
flagRadius = false;
flagCoorAll = false;
flagCoorBond = true;
count = 0 + countTotal;
continue;
}

if (flagXYZ)
{

for (int j = 0; j < split.Length - 1; j++) //coordinate
{
if (j % 3 == 0)
{
particleDataFromVtk[count, 1] = double.Parse(split[j]);
continue;
}
if (j % 3 == 1)
{
particleDataFromVtk[count, 2] = double.Parse(split[j]);
continue;
```

```
}
if (j % 3 == 2)
{
particleDataFromVtk[count, 3] = double.Parse(split[j]);
particleDataFromVtk[count, 0] = (double)((current_timeStep - start_timestep) * DEM_time_discrete);
count++;
continue;
}
}
}
if (flagVelPart)
for (int j = 0; j < split.Length - 1; j++)//velocity
{
if (j % 3 == 0)
{
particleDataFromVtk[count, 4] = double.Parse(split[j]);
continue;
}
if (j % 3 == 1)
{
particleDataFromVtk[count, 5] = double.Parse(split[j]);
continue;
}
if (j % 3 == 2)
{
particleDataFromVtk[count, 6] = double.Parse(split[j]);
particleDataFromVtk[count, 0] = (double)((current_timeStep - start_timestep) * DEM_time_discrete);
count++;
continue;
}
}
if (flagRadius)
for (int j = 0; j < split.Length - 1; j++)
{
particleDataFromVtk[count, 9] = double.Parse(split[j]);
count++;
continue;
}
if (flagCoorAll)
for (int j = 0; j < split.Length - 1; j++)
{
particleDataFromVtk[count, 10] = double.Parse(split[j]);
count++;
continue;
}
if (flagCoorBond)
for (int j = 0; j < split.Length - 1; j++)
{
particleDataFromVtk[count, 11] = double.Parse(split[j]);
count++;
continue;
}
}

count = count - countTotal;
numberParticlesTime[l] = count;
file_2.Close();
countTotal = countTotal + count;

}

sWatch.Stop();

TimeSpan tSpan = sWatch.Elapsed;
// Format and display the TimeSpan value.
string elapsedTime = String.Format("{0:00}:{1:00}:{2:00}.{3:00}", tSpan.Hours, tSpan.Minutes, tSpan.Seconds, tSpan.Milliseconds / 10);
Console.WriteLine("RunTime of vtkReader " + elapsedTime);

}
//подсчет
int sumOfParticles = 0;
for (int i = 0; i < numberParticlesTime.Length; i++)
sumOfParticles += numberParticlesTime[i];
Console.WriteLine($"Accurate number of elements in massive: {sumOfParticles}");

//создаем массив с точным количеством частиц, без излишеств
double[,] particleDataFromVtkAccurate = new double[sumOfParticles, 12];
int temp = 0;
for (int i = 0; i < particleDataFromVtk.GetLength(0); i++)
{
if (particleDataFromVtk[i, 0] != 0)
for (int ii = 0; ii < particleDataFromVtk.GetLength(1); ii++)
particleDataFromVtkAccurate[temp, ii] = particleDataFromVtk[i, ii];
temp++;
}

double[,] DragForceCoef = new double[sumOfParticles, 15];//time, position(3 values), dragForce(3 values), Urel(3 values), //Rep,Cd, voidfraction, Velocity Magnitude, Velocity_quartile average
double[,] ViscousForceCoef = new double[sumOfParticles, 9];//time, position(3 values), viscousForce(3 values), Us(1 values), viscous Magnitude average
double[,] GradientPForceCoef = new double[sumOfParticles, 10];//time, position(3 values), GradientPForce(3 values), Us(1 values), pho, gradientP Magnitude average
int[] indexVTKtoDrag = new int[sumOfParticles];
```

```
bool flag_particleProbes = true;
bool flag_particleProbesDrag = true;
bool flag_particleProbesVisc = true;
bool flag_particleProbesGradP = true;
if (flag_particleProbes)
{
sWatch.Start();
if (flag_particleProbesDrag)
{


//string writePath = @"E:\Programming\DragForce\HO_particleProbes\particleProbes\0\dragForce.txt";
bool flag = false;
int[] currentIndexDrafForce = new int[particleProb_time];
currentIndexDrafForce[0] = 0;
for (int i = 1; i < numberParticlesTime.Length; i++)
{
currentIndexDrafForce[i] = currentIndexDrafForce[i - 1] + numberParticlesTime[i];
//Console.WriteLine($"The begin/current index of probeTimeDrag : {currentIndexDrafForce[i]}");
}

//reding from particleProbes
int index_current = 0;
for (int i = 0; i < number_of_cores; i++)
{
string FileNameShort_Drag1 = FileNameShort_Drag + Convert.ToString(i);// Convert.ToString(".vtk");
StreamReader file_z1 = new StreamReader(FileNameShort_Drag1);

flag = false;
//Console.WriteLine("Reading data from file " + (i));
while ((line = file_z1.ReadLine()) != null)
{
string[] split = line.Split(' ', '|', "", ");

//System.Console.WriteLine(line);
if (split[0] == "#index")
{
flag = true;
continue;
}



if (flag)
{
int l = (int)(Convert.ToDouble(string.Format("{0:0.0000}", split[3])) / (double)probTimeDiscrete) - 1;
if (l > 350)
{
int sdfsdfsd = 0;
}
index_current = currentIndexDrafForce[l];
DragForceCoef[index_current, 0] = double.Parse(split[3]);//time
DragForceCoef[index_current, 1] = double.Parse(split[48]); //position(3 values)
DragForceCoef[index_current, 2] = double.Parse(split[51]);
DragForceCoef[index_current, 3] = double.Parse(split[54]);
DragForceCoef[index_current, 4] = double.Parse(split[11]); //dragForce(3 values)
DragForceCoef[index_current, 5] = double.Parse(split[14]);
DragForceCoef[index_current, 6] = double.Parse(split[17]);
DragForceCoef[index_current, 7] = double.Parse(split[20]);//Urel(3 values)
DragForceCoef[index_current, 8] = double.Parse(split[23]);
DragForceCoef[index_current, 9] = double.Parse(split[26]);
DragForceCoef[index_current, 10] = double.Parse(split[34]);// Rep
DragForceCoef[index_current, 11] = double.Parse(split[37]);// Cd
DragForceCoef[index_current, 12] = double.Parse(split[40]); //voidfraction
currentIndexDrafForce[l]++;
}
}
file_z1.Close();
counterFiles++;
Console.WriteLine("{0}-file dragReading", i);

}
}
if (flag_particleProbesVisc)
{



bool flag = false;
int[] currentIndexDrafForce = new int[particleProb_time];
currentIndexDrafForce[0] = 0;
for (int i = 1; i < numberParticlesTime.Length; i++)
{
currentIndexDrafForce[i] = currentIndexDrafForce[i - 1] + numberParticlesTime[i];
//Console.WriteLine($"The begin/current index of probeTimeVisc : {currentIndexDrafForce[i]}");
}

//reding from particleProbes
int index_current = 0;
for (int i = 0; i < number_of_cores; i++)
{
```

```
string FileNameShort_Visc1 = FileNameShort_Visc + Convert.ToString(i);// Convert.ToString(".vtk");
StreamReader file_z1 = new StreamReader(FileNameShort_Visc1);

flag = false;
//Console.WriteLine("Reading data from file " + (i));
while ((line = file_z1.ReadLine()) != null)
{
string[] split = line.Split(' ', '|', "", ");

//System.Console.WriteLine(line);
if (split[0] == "#index")
{
flag = true;
continue;
}

if (flag)
{
int l = (int)(double.Parse(split[3]) / (double)probTimeDiscrete) - 1;
index_current = currentIndexDrafForce[l];

ViscousForceCoef[index_current, 0] = double.Parse(split[3]);//time
ViscousForceCoef[index_current, 1] = double.Parse(split[33]); //position(3 values)
ViscousForceCoef[index_current, 2] = double.Parse(split[36]);
ViscousForceCoef[index_current, 3] = double.Parse(split[39]);
ViscousForceCoef[index_current, 4] = double.Parse(split[11]); //ViscForce(3 values)
ViscousForceCoef[index_current, 5] = double.Parse(split[14]);
ViscousForceCoef[index_current, 6] = double.Parse(split[17]);
ViscousForceCoef[index_current, 7] = double.Parse(split[25]);//Us(1 values)

currentIndexDrafForce[l]++;
}
}
file_z1.Close();
counterFiles++;
Console.WriteLine("{0}-file ViscousReading", i);

}
}
if (flag_particleProbesGradP)
{



bool flag = false;
int[] currentIndexDrafForce = new int[particleProb_time];
currentIndexDrafForce[0] = 0;
for (int i = 1; i < numberParticlesTime.Length; i++)
{
currentIndexDrafForce[i] = currentIndexDrafForce[i - 1] + numberParticlesTime[i];
//Console.WriteLine($"The begin/current index of probeTimeVisc : {currentIndexDrafForce[i]}");
}

//reding from particleProbes
int index_current = 0;
for (int i = 0; i < number_of_cores; i++)
{
string FileNameShort_GradP1 = FileNameShort_GradP + Convert.ToString(i);// Convert.ToString(".vtk");
StreamReader file_z1 = new StreamReader(FileNameShort_GradP1);

flag = false;
//Console.WriteLine("Reading data from file " + (i));
while ((line = file_z1.ReadLine()) != null)
{
string[] split = line.Split(' ', '|', "", ");

//System.Console.WriteLine(line);
if (split[0] == "#index")
{
flag = true;
continue;
}


if (flag)
{
int l = (int)(double.Parse(split[3]) / (double)probTimeDiscrete) - 1;
index_current = currentIndexDrafForce[l];

GradientPForceCoef[index_current, 0] = double.Parse(split[3]);//time
GradientPForceCoef[index_current, 1] = double.Parse(split[36]); //position(3 values)
GradientPForceCoef[index_current, 2] = double.Parse(split[39]);
GradientPForceCoef[index_current, 3] = double.Parse(split[42]);
GradientPForceCoef[index_current, 4] = double.Parse(split[11]); //ViscForce(3 values)
GradientPForceCoef[index_current, 5] = double.Parse(split[14]);
GradientPForceCoef[index_current, 6] = double.Parse(split[17]);
GradientPForceCoef[index_current, 7] = double.Parse(split[25]);//Us(1 values)
GradientPForceCoef[index_current, 8] = double.Parse(split[28]);//Us(1 values)
currentIndexDrafForce[l]++;
}
}
file_z1.Close();
```

```
counterFiles++;
Console.WriteLine("{0}-file GradPReading", i);

}
}

sWatch.Stop();
TimeSpan tSpan = sWatch.Elapsed;
// Format and display the TimeSpan value.
string elapsedTime = String.Format("{0:00}:{1:00}:{2:00}.{3:00}", tSpan.Hours, tSpan.Minutes, tSpan.Seconds, tSpan.Milliseconds / 10);
Console.WriteLine("RunTime of ForceReader " + elapsedTime);
}

//manipulation with data
bool flag_manipulation = true;
flag_particleProbes = true;
if (flag_manipulation && flag_reading_vtk && flag_particleProbes)
{
sWatch.Start();

//sorting the data to speed up
int[] currentIndexDrafForce = new int[particleProb_time];
currentIndexDrafForce[0] = 0;
for (int i = 1; i < numberParticlesTime.Length; i++)
{
currentIndexDrafForce[i] = currentIndexDrafForce[i - 1] + numberParticlesTime[i];
//Console.WriteLine($"The begin index of probeTime Manipulation data : {currentIndexDrafForce[i]}");
}
//the most inportant parts!
for (int i = 0; i < particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0] / (double)probTimeDiscrete) - 1;
if (l % forceFrequency == 0)
{
int begin_index = currentIndexDrafForce[l];
int end_index = begin_index + numberParticlesTime[l] - 1;
for (int ii = begin_index; ii <= end_index; ii++)
if (particleDataFromVtkAccurate[i, 1] == DragForceCoef[ii, 1])
if (particleDataFromVtkAccurate[i, 2] == DragForceCoef[ii, 2])
if (particleDataFromVtkAccurate[i, 3] == DragForceCoef[ii, 3])
{
indexVTKtoDrag[i] = ii;
break;
}
}
}

//for (int u = 0; u < sumOfParticles; u++)
//indexVTKtoDrag[u] = u;


//Magnitude of dragForce and velocities ; //time, position(3 values), dragForce(3 values), Urel(3 values), Rep,Cd, voidfraction, Drag Magnitude
for (int i = 0; i < sumOfParticles; i++)
{
DragForceCoef[i, 13] = (double)Math.Sqrt(DragForceCoef[i, 4] * DragForceCoef[i, 4] + DragForceCoef[i, 5] * DragForceCoef[i, 5] + DragForceCoef[i, 6] * DragForceCoef[i, 6]);
ViscousForceCoef[i, 8] = (double)Math.Sqrt(ViscousForceCoef[i, 4] * ViscousForceCoef[i, 4] + ViscousForceCoef[i, 5] * ViscousForceCoef[i, 5] + ViscousForceCoef[i, 6] * ViscousForceCoef[i, 6]);
GradientPForceCoef[i, 9] = (double)Math.Sqrt(GradientPForceCoef[i, 4] * GradientPForceCoef[i, 4] + GradientPForceCoef[i, 5] * GradientPForceCoef[i, 5] + GradientPForceCoef[i, 6] * GradientPForceCoef[i, 6]);
particleDataFromVtkAccurate[i, 7] = (double)Math.Sqrt(particleDataFromVtkAccurate[i, 4] * particleDataFromVtkAccurate[i, 4] + particleDataFromVtkAccurate[i, 5] * particleDataFromVtkAccurate[i, 5] + particleDataFromVtkAccurate[i, 6] * particleDataFromVtkAccurate[i, 6]);

}

//calculate averaged particle Magnitude velocities: 1,2,3,4 quartile;
//double[,] particleDataFromVtk = new double[particle_number * particleProb_time, 8];//time, position(3 values), velocities(3 values), velocity magnitude


//the Storage for quantile values

/*double[] TempDragMagnit = new double[particle_number];
double[] TempVelMagnit = new double[particle_number];
double[,,] QuartileTimeParam = new double[2, particleProb_time, 3]; // 0-DragMagnit,1-VelMagnit; particleProb_time for all time; q1,q2,q3


for (int j = 0; j < particleProb_time_2; j++)
{
int current_timeStep = start_timestep + (j + 1) * 5000;
```

```
double relative_time = (double)((current_timeStep - start_timestep) * DEM_time_discrete);
int k = 0; int k1 = 0;
for (int i = 0; i < sumOfParticles; i++)
{
if (DragForceCoef[i, 0] == relative_time)
{
TempDragMagnit[k] = (double)DragForceCoef[i, 13];
k++;
}
if (particleDataFromVtk[i, 0] == relative_time)
{
TempVelMagnit[k1] = (double)particleDataFromVtkAccurate[i, 7];
k1++;
}
}
////////////////find new quartile1-3 for TempDragMagnit//////////////////////////////////////////////////////
////
Quartiles(TempVelMagnit);
for (int t = 0; t < 3; t++)
QuartileTimeParam[0, j, t] = quartile[t];

Quartiles(TempDragMagnit);
for (int t = 0; t < 3; t++)
QuartileTimeParam[1, j, t] = quartile[t];
Console.WriteLine("Run QuartileCalculation " + (j));
}

//calculation of quartiles for dragForce
for (int i = 0; i < sumOfParticles; i++)
{

double time = DragForceCoef[i, 0];//
int quartel_index = (int)(time / 0.0001) - 1;
if (quartel_index >= 0)
{
if (DragForceCoef[i, 13] < QuartileTimeParam[1, quartel_index, 0])
{
DragForceCoef[i, 14] = (double)1;
continue;
}

if ((DragForceCoef[i, 13] >= QuartileTimeParam[1, quartel_index, 0]) && (DragForceCoef[i, 13] < QuartileTimeParam[1, quartel_index, 1]))
{
DragForceCoef[i, 14] = (double)2;
continue;
}

if ((DragForceCoef[i, 13] >= QuartileTimeParam[1, quartel_index, 1]) && (DragForceCoef[i, 13] < QuartileTimeParam[1, quartel_index, 2]))
{
DragForceCoef[i, 14] = (double)3;
continue;
}

if (DragForceCoef[i, 13] >= QuartileTimeParam[1, quartel_index, 2])
{
DragForceCoef[i, 14] = (double)4;
continue;
}
}

}
//calculation of quartiles for Velocity
for (int i = 0; i < sumOfParticles; i++)
{

double time = particleDataFromVtk[i, 0];//
int quartel_index = (int)(time / 0.0001) - 1;
if (quartel_index >= 0)
{
if (particleDataFromVtkAccurate[i, 7] < QuartileTimeParam[0, quartel_index, 0])
{
particleDataFromVtkAccurate[i, 8] = (double)1;
continue;
}

if ((particleDataFromVtkAccurate[i, 7] >= QuartileTimeParam[0, quartel_index, 0]) && (particleDataFromVtkAccurate[i, 7] < QuartileTimeParam[0, quartel_index, 1]))
{
particleDataFromVtkAccurate[i, 8] = (double)2;
continue;
}

if ((particleDataFromVtkAccurate[i, 7] >= QuartileTimeParam[0, quartel_index, 1]) && (particleDataFromVtkAccurate[i, 7] < QuartileTimeParam[0, quartel_index, 2]))
{
particleDataFromVtkAccurate[i, 8] = (double)3;
continue;
}

if (particleDataFromVtkAccurate[i, 7] >= QuartileTimeParam[0, quartel_index, 2])
{
particleDataFromVtkAccurate[i, 8] = (double)4;
continue;
}
}

}*/
//Normalize velocity field
double[] averageVeloc = new double[particleProb_time];
for (int i = 0; i < particleDataFromVtkAccurate.GetLength(0); i++)
{
```

```
int l = (int)((double)particleDataFromVtkAccurate[i, 0] / (double)probTimeDiscrete) - 1;
averageVeloc[l] += particleDataFromVtkAccurate[i, 7] / numberParticlesTime[l];
}
for (int i = 0; i < particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0] / (double)probTimeDiscrete) - 1;
particleDataFromVtkAccurate[i, 8] = particleDataFromVtkAccurate[i, 7] / (0.000139803);// averageVeloc[l];
}
sWatch.Stop();

TimeSpan tSpan = sWatch.Elapsed;
// Format and display the TimeSpan value.
string elapsedTime = String.Format("{0:00}:{1:00}:{2:00}.{3:00}", tSpan.Hours, tSpan.Minutes, tSpan.Seconds, tSpan.Milliseconds / 10);
Console.WriteLine("RunTime of Manipulation The main important part " + elapsedTime);

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

}
else { Console.WriteLine("You must read all necessary data!"); }


//1) coord number, bond_number, porosity, the all 3 forces(drag, visc, gradP), center of mass(z, x*x+y*y), particle velocity
//particleDataFromVtkAccurate = new double[particle_number * particleProb_time, 12];//time, position(3 values), velocities(3 values),velocity magnitude, velocity_norm, r, coorAll,coorBond
bool flagScalardataTotal = true;
if (flagScalardataTotal)
{
int nubSubZ = (int)(Z_max / PerfRadius / numPerR) - 1;
double[,] particlesVolumeInZ = new double[particleProb_time, nubSubZ];
double[,] particlesNumberInZ = new double[particleProb_time, nubSubZ];
double[,,] scalarParamerersOutputVolumeInZ = new double[particleProb_time, nubSubZ, 13];//0-2 -center of massZ,X,Y;3-5 -center of mass VelXYZ; 6-center of massR;9-Drag;12-Visc

double[] sampleVolumeIn = new double[nubSubZ];

double[,] scalarParamerersOutput = new double[particleProb_time, 20]; //coord number(0), bond_number(1),porosity(2),vel magnitude(3),drag(4),visc(5),gradP(6), center of massZ(7),
 //center of massX(8), center of massY(9), center of VelX(10),center of VelY(11),center of VelZ(12),center of massR(13), DragForce(16),GradP(19)
double[] particlesVolumeIn = new double[particleProb_time];// volume of particles inside the cylinder
double[] particlesNumberIn = new double[particleProb_time];
double[] particlesNumberInCoorBond = new double[particleProb_time];
double[,] numberPSDProbTime = new double[particleProb_time, PSD.Length];
double[] numberPSDzeroTime = new double[PSD.Length];
double[,] delatedPSDProbTime = new double[particleProb_time, PSD.Length];
string OutputGeneral = string.Empty;

for (int kk = 0; kk < nubSubZ; kk++)
sampleVolumeIn[kk] = 0.5 * 4.0 / 3.0 * Math.PI * ((kk + 1) * (kk + 1) * (kk + 1) - kk * kk * kk) * (PerfRadius * numPerR) * (PerfRadius * numPerR) * (PerfRadius * numPerR);

for (int i = 0; i < particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0] / (double)probTimeDiscrete) - 1;

if (l == 0)
for (int jj = 0; jj < PSD.Length; jj++)
if (particleDataFromVtkAccurate[i, 9] == PSD[jj])
{
numberPSDzeroTime[jj] += 1;
break;
}
particlesNumberInCoorBond[l] += 1;////
if (particleDataFromVtkAccurate[i, 3] - particleDataFromVtkAccurate[i, 9] <= Z_max)
{
if (particleDataFromVtkAccurate[i, 3] <= Z_max)
{
double rCurrent = Math.Sqrt(particleDataFromVtkAccurate[i, 1] * particleDataFromVtkAccurate[i, 1] + particleDataFromVtkAccurate[i, 2] * particleDataFromVtkAccurate[i, 2] + (particleDataFromVtkAccurate[i, 3] - Z_max) * (particleDataFromVtkAccurate[i, 3] - Z_max));
int rIndex = 0;
for (int gg = 0; gg < nubSubZ; gg++)
if ((numPerR * PerfRadius * gg <= rCurrent) && (numPerR * PerfRadius * (gg + 1) > rCurrent))
{
rIndex = gg;
```

```
particlesVolumeInZ[l, rIndex] += (4.0 / 3.0) * Math.PI
* Math.Pow(particleDataFromVtkAccurate[i, 9], 3);
particlesNumberInZ[l, rIndex] += 1;

scalarParamerersOutputVolumeInZ[l, rIndex, 0] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 3]) /
sampleVolumeIn[rIndex];//CenterOfMassZ
scalarParamerersOutputVolumeInZ[l, rIndex, 1] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 1]) /
sampleVolumeIn[rIndex];//CenterOfMassX
scalarParamerersOutputVolumeInZ[l, rIndex, 2] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 2]) /
sampleVolumeIn[rIndex];//CenterOfMassY
scalarParamerersOutputVolumeInZ[l, rIndex, 3] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 4]) /
sampleVolumeIn[rIndex];//CenterOfMassVelX
scalarParamerersOutputVolumeInZ[l, rIndex, 4] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 5]) /
sampleVolumeIn[rIndex];//CenterOfMassVelY
scalarParamerersOutputVolumeInZ[l, rIndex, 5] +=
((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 6]) /
sampleVolumeIn[rIndex];//CenterOfMassZVelZ

break;
}
}

particlesVolumeIn[l] += (4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3);
particlesNumberIn[l] += 1;
for (int ii = 0; ii < PSD.Length; ii++)
if (particleDataFromVtkAccurate[i, 9] == PSD[ii])
{
numberPSDProbTime[l, ii] += 1;
break;
}
//if (particleDataFromVtkAccurate[i, 11] >= 1)
//particlesNumberInCoorBond[l] += 1;


}
}

for (int i = 0; i <
particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0]
/ (double)probTimeDiscrete) - 1;
if (l % forceFrequency == 0)
{
scalarParamerersOutput[l, 1] +=
particleDataFromVtkAccurate[i, 11] /
particlesNumberInCoorBond[l];//bond number
if (particleDataFromVtkAccurate[i, 3] -
particleDataFromVtkAccurate[i, 9] <= Z_max)
{
if (particleDataFromVtkAccurate[i, 3] <= Z_max)
{
double rCurrent =
Math.Sqrt(particleDataFromVtkAccurate[i, 1] *
particleDataFromVtkAccurate[i, 1] +
particleDataFromVtkAccurate[i, 2] *
particleDataFromVtkAccurate[i, 2] +
(particleDataFromVtkAccurate[i, 3] - Z_max) *
(particleDataFromVtkAccurate[i, 3] - Z_max));
int rIndex = 0;
for (int gg = 0; gg < nubSubZ; gg++)
if ((numPerR * PerfRadius * gg <= rCurrent) &&
(numPerR * PerfRadius * (gg + 1) > rCurrent))
{
rIndex = gg;

scalarParamerersOutputVolumeInZ[l, rIndex, 7] +=
(DragForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberInZ[l, rIndex];//Center of massDragX
scalarParamerersOutputVolumeInZ[l, rIndex, 8] +=
(DragForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberInZ[l, rIndex];//Center of massDragY
scalarParamerersOutputVolumeInZ[l, rIndex, 9] +=
(DragForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberInZ[l, rIndex];//Center of massDragZ
scalarParamerersOutputVolumeInZ[l, rIndex, 10] +=
(GradientPForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberInZ[l, rIndex];//Center of massGradX
scalarParamerersOutputVolumeInZ[l, rIndex, 11] +=
(GradientPForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberInZ[l, rIndex];//Center of massGradY
scalarParamerersOutputVolumeInZ[l, rIndex, 12] +=
(GradientPForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberInZ[l, rIndex];//Center of massGradZ


break;
}
}

//if (particleDataFromVtkAccurate[i, 11] >= 1)
//scalarParamerersOutput[l, 1] +=
particleDataFromVtkAccurate[i, 11] /
particlesNumberInCoorBond[l];//bond number

scalarParamerersOutput[l, 0] +=
particleDataFromVtkAccurate[i, 10] /
particlesNumberIn[l]; //coord number
```

```
scalarParamerersOutput[l, 2] += (4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) /
sampleVolume;
scalarParamerersOutput[l, 3] +=
particleDataFromVtkAccurate[i, 7] /
particlesNumberIn[l];//vel mag
scalarParamerersOutput[l, 4] +=
DragForceCoef[indexVTKtoDrag[i], 13] /
particlesNumberIn[l];//drag
scalarParamerersOutput[l, 5] +=
ViscousForceCoef[indexVTKtoDrag[i], 8] /
particlesNumberIn[l];//visc
scalarParamerersOutput[l, 6] +=
GradientPForceCoef[indexVTKtoDrag[i], 9] /
particlesNumberIn[l];//gradP
scalarParamerersOutput[l, 7] += ((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 3]) /
particlesVolumeIn[l];//CenterOfMassZ
scalarParamerersOutput[l, 8] += ((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 1]) /
particlesVolumeIn[l];//Center of massX
scalarParamerersOutput[l, 9] += ((4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 2]) /
particlesVolumeIn[l];//Center of massY

scalarParamerersOutput[l, 10] += ((4.0 / 3.0) * Math.PI
* Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 4]) /
particlesVolumeIn[l];//Center of massVelocX
scalarParamerersOutput[l, 11] += ((4.0 / 3.0) * Math.PI
* Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 5]) /
particlesVolumeIn[l];//Center of massVelocY
scalarParamerersOutput[l, 12] += ((4.0 / 3.0) * Math.PI
* Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 6]) /
particlesVolumeIn[l];//Center of massVelocZ

scalarParamerersOutput[l, 14] +=
(DragForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberIn[l];//Center of massDragX
scalarParamerersOutput[l, 15] +=
(DragForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberIn[l];//Center of massDragY
scalarParamerersOutput[l, 16] +=
(DragForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberIn[l];//Center of massDragZ
scalarParamerersOutput[l, 17] +=
(GradientPForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberIn[l];//Center of massGradX
scalarParamerersOutput[l, 18] +=
(GradientPForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberIn[l];//Center of massGradY
scalarParamerersOutput[l, 19] +=
(GradientPForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberIn[l];//Center of massGradZ
}
}
}

for (int t = 0; t < particleProb_time; t++)
{
scalarParamerersOutput[t, 13] =
Math.Sqrt(scalarParamerersOutput[t, 8] *
scalarParamerersOutput[t, 8] +
scalarParamerersOutput[t, 9] *
scalarParamerersOutput[t, 9]);
for (int tt = 0; tt < nubSubZ; tt++)
scalarParamerersOutputVolumeInZ[t, tt, 6] =
Math.Sqrt(scalarParamerersOutputVolumeInZ[t, tt, 1] *
scalarParamerersOutputVolumeInZ[t, tt, 1] +
scalarParamerersOutputVolumeInZ[t, tt, 2] *
scalarParamerersOutputVolumeInZ[t, tt, 2]);
}



for (int t = 0; t < particleProb_time; t++)
for (int tt = 0; tt < PSD.Length; tt++)
delatedPSDProbTime[t, tt] = numberPSDzeroTime[tt] -
numberPSDProbTime[t, tt]; //deleated particles PSD
  //to write data
OutputGeneral = "Time number coord_all bond poro
VelMagAver DragFAver ViscFAver GradPFAver
CenterMassZ CenterMassR CenterVelX CenterVelY
CenterVelZ DragFX DragFY DragFZ GradPFX
GradPFY GradPFZ BondParticles ParticlesAll D1 D2
D3 D4 D5 D6 D7 D8 poro1 poro2 poro3" +
" CenterMassZ1 CenterMassZ2 CenterMassZ3
CenterMassR1 CenterMassR2 CenterMassR3
CenterMassVelX1 CenterMassVelX2
CenterMassVelX3 CenterMassVelY1
CenterMassVelY2 CenterMassVelY3
CenterMassVelZ1 CenterMassVelZ2
CenterMassVelZ3 DragX1 DragX2 DragX3 DragY1
DragY2 DragY3 DragZ1 DragZ2 DragZ3" +
" GradPX1 GradPX2 GradPX3 GradPY1 GradPY2
GradPY3 GradPZ1 GradPZ2 GradPZ3 \r\n";
for (int t = 0; t < particleProb_time; t++)
{
if (t % forceFrequency == 0)
{
OutputGeneral += "Time " + Convert.ToString(t);
for (int tt = 0; tt <
scalarParamerersOutput.GetLength(1); tt++)
{
if ((tt != 9) && (tt != 13) && (tt != 8))
{
if (tt == 2) OutputGeneral += " " + Convert.ToString(1
- scalarParamerersOutput[t, tt]);
else OutputGeneral += " " +
Convert.ToString(scalarParamerersOutput[t, tt]);
}
if (tt == 8)
```

```
{
OutputGeneral += " " + Convert.ToString(scalarParamerersOutput[t, 13]);
}
}
OutputGeneral += " " + Convert.ToString(particlesNumberInCoorBond[t]) + " " + Convert.ToString(particlesNumberIn[t]);

for (int tt = 0; tt < 8; tt++)
OutputGeneral += " " + Convert.ToString(delatedPSDProbTime[t, tt]);

for (int tt = 0; tt < nubSubZ; tt++)
OutputGeneral += " " + Convert.ToString(1 - (particlesVolumeInZ[t, tt] / sampleVolumeIn[tt]));


for (int yy = 0; yy < scalarParamerersOutputVolumeInZ.GetLength(2); yy++)
{
if ((yy != 2) && (yy != 1) && (yy != 6))
{
for (int tt = 0; tt < nubSubZ; tt++)
OutputGeneral += " " + Convert.ToString(scalarParamerersOutputVolumeInZ[t, tt, yy]);
}
if (yy == 1)
{
for (int tt = 0; tt < nubSubZ; tt++)
OutputGeneral += " " + Convert.ToString(scalarParamerersOutputVolumeInZ[t, tt, 6]);
}
}

OutputGeneral += "\r\n";
}
}

using (FileStream fstream = new FileStream(FileNamePrintAll, FileMode.OpenOrCreate))
{
StreamWriter w = new StreamWriter(fstream, System.Text.Encoding.Default);
w.WriteLine(OutputGeneral);
w.Close();

}
}
bool flagScalardataPart = true;
//добавить center of mass Z,R,Vx,VyVZ,Dx,Dy,Dz,ViscX,ViscY,ViscZ,GradX, GradY,GradZ
if (flagScalardataPart)
{
int nubSubXY = (int)(OuterRadius / PerfRadius / numPerR);


double[,,] scalarParamerersOutput = new double[particleProb_time, nubSubXY, 9]; //coord number(0), bond_number(1),porosity(2),vel magnitude(3),drag(4),visc(5),gradP(6), center of massZ(7),center of massR(8)
double[,] particlesVolumeIn = new double[particleProb_time, nubSubXY];// volume of particles inside the cylinder
double[,] particlesNumberIn = new double[particleProb_time, nubSubXY];
double[,] particlesNumberInCoorBond = new double[particleProb_time, nubSubXY];
double[,] numberPSDProbTime = new double[particleProb_time, PSD.Length];
double[] numberPSDzeroTime = new double[PSD.Length];
double[,] delatedPSDProbTime = new double[particleProb_time, PSD.Length];
double[] sampleVolumeIn = new double[nubSubXY];

double[,,] scalarParametersOutputAdd = new double[particleProb_time, nubSubXY, 16]; //mass x,y,z,R, Vx, Vy,VZ, Dx, Dy, Dz, ViscX, ViscY, ViscZ, GradX, GradY, GradZ

for (int kk = 0; kk < nubSubXY; kk++)
{
sampleVolumeIn[kk] = Math.PI * ((kk + 1) * (kk + 1) - kk * kk) * Z_max * (PerfRadius * numPerR) * (PerfRadius * numPerR);
}

string OutputGeneral = string.Empty;

for (int i = 0; i < particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0] / (double)probTimeDiscrete) - 1;
if (l == 0)
for (int jj = 0; jj < PSD.Length; jj++)
if (particleDataFromVtkAccurate[i, 9] == PSD[jj])
{
numberPSDzeroTime[jj] += 1;
break;
}

if (particleDataFromVtkAccurate[i, 3] - particleDataFromVtkAccurate[i, 9] <= Z_max)
{
double rCurrent = Math.Sqrt(particleDataFromVtkAccurate[i, 1] * particleDataFromVtkAccurate[i, 1] + particleDataFromVtkAccurate[i, 2] * particleDataFromVtkAccurate[i, 2]);
```

```
int rIndex = 0;
for (int gg = 0; gg < nubSubXY; gg++)
{
if ((numPerR * PerfRadius * gg <= rCurrent) &&
(numPerR * PerfRadius * (gg + 1) > rCurrent))
{
rIndex = gg;
particlesVolumeIn[l, rIndex] += (4.0 / 3.0) * Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3);
particlesNumberIn[l, rIndex] += 1;

break;
}
}

for (int ii = 0; ii < PSD.Length; ii++)
if (particleDataFromVtkAccurate[i, 9] == PSD[ii])
{
numberPSDProbTime[l, ii] += 1;
break;
}
//if (particleDataFromVtkAccurate[i, 11] >= 1)
particlesNumberInCoorBond[l, rIndex] += 1;

}
}

for (int i = 0; i <
particleDataFromVtkAccurate.GetLength(0); i++)
{
int l = (int)((double)particleDataFromVtkAccurate[i, 0]
/ (double)probTimeDiscrete) - 1;


if ((l % forceFrequency == 0))
{
if (particleDataFromVtkAccurate[i, 3] -
particleDataFromVtkAccurate[i, 9] <= Z_max)
{
double rCurrent =
Math.Sqrt(particleDataFromVtkAccurate[i, 1] *
particleDataFromVtkAccurate[i, 1] +
particleDataFromVtkAccurate[i, 2] *
particleDataFromVtkAccurate[i, 2]);
int rIndex = -1;
for (int gg = 0; gg < nubSubXY; gg++)
{
if ((numPerR * PerfRadius * gg <= rCurrent) &&
(numPerR * PerfRadius * (gg + 1) > rCurrent))
{
rIndex = gg;
scalarParametersOutputAdd[l, rIndex, 0] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 1]) /
particlesVolumeIn[l, rIndex];//CenterOfMassX
scalarParametersOutputAdd[l, rIndex, 1] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 2]) /
particlesVolumeIn[l, rIndex];//CenterOfMassY

scalarParametersOutputAdd[l, rIndex, 2] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 3]) /
particlesVolumeIn[l, rIndex];//CenterOfMassZ
  //scalarParametersOutputAdd[l, rIndex, 3] +=
Math.Sqrt(scalarParametersOutputAdd[l, rIndex, 0] *
scalarParametersOutputAdd[l, rIndex, 0] +
scalarParametersOutputAdd[l, rIndex, 1] *
scalarParametersOutputAdd[l, rIndex, 1])/
particlesVolumeIn[l, rIndex];

scalarParametersOutputAdd[l, rIndex, 4] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 4]) /
particlesVolumeIn[l, rIndex];//CenterOfMassVelX
scalarParametersOutputAdd[l, rIndex, 5] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 5]) /
particlesVolumeIn[l, rIndex];//CenterOfMassVelY
scalarParametersOutputAdd[l, rIndex, 6] += ((4.0 / 3.0)
* Math.PI *
Math.Pow(particleDataFromVtkAccurate[i, 9], 3) *
particleDataFromVtkAccurate[i, 6]) /
particlesVolumeIn[l, rIndex];//CenterOfMassZVelZ

scalarParametersOutputAdd[l, rIndex, 7] +=
(DragForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberIn[l, rIndex]; ;//Center of massDragX
scalarParametersOutputAdd[l, rIndex, 8] +=
(DragForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberIn[l, rIndex]; ;//Center of massDragY
scalarParametersOutputAdd[l, rIndex, 9] +=
(DragForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberIn[l, rIndex]; ;//Center of massDragZ

scalarParametersOutputAdd[l, rIndex, 10] +=
(GradientPForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberIn[l, rIndex];//Center of massGradX
scalarParametersOutputAdd[l, rIndex, 11] +=
(GradientPForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberIn[l, rIndex];//Center of massGradPY
scalarParametersOutputAdd[l, rIndex, 12] +=
(GradientPForceCoef[indexVTKtoDrag[i], 6]) /
particlesNumberIn[l, rIndex];//Center of massGradPZ

scalarParametersOutputAdd[l, rIndex, 13] +=
(ViscousForceCoef[indexVTKtoDrag[i], 4]) /
particlesNumberIn[l, rIndex];//Center of massViscY
scalarParametersOutputAdd[l, rIndex, 14] +=
(ViscousForceCoef[indexVTKtoDrag[i], 5]) /
particlesNumberIn[l, rIndex];//Center of massViscZ
```

```
scalarParametersOutputAdd[l, rIndex, 15] += (ViscousForceCoef[indexVTKtoDrag[i], 6]) / particlesNumberIn[l, rIndex];//Center of massViscZ

break;
}
}
if (rIndex != -1)
{
//if (particleDataFromVtkAccurate[i, 11] >= 1)
scalarParamerersOutput[l, rIndex, 1] += particleDataFromVtkAccurate[i, 11] / particlesNumberInCoorBond[l, rIndex];//bond number

scalarParamerersOutput[l, rIndex, 0] += particleDataFromVtkAccurate[i, 10] / particlesNumberIn[l, rIndex]; //coord number
scalarParamerersOutput[l, rIndex, 3] += particleDataFromVtkAccurate[i, 7] / particlesNumberIn[l, rIndex];//vel mag
scalarParamerersOutput[l, rIndex, 4] += DragForceCoef[indexVTKtoDrag[i], 13] / particlesNumberIn[l, rIndex];//drag
scalarParamerersOutput[l, rIndex, 5] += ViscousForceCoef[indexVTKtoDrag[i], 8] / particlesNumberIn[l, rIndex];//visc
scalarParamerersOutput[l, rIndex, 6] += GradientPForceCoef[indexVTKtoDrag[i], 9] / particlesNumberIn[l, rIndex];//gradP
scalarParamerersOutput[l, rIndex, 2] += (4.0 / 3.0) * Math.PI * Math.Pow(particleDataFromVtkAccurate[i, 9], 3) / sampleVolumeIn[rIndex];
scalarParamerersOutput[l, rIndex, 7] += ((4.0 / 3.0) * Math.PI * Math.Pow(particleDataFromVtkAccurate[i, 9], 3)
* particleDataFromVtkAccurate[i, 3]) / particlesVolumeIn[l, rIndex];//CenterOfMassZ
scalarParamerersOutput[l, rIndex, 8] += ((4.0 / 3.0) * Math.PI * Math.Pow(particleDataFromVtkAccurate[i, 9], 3)
* Math.Sqrt(particleDataFromVtkAccurate[i, 2] * particleDataFromVtkAccurate[i, 2] + particleDataFromVtkAccurate[i, 1] * particleDataFromVtkAccurate[i, 1])) / particlesVolumeIn[l, rIndex];//CenterOfMassZ//Center of massR
}
}
}
}

for (int t = 0; t < particleProb_time; t++)
{

for (int tt = 0; tt < nubSubXY; tt++)
scalarParametersOutputAdd[t, tt, 3] += Math.Sqrt(scalarParametersOutputAdd[t, tt, 0] * scalarParametersOutputAdd[t, tt, 0] + scalarParametersOutputAdd[t, tt, 1] * scalarParametersOutputAdd[t, tt, 1]);
}


for (int t = 0; t < particleProb_time; t++)
for (int tt = 0; tt < PSD.Length; tt++)
delatedPSDProbTime[t, tt] = numberPSDzeroTime[tt] - numberPSDProbTime[t, tt]; //deleated particles PSD
 //to write data
OutputGeneral = "Time number ";
string[] outputGeneral = new string[] { "coord_all", "bond", "poro", "VelMagAver", "DragFAver", "ViscFAver", "GradPFAver", "CenterMassZ", "BondParticles", "ParticlesAll", "CenterOfMassZ", "CenterOfMassR", "CenterVelX", "CenterVelY", "CenterVelZ", "DragFX", "DragFY", "DragFZ", "GradPFX", "GradPFY", "GradPFZ", "ViscFX", "ViscFY", "ViscFZ" };

for (int oo = 0; oo < outputGeneral.Length; oo++)
for (int r = 0; r < nubSubXY; r++)
OutputGeneral += " " + outputGeneral[oo] + Convert.ToString(r);

OutputGeneral += "\r\n";


for (int t = 0; t < particleProb_time; t++)
{
if (t % forceFrequency == 0)
{
OutputGeneral += "Time " + Convert.ToString(t);
for (int tt = 0; tt < scalarParamerersOutput.GetLength(2) - 1; tt++)

for (int r = 0; r < nubSubXY; r++)
{
if (tt == 2) OutputGeneral += " " + Convert.ToString(1 - scalarParamerersOutput[t, r, tt]);
if ((tt != 2) && (tt != 8)) OutputGeneral += " " + Convert.ToString(scalarParamerersOutput[t, r, tt]);
}

for (int r = 0; r < nubSubXY; r++)
{
OutputGeneral += " " + Convert.ToString(particlesNumberInCoorBond[t, r]);
}
for (int r = 0; r < nubSubXY; r++)
{
OutputGeneral += " " + Convert.ToString(particlesNumberIn[t, r]);
}

for (int tt = 2; tt < scalarParametersOutputAdd.GetLength(2); tt++)
for (int r = 0; r < nubSubXY; r++)
```

```
{
OutputGeneral += " " + Convert.ToString(scalarParametersOutputAdd[t, r, tt]);
}

OutputGeneral += "\r\n";
}
}

using (FileStream fstream = new FileStream(FileNamePrintPart, FileMode.OpenOrCreate))
{
StreamWriter w = new StreamWriter(fstream, System.Text.Encoding.Default);
w.WriteLine(OutputGeneral);
w.Close();

}
}

//writing to the end of VTK
bool flag_print_in_vtk = true;
if (flag_print_in_vtk && flag_manipulation && flag_reading_vtk && flag_particleProbes)
{
sWatch.Start();

int[] currentIndexDrafForce = new int[particleProb_time];
currentIndexDrafForce[0] = 0;
for (int i = 1; i < numberParticlesTime.Length; i++)
{
currentIndexDrafForce[i] = currentIndexDrafForce[i - 1] + numberParticlesTime[i];
//Console.WriteLine($"The begin index of probeTime print data : {currentIndexDrafForce[i]}");
}
//omega 3 100000 double
//time, position(3 values), velocities(3 values),velocity magnitude, velocity_quartile
//double[,] particleDataFromVtk = new double[particle_number * particleProb_time, 9];//time, position(3 values), velocities(3 values), velocity magnitude, velocity

bool flagVeloc = false;
bool flagVelNorm = true;
bool flagDrag = true;
bool flagVisc = true;
bool flagGradP = true;
string velData, velNormData, dragData, viscData, gradPData;
velData = velNormData = dragData = viscData = gradPData = string.Empty;

bool[] printFlag = new bool[5] { flagVeloc, flagVelNorm, flagDrag, flagVisc, flagGradP };

int p = 0;
for (int r = 0; r < printFlag.GetLength(0); r++)
if (printFlag[r] == true) p++;

for (int l = 0; l < particleProb_time_2; l++)
{
if ((l % (forceFrequency * 2) == 0) && (l > -1))//(l % forceFrequency == 0
{

velData = velNormData = dragData = viscData = gradPData = string.Empty;
int current_timeStep = start_timestep + (l + 1) * frequence_of_prob * couplingInterval;
string FileNameLong1 = FileName_z2 + Convert.ToString(current_timeStep) + Convert.ToString(".vtk");
string FileNameLongWrite1 = FileNamePrintVTK + Convert.ToString("_test_") + Convert.ToString(current_timeStep) + Convert.ToString(".vtk");
int count2 = 0;
double relative_time = (double)((current_timeStep - start_timestep) * DEM_time_discrete);


StreamReader file_2 = new StreamReader(FileNameLong1);
while ((line = file_2.ReadLine()) != null)
{
string[] split = line.Split(' ');
count2++;
}

int line_number2 = 0;
using (StreamReader file3 = new StreamReader(FileNameLong1))
{
using (StreamWriter newOne = new StreamWriter(FileNameLongWrite1))
{
while ((line = file3.ReadLine()) != null)
{
string[] split = line.Split(' ');
if (Convert.ToString(split[0]) == "FIELD")
{
line = Convert.ToString("FIELD FieldData ") + Convert.ToString(10 + p) + Convert.ToString("\r\n");
//10+ parameters---very important
}
line_number2++;
if (line_number2 > count2)
continue;
newOne.WriteLine(line);
}
}
}
```

```
if (flagVelNorm)
{
int count3 = 0;
velNormData = "VelocityNorm" + " 1 " + Convert.ToString(numberParticlesTime[l]) + " double\n";
for (int i = 0; i < particleDataFromVtk.GetLength(0); i++)
{
if (particleDataFromVtk[i, 0] == relative_time)
{
if (count3 != numberParticlesTime[l] - 1)
{
velNormData += Convert.ToString(particleDataFromVtkAccurate[i, 8]) + " ";//velocity quartile
}
if (count3 == numberParticlesTime[l] - 1)
{
velNormData += particleDataFromVtkAccurate[i, 8].ToString("");
}
if ((count3 + 1) % (9) == 0 && (count3 >= 8))
{
velNormData += "\r\n";
}
if (count3 >= numberParticlesTime[l]) break;
count3++;
}

}
}
if (flagVeloc)
{
int count3 = 0;
velData = "Velocity" + " 1 " + Convert.ToString(numberParticlesTime[l]) + " double\n";
for (int i = 0; i < particleDataFromVtk.GetLength(0); i++)
{
if (particleDataFromVtk[i, 0] == relative_time)
{

if (count3 != numberParticlesTime[l] - 1)
{
velData += Convert.ToString(particleDataFromVtk[i, 7]) + " ";//velocity quartile
}
if (count3 == numberParticlesTime[l] - 1)
{
velData += particleDataFromVtk[i, 7].ToString("");
}
if ((count3 + 1) % (9) == 0 && (count3 >= 8))
{
velData += "\r\n";
}
if (count3 >= numberParticlesTime[l]) break;
count3++;
}
}
}
if (flagDrag) ////time, position(3 values), dragForce(3 values),
{
int count3 = 0;
dragData = "DragForce" + " 3 " + Convert.ToString(numberParticlesTime[l]) + " double \n";
int begin_index = currentIndexDrafForce[l];
int end_index = begin_index + numberParticlesTime[l] - 1;
for (int i = begin_index; i <= end_index; i++)
{
if (count3 != 3 * numberParticlesTime[l] - 1)
{
dragData += Convert.ToString(DragForceCoef[indexVTKtoDrag[i], 4]) + " ";//velocity quartile
dragData += Convert.ToString(DragForceCoef[indexVTKtoDrag[i], 5]) + " ";//velocity quartile
dragData += Convert.ToString(DragForceCoef[indexVTKtoDrag[i], 6]) + " ";//velocity quartile
count3 = count3 + 3;
}
if (count3 == 3 * numberParticlesTime[l] - 1)
{
dragData += DragForceCoef[indexVTKtoDrag[i], 6].ToString("");
}
if ((count3) % (9) == 0 && (count3 >= 8))
{
dragData += "\r\n";
}
if (count3 >= 3 * numberParticlesTime[l]) break;

}
}
if (flagVisc)
{
int count3 = 0;
viscData = "ViscForce" + " 3 " + Convert.ToString(numberParticlesTime[l]) + " double \n";
int begin_index = currentIndexDrafForce[l];
int end_index = begin_index + numberParticlesTime[l] - 1;
for (int i = begin_index; i <= end_index; i++)
{
if (count3 != 3 * numberParticlesTime[l] - 1)
{
viscData += Convert.ToString(ViscousForceCoef[indexVTKtoDrag[i], 4]) + " ";//velocity quartile
```

```
viscData += Convert.ToString(ViscousForceCoef[indexVTKtoDrag[i], 5]) + " ";//velocity quartile
viscData += Convert.ToString(ViscousForceCoef[indexVTKtoDrag[i], 6]) + " ";//velocity quartile
count3 = count3 + 3;
}
if (count3 == 3 * numberParticlesTime[l] - 1)
{
viscData += ViscousForceCoef[indexVTKtoDrag[i], 6].ToString("");
}
if ((count3) % (9) == 0 && (count3 >= 8))
{
viscData += "\r\n";
}
if (count3 >= 3 * numberParticlesTime[l]) break;

}
}
if (flagGradP)
{
int count3 = 0;
gradPData = "GradPForce" + " 3 " + Convert.ToString(numberParticlesTime[l]) + " double \n";
int begin_index = currentIndexDrafForce[l];
int end_index = begin_index + numberParticlesTime[l] - 1;
for (int i = begin_index; i <= end_index; i++)
{
if (count3 != 3 * numberParticlesTime[l] - 1)
{
gradPData += Convert.ToString(GradientPForceCoef[indexVTKtoDrag[i], 9]) + " ";//velocity quartile
gradPData += Convert.ToString(GradientPForceCoef[indexVTKtoDrag[i], 9]) + " ";//velocity quartile
gradPData += Convert.ToString(GradientPForceCoef[indexVTKtoDrag[i], 9]) + " ";//velocity quartile
count3 = count3 + 3;
}
if (count3 == 3 * numberParticlesTime[l] - 1)
{
gradPData += GradientPForceCoef[indexVTKtoDrag[i], 9].ToString("");
}
if ((count3) % (9) == 0 && (count3 >= 8))
{
gradPData += "\r\n";
}
if (count3 >= 3 * numberParticlesTime[l]) break;

}
}

string[] printParam = new string[5] { velData, velNormData, dragData, viscData, gradPData };
for (int rr = 0; rr < printParam.Length; rr++)
{
string tem;
if (printFlag[rr])
{
tem = printParam[rr];
StreamWriter newOne2 = new StreamWriter(FileNameLongWrite1, true, System.Text.Encoding.Default);
newOne2.WriteLine(tem);
newOne2.Close();
}
}
//-----------------------------------------------------------------------------------------------------------------------
Console.WriteLine("File of print number " + l);
}
}

sWatch.Stop();

TimeSpan tSpan = sWatch.Elapsed;
// Format and display the TimeSpan value.
string elapsedTime = String.Format("{0:00}:{1:00}:{2:00}.{3:00}", tSpan.Hours, tSpan.Minutes, tSpan.Seconds, tSpan.Milliseconds / 10);
Console.WriteLine("Copy and print in file " + elapsedTime);


}
else { Console.WriteLine("You must read and calculate all necessary data!"); }

Console.WriteLine("Everething is finished!");
Console.ReadKey();
}
}
}
```