Algebraic Multiscale Framework for Fractured Reservoir Simulation

Tene, Matei

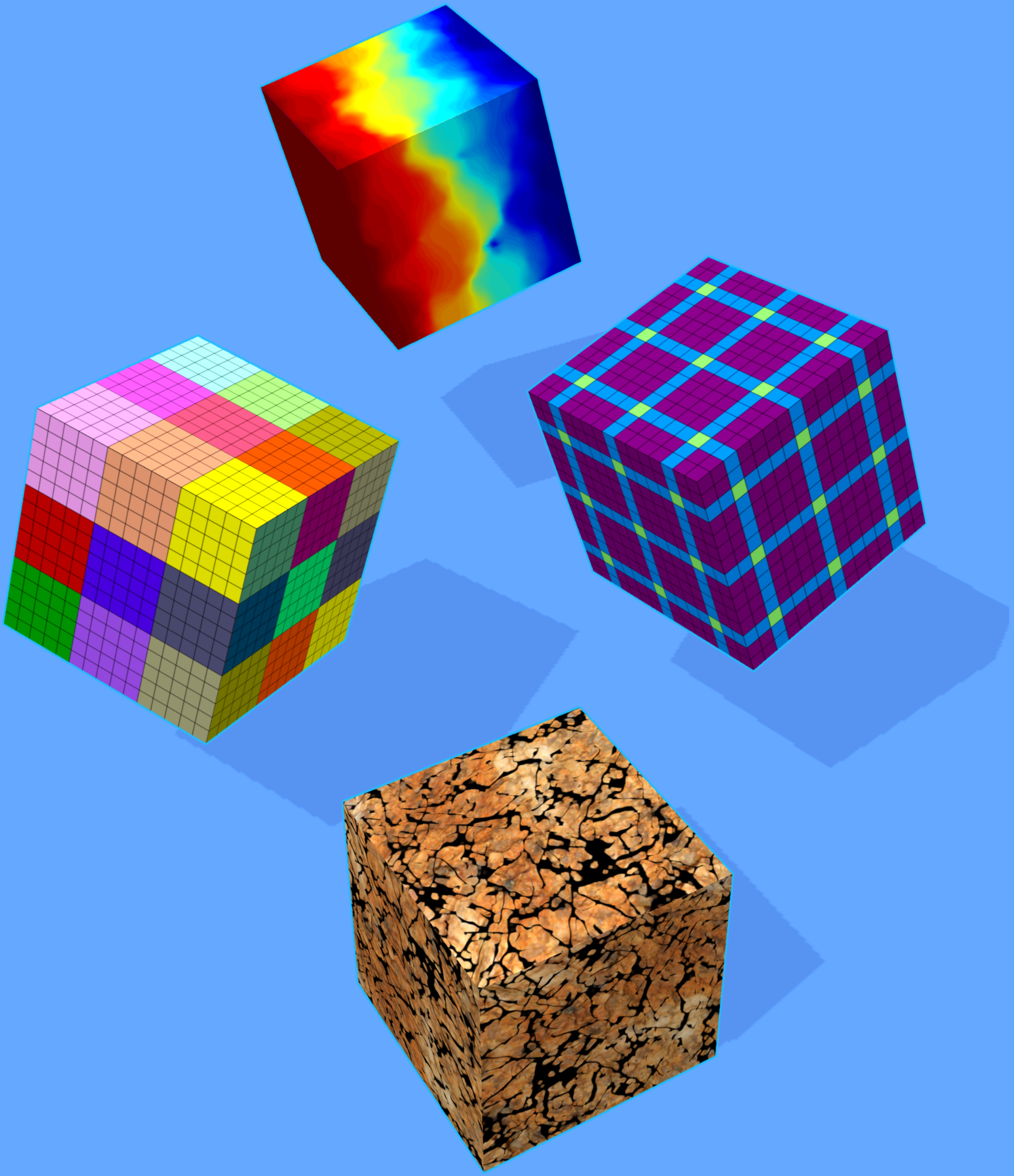
DOI
10.4233/uuid:0ce45ddb-4932-4808-aa92-bd13deb85fa9

Publication date
2018

Document Version
Final published version

Citation (APA)
Tene, M. (2018). *Algebraic Multiscale Framework for Fractured Reservoir Simulation*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:0ce45ddb-4932-4808-aa92-bd13deb85fa9

Algebraic Multiscale Framework for Fractured Reservoir Simulation



Matei Țene

Propositions

accompanying the dissertation

Algebraic Multiscale Framework for Fractured Reservoir Simulation

by

Matei Ţene

1. When computing heterogeneity-aware basis functions for Algebraic MultiScale (AMS) methods, the simplest formulation is always the best choice. (Chapters 2 and 3)
2. The coarsening factor dictates the computational performance of AMS, when used as a fully converged iterative solver. (Chapters 2, 3 and 4)
3. A Two-Point Flux Approximation (TPFA) stencil is not sufficient to represent flow in the presence of impermeable sub-grid features, embedded in the matrix rock. (Chapter 4)
4. A sequentially-coupled approach between equations governing distinct physical processes, is the best option for commercial reservoir simulation. (Appendix D)
5. Accounting for modelling errors is crucial when performing history matching and optimization in a Closed Loop Reservoir Management (CLRM) system.
6. Any attempt to perform model validation, based on experimental core-flood data, has to operate on a 3D heterogeneous domain.
7. Participation in an internship or academic exchange must be included in the core of the doctoral education program for PhD candidates at the TU Delft.
8. The recipe for success in research is the same as in dancing: 45% persistent hard work, 10% spontaneous creativity, 45% good communication with your partner.
9. The main difference between a veteran programmer and a beginner is that the latter still has the impulse to blame the compiler for errors in their code.
10. If the people of Eastern and Western Europe would swap places overnight, the cultural shock would be less than if the North were to switch with the South.

These propositions are regarded as opposable and defendable, and have been approved as such by the promotors, prof. dr. ir. J.D. Jansen and dr. H. Hajibeygi.

Stellingen

behorende bij het proefschrift

Algebraic Multiscale Framework for Fractured Reservoir Simulation

door

Matei Țene

1. Bij het berekenen van heterogeniteitsbewuste basisfuncties voor Algebraïsche MultiSchaal (AMS) methoden, is de eenvoudigste formulering altijd de beste keuze. (Hoofdstukken 2 en 3)
2. De coarsening-factor bepaalt de computationele prestaties van AMS, waneer het als volledig geconvergeerde iteratieve oplosser gebruikt is. (Hoofdstukken 2, 3 en 4)
3. Een tweepuntenfluxbenadering (TPFA) sjabloon is niet voldoende om de stroming in de aanwezigheid van ondoordringbare sub-raster structuren, ingebed in de matrixgesteente, weer te geven. (Hoofdstuk 4)
4. De sequentieel gekoppelde benadering tussen vergelijkingen die verschillende fysieke processen besturen, is de beste optie voor commerciële reservoirsimulatie. (Bijlage D)
5. Het overleg van modelleringsfouten is cruciaal bij het uitvoeren van geschiedenisvergelijking en optimalisatie in een Closed Loop Reservoir Management (CLRM) systeem.
6. Elke poging tot modelvalidatie, gebaseerd op experimentele kernstroomgegevens, moet gebruik maken van een heterogeen 3D-domein.
7. Deelname aan een stage of academische uitwisseling moet in de kern van de doctoraatsopleiding voor alle promovendi aan de TU Delft ingesloten zijn.
8. Het recept voor succes in onderzoek is hetzelfde als dat in dansen: 45% hardnekkig hard werken, 10% spontane creativiteit, 45% goede communicatie met je partner.
9. Het belangrijkste verschil tussen een ervaren programmeur en een beginner is dat de laatste nog steeds de neiging heeft om de compiler de schuld te geven van fouten in hun code.
10. Als de mensen in Oost- en West-Europa van plaats zouden ruilen, zou de culturele schok minder zijn dan die tussen het Noorden en het Zuiden.

Deze stellingen worden opponeerbaar en verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotors, prof. dr. ir. J.D. Jansen en dr. H. Hajibeygi.

Algebraic Multiscale Framework for Fractured Reservoir Simulation

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
dinsdag, 11 september 2018 om 12:30 uur

door

Matei Țene

Master of Science in Applied Mathematics,
Technische Universiteit Delft, Nederland,
geboren te Boekarest, Roemenië.

Dit proefschrift is goedgekeurd door de promotoren.

Samenstelling promotiecommissie bestaat uit:

Rector Magnificus	voorzitter
Prof. dr. ir. J.D. Jansen	TU Delft, promotor
Dr. H. Hajibeygi	TU Delft, copromotor
Onafhankelijke leden:	
Prof. dr. S. Geiger	Heriot-Watt University
Prof. dr. ir. L.J. Sluys	TU Delft
Prof. dr. H.A. Tchelepi	Stanford University
Prof. dr. ir. C. Vuik	TU Delft
Prof. dr. W.R. Rossen	TU Delft, reservelid
Overige leden:	
Dr. M.S. Al Kobaisi	Khalifa University, Abu Dhabi

This research was carried out within the Delft Advanced Reservoir Simulation group (DARSim) at TU Delft, in collaboration with Khalifa University and sponsored by the Abu Dhabi National Oil Company (ADNOC).

DARSim
Delft Advanced Reservoir Simulation

جامعـــة خليفـــة
Khalifa University

ادنوك
ADNOC

Keywords:	algebraic multiscale methods, naturally fractured porous media, conductivity contrasts, compressible flow, multiphase transport
Printed by:	Gildeprint
Front & Back:	Matei Ţene

ISBN 978-94-6186-956-2

An electronic version of this dissertation is available at http://repository.tudelft.nl/.

Not all heroes wear capes

Owl City

Pentru Silica, Nicu, Mihai und Alina

Contents

Preface

Dear Reader,

Regardless of how you stumbled upon it – be it on a desk, bookshelf, under a monitor or in PDF from the TU Delft repository – thank you for your interest in my thesis! What follows is an x-ray of the past 4 years, spent along the picturesque canals of Delft or travelling to several conferences in Europe and North America. However, while bluntly factual, no x-ray can paint a complete picture of reality. So here is a glimpse behind-the-scenes, hoping to add a bit of colour to the narrative to come.

It all started at the end of 2013, shortly after the conclusion of my Master's in Applied Math at TU Delft. Dreaming of a positive answer to my job application at Google, while reading about post-graduate positions in Tasmania and the UK, I found out about a PhD opening right across the street. The news came from Prof. Jan-Dirk Jansen, who was part of my MSc. defense committee. The topic – multiscale methods for flow in porous media – was another step outside my comfort zone. Admittedly, not as much as the one I had made 2 years before, when I landed at Amsterdam's Schiphol airport, armed with a Computer Science background to confront my fear of Math.

Soon thereafter, I met my future supervisor, Dr. Hadi Hajibeygi, who put my worries to rest – we were going to get me up-to-speed on reservoir simulation. So, enter the waterfall – for, after having changed 3 accommodations in 4 months, it was time to pack my bags for a visit to Stanford University. There, with the help of Hadi, Yixuan and Prof. Hamdi Tchelepi, I got a crash course on algebraic multiscale methods.

Still, it turned out that multiscale was only one piece of the PhD puzzle. Three conditions were laid out before me, on my road to graduation. First and foremost, journal papers had to be published. Then, given 4 years, somehow, someway, somewhere in the town of Delft or its surroundings, I would have to find myself a girlfriend. Finally, but not to say the least, I would have to win a table tennis match against Hadi. Up to the writing of the present chapter, that last point remains unfulfilled …

Modelling flow through fractures brought additional challenges, in terms of data structures, condition numbers, coarsening. While addressing these, I also had the pleasure to present at ECMOR, the SPE RSS, CMWR, SIAM GS and InterPore. The conference papers paved the way towards my first journal publications. I also embraced the opportunity to teach, as student assistant during Hadi's reservoir simuation course.

Delft is a vibrant community, offering many social activities for students of all ages – from team sports (football, basketball, volleyball) to dance classes and salsa parties. After several years of being a SoSalsa member, I got the ambition to organize my own dance event – a ballroom / latin night at the university Culture Center. I did not expect it, but the party ignited the spirits of two enthusiastic girls – Marieke and Marit – which – together with myself and Lanleigh – were to form the founding board of the Blue Suede Shoes society for dancing students. With a focus on ballroom, latin and Argentinian tango, BSS remains a hub for couple-dancers at TU Delft and flag-carrier for the university during student dancesport competitions (NTDS, ETDS).

Speaking of activities, all fresh PhD candidates at the CiTG faculty in TU Delft are warmly welcomed to join the MV student mining society. They organize a wide range of events, from pub nights in Het Noorden or barbecues in front of PSOR, to company and field visits. I also enjoyed being a member of the SPE and SIAM student chapters housed by the university.

During my nearly 7 years on Dutch soil, I also had a go at learning the language. Gathering what I could recall from my years in elementary school learning German, I enrolled into the "Delftse methode" course taught at the university. And that's how I finally came to terms with words like "gezellig" and "hagelslag". So when the Dutch postal service shortlisted me for a summer job, I was over the moon – too bad my limited post-PhD contract availability couldn't allow it ...

Looking back, I see a lot of happy memories, with a few hiccups and missed deadlines, here and there. And both sides of the coin brought lots to learn – on the scientific side and, equally important, in terms of personal and social skills. So, as the rollercoaster continued, I got to supervise Master students and started to slowly get to grips with effective time management, while doing my best to maintain good communication with Hadi. The PhD program provided an excellent sandbox for me to hone these skills, both through Graduate School courses and hands-on, although, I have to admit, I am still on the learning curve.

Last, but definitely not least, I was fortunate to spend 3 months as an intern at Chevron, in Houston, TX. There, as a member in Dr. Xian-Huan Wen's RPP team and under the guidance of Dr. Seong Lee, I explored the topic of modelling compositional flow using a sequentailly-implicit approach.

Now, standing at the end of the 4 years, I look forward to continue my work on multiscale methods, this time from the perspective of the industry. Wearing the Schlumberger hat, I hope to understand the practical implications of my research and further my knowledge of petroleum engineering. Cheers to that!

Yours faithfully,

Matei Ţene
Delft, June 2018

Summary

Despite welcome increases in the adoption of renewable energy sources, oil and natural gas are likely to remain the main ingredient in the global energy diet for the decades to come. Therefore, the efficient exploitation of existing suburface reserves is essential for the well-being of society. This has stimulated recent developments in computer models able to provide critical insight into the evolution of the flow of water, gas and hydrocarbons through rock pores. Any such endeavour, however, has to tackle a number of challenges, including the considerable size of the domain, the highly heterogeneous spatial distribution of geological properties, as well as the intrinsic uncertainty and limitations associated with field data acquisition. In addition, the naturally-formed or artificially induced networks of fractures, present in the rock, require special treatment, due to their complex geometry and crucial impact on fluid flow patterns.

From a numerical point of view, a reservoir simulator's operation entails the solution of a series linear systems, as dictated by the spatial and temporal discretization of the governing equations. The difficulty lies in the properties of these systems, which are large, ill-conditioned and often have an irregular sparsity pattern. Therefore, a brute-force approach, where the solutions are directly computed at the original fine-scale resolution, is often an impractically expensive venture, despite recent advances in parallel computing hardware. On the other hand, switching to a coarser resolution to obtain faster results, runs the risk of omitting important features of the flow, which is especially true in the case of fractured porous media.

This thesis describes an algebraic multiscale approach for fractured reservoir simulation. Its purpose is to offer a middle-ground, by delivering results at the original resolution, while solving the equations on the coarse-scale. This is made possible by the so-called basis functions – a set of locally-supported cross-scale interpolators, conforming to the heterogeneities in the domain. The novelty of the work lies in the extension of these methods to capture the effect of fractures. Importantly, this is done in fully algebraic fashion, i.e. without making any assumptions regarding geometry or conductivity properties.

In order to elicit the generality of the proposed approach, a series of sensitivity studies are conducted on a proof-of-concept implementation. The results, which include both CPU times and convergence behaviour, are discussed and compared to those obtained using an industrial-grade AMG package. They serve as benchmarks, recommending the inclusion of multiscale methods in next-generation commercial reservoir simulators.

Samenvatting

Ondanks de welkome toename van hernieuwbare energiebronnen, blijven olie en aardgas de komende decennia waarschijnlijk het belangrijkste ingrediënt in het wereldwijde energiedieet. Daarom is de efficiënte exploitatie van bestaande reserves aan de oppervlakte essentieel voor het welzijn van de samenleving. Dit heeft recente ontwikkelingen in computermodellen gestimuleerd die kritisch inzicht kunnen verschaffen in de evolutie van de stroom van water, gas en koolwaterstoffen door de poriën van de rotsen. Een dergelijke inspanning moet echter een aantal uitdagingen aanpakken, waaronder de aanzienlijke omvang van het domein, de zeer heterogene ruimtelijke verdeling van geologische eigenschappen, evenals de intrinsieke onzekerheid en beperkingen specifiek voor de verzameling van veldgegevens. Bovendien vereisen de natuurlijk gevormde of kunstmatig geïnduceerde netwerken van fracturen, aanwezig in de rots, een speciale behandeling vanwege hun complexe geometrie en cruciale impact op stromingspatronen van de vloeistof.

Vanuit een numeriek oogpunt omvat de werking van een reservoirsimulator de oplossing van een serie-lineaire systemen, zoals gedicteerd door de ruimtelijke en temporele discretisatie van de bestaande vergelijkingen. De moeilijkheid ligt in de eigenschappen van deze systemen, die groot, slecht geconditioneerd zijn en vaak een onregelmatig schaarsheidspatroon hebben. Daarom is een brute-force-benadering, waarbij de oplossingen rechtstreeks worden berekend met de oorspronkelijke fijnafstemming, vaak een onpraktisch dure onderneming, ondanks recente ontwikkelingen in parallelle computersystemen. Aan de andere kant loopt het overschakelen naar een grovere resolutie om snellere resultaten te verkrijgen, het risico van het weglaten van belangrijke kenmerken van de stroom, en dit is bijzonder waar in het geval van gefractureerde poreuze media.

Dit proefschrift beschrijft een algebraïsche multischaal benadering voor reservoirsimulatie. Het doel is om een middenweg aan te bieden, door resultaten te leveren met de oorspronkelijke resolutie, terwijl de vergelijkingen op de ruwe schaal worden opgelost. Dit wordt mogelijk gemaakt door de zogenaamde basisfuncties - een set lokaal ondersteunde cross-schaal interpolatoren, bewust van de heterogeniteiten in het domein. De nieuwigheid van het werk ligt in de uitbreiding van deze methoden om het effect van fracturen vast te leggen. Belangrijk is dat dit op volledig algebraïsche wijze wordt gedaan, zonder aannames te doen met betrekking tot geometrie of geleidbaarheidseigenschappen.

Om de algemeenheid van de voorgestelde aanpak te ontlokken, wordt een reeks gevoeligheidsstudies uitgevoerd met betrekking tot een proof-of-concept-implementatie. De resultaten, die zowel CPU-tijden als convergentiegedrag omvatten, worden besproken en vergeleken met die verkregen met een industrieel AMG-pakket. Ze dienen als benchmarks en bevelen de opname van multischaal methoden in commerciële reservoirsimulators van de volgende generatie aan.

1

Introduction

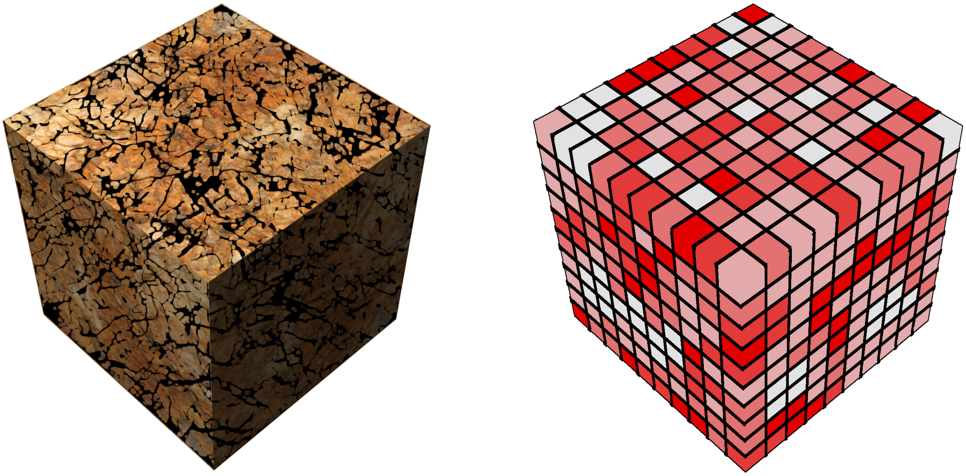
The demand for energy across the world is increasing every year (+1% over the course of 2017) with an appetite that is shifting away from the use of coal, towards sources with a lower carbon footprint [1]. The resulting gap in the global energy diet is compensated, to a large extent, by an increase in hydrocarbon extraction (i.e. oil and natural gas). Traditional water flooding methods applied to conventional petroleum reservoirs (sandstone or carbonate rocks) typically achieve recovery factors between 20 - 40% of the original oil in-place [2]. Therefore, Enhanced Oil Recovery (EOR) methods that use, e.g. miscible gas, chemicals (surfactants, solvents, polymers, nano-particles, etc.) or thermal effects (steam, in-situ combustion) to target the remaining 60 - 80% of reserves constitute an active field of research and form the subject of pilot studies [3]. At the same time, the oil and gas industry reports successful developments in unconventional reservoirs, such as shale formations, by means of hydraulic fracturing and stimulation [4].

The host rock remains of interest even after the hydrocarbon content is depleted. Its pore space can serve as storage for greenhouse gases, such as CO_2, which appear as disposable by-products in industrial operations [5–7]. Alternatively, it can be used to hold the hydrogen resulting from the hydrolysis of water [8–10]. Then, due to the reversibility of the chemical reaction, the reservoir effectively acts as a capacitor, dampening the peaks and troughs created by the fluctuations of supply and demand across the power grid.

Recent years have also brought considerable advances in the exploitation of renewable sources of energy [1]. Among them, geothermal energy projects have seen increased development, targeting geological formations with steep thermal gradients [11–13]. Their competitive advantage, when compared to hydro, wind, solar and nuclear power, is the continuous heat flux, which can be stopped and restarted without major delay, as demand dictates.

Zooming out and away from the energy sector, the subsurface is also home to important ecological processes, such as groundwater flow [14]. Alongside its crucial role in the natural water cycle, the understanding of contaminant propagation through groundwater is necessary to facilitate remediation efforts [15].

The success of any of the subsurface projects described above, relies on a good understanding of the physics of fluid flow through porous media. This has sparked significant interest, both within the industry and the scientific community, for the development of *reservoir simulators*. They operate by solving the governing equations, in order to model the movement of the target fluids and their interaction with the host underground rock, over time. This is made possible by the consideration of a discrete representation of reality, described by a field of flow-relevant properties (e.g. storage volume, conductivity, etc.) defined at specific locations, according to a spatial grid (see Fig. 1.1). Then, after prescribing appropriate boundary conditions, the quantities of interest (e.g. pore pressure, fluid saturation, rock displacement, etc.) can be computed in each grid cell, over a series of consecutive time-steps, up to (finite) *machine precision*, ϵ_m.



(a) Reservoir

(b) Simulation grid – the colour intensity is proportional to the cell permeability

Figure 1.1: Reservoir simulators construct a discrete representation of reality, composed of a set of properties defined on a spatial grid. The evolution of the flow can then be simulated over a series of consecutive time-steps.

The forecasting power of reservoir simulation results is crucial for effective decision-making, operational safety, as well as the assessment and minimization of any environmental side-effects during the project. In order to ensure it, however, one must overcome a number of technical challenges.

1.1. Challenges in fractured reservoir simulation

The large physical size of the target subsurface formations (kilometers) and the rapidly-varying flow conductivities between different locations (mili- or centimeter scale) pose serious difficulties for traditional and next-generation reservoir simulators. These heterogeneous properties appear, on the one hand, due to the composition and spatial distribution of the solid grains that form the *matrix rock* (i.e. porosity, permeability, wettability, etc., see [16] and Fig. 1.4(a)) and, on the other hand, the (natural or induced) networks of interconnected *fractures* contained within (Fig. 1.2).

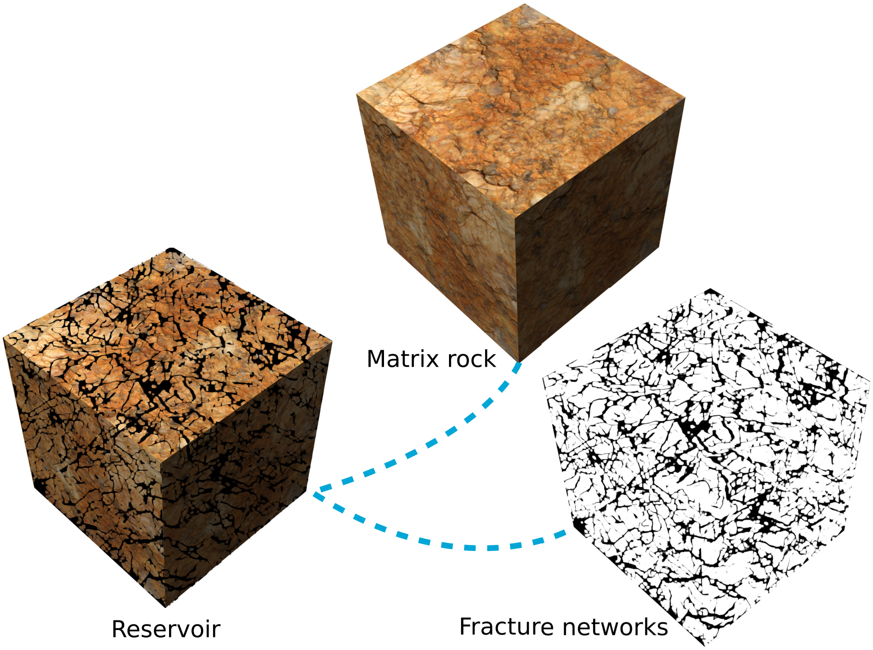


Figure 1.2: Two main sources of heterogeneity in a fractured reservoir.

Rock properties are represented as effective parameters, with values defined in each grid cell of the reservoir model (Fig. 1.1(b)). Their resolution is bounded, on the one end, by the *Representative Elementary Volume* (REV, see e.g. [17]) and, on the other, by the granularity and tolerances of the field measurements, as well as the limited computational resources available for the simulation task. Meaningful results must account for the intrinsic spatial correlations of rock porosity and permeability, dictated by the geological structure (channels, layers, etc.), alongside the evolution of the consolidation processes involved in the formation of the reservoir. The resulting fine-scale heterogeneities translate into variations in the flow properties of the rock (storage capacity and conductivity), spanning several orders of magnitude. Consequently, the linear systems arising from the discretization of the governing PDEs are ill-conditioned and raise challenges for numerical solution methods.

Fractures introduce additional representation difficulties when designing a reservoir model. They span multiple length scales and exhibit complex intersections at arbitrary angles. In addition, their flow properties are in sharp contrast to those of the matrix rock, ranging from highly permeable fluid conduits, to cemented barriers, which inhibit the flow. In reservoir models, the small apertures of fractures classifies them as sub-grid features (i.e. 2D surfaces embedded into the 3D domain) and, in order to speed up computations, a hierarchical approach is typically employed [18]. More specifically, before the simulation, fractures shorter than the size of a grid cell are homogenized by altering the effective permeability of the host rock [19, 20], while the remainder are treated explicitly. The literature has seen the emergence of several types of fracture models, including Dual-Porosity / Dual-Permeability (DPDK) [21–23], where fractures are the flow actors and the matrix acts as fluid storage, Discrete Fracture Models (DFM) [24–26], which confine fractures at the interfaces between rock cells using complex unstructured grids, and, more recently, Embedded Discrete Fracture Models (EDFM) [18, 19, 27]. The latter avoid the geometrical complexities arising from the intersections within a fracture network, by defining separate independent grids for the fracture and rock domains. The coupling is instead achieved by appropriately-defined cross-media source / sink terms.

Finally, despite advances in field data acquisition and interpretation techniques [28], the considerable depths at which the target geological formations reside (1-10 km), makes the accurate measurement of their rock properties a difficult task. This adds a layer of uncertainty, which must be taken into account when performing flow simulations for robust forecasting. A common way of tackling this challenge is to consider a set of possible geological scenarios [29], derived e.g. from geostatistics and expert assessment.

In conclusion, having to perform an *ensemble* (typically, 100 - 1000) of large-scale high resolution simulations over long time periods (decades), makes modeling flow through real-field reservoirs under uncertainty a computationally expensive task. And this goes without even mentioning the numerical aspects (i.e. stability and convergence) of solving the complex (non-linear) system of coupled Partial Differential Equations (PDEs) associated with the physical processes involved.

1.2. Simulation methods – literature review

Fig. 1.3 gives some examples of methods designed to tackle these challenges, separated into three categories, based on the trade-off between accuracy and computational efficiency they employ.

After identifying the equations that govern the relevant physical processes (e.g. mass, momentum, energy balance), one can choose to perform the simulation at full (fine-scale) resolution (Fig. 1.4(c)). This approach ensures maximum accuracy, however, given the large number of degrees of freedom (DOF) in the representation of the solution (up to billions in real-field cases), which need to be resolved at each time-step, it can prove impractical on even the latest generation of computing hardware.

On the opposite side of the spectrum lie methods that reduce the number of computations by limiting the number of DOFs (see, e.g., [30–32]). Among them, *upscaling* methods solve the equations on a coarse domain. To facilitate this, ef-

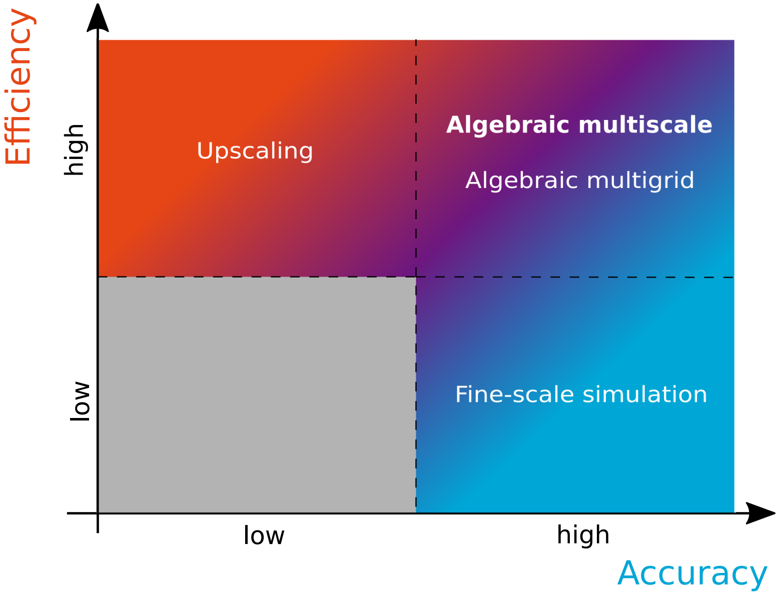


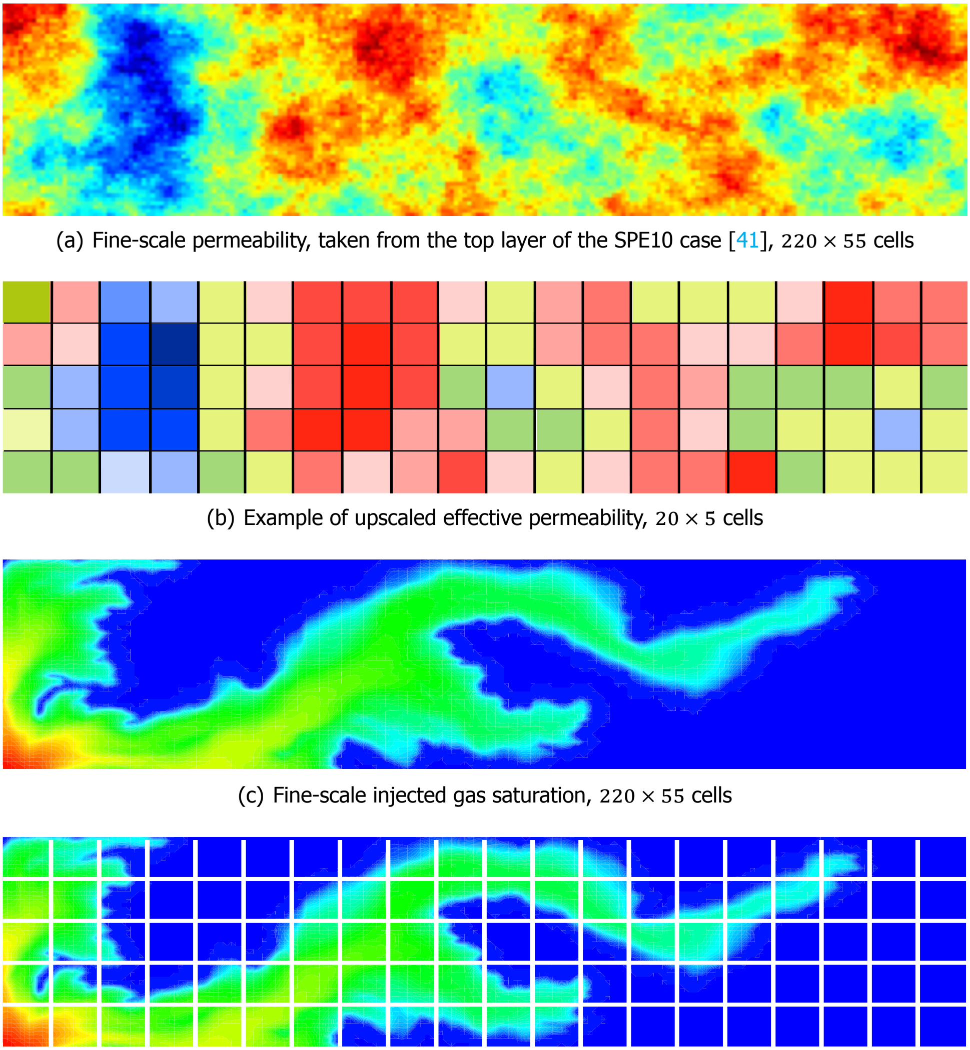
Figure 1.3: Examples of real-field flow simulation techniques and the trade-off between accuracy and computational efficiency they achieve.

fective coarse-scale conductivities are determined from the fine-scale geological description (Fig. 1.4(b)), e.g. using a weighted averaging approach [33–36] or by solving the flow equations locally [37–40]. Significant speed-ups can be achieved, as a result, however, the loss in resolution can be detrimental to their forecasting power, especially in cases when small-scale features, such as fractures, are dictating the flow in the domain.

The middle-ground features methods that interpolate the solution from the coarse domain (composed either of one coarse grid or a hierarchy of coarse levels) back onto the original resolution, thus recovering the fine-scale details of the flow patterns. This is the philosophy behind Algebraic MultiGrid (AMG) methods, which constitute the subject of a mature body of literature [44]. The proven robustness and wide applicability has lead to their availability as commercial black-box linear system solvers [45].

MultiScale Finite Element (MSFE) [46–51] and Finite Volume (MSFV) [42, 52–55] methods have recently emerged, as alternative to upscaling, for accurate and efficient porous media flow simulation. They map the solution between coarse- and fine-scale using a set of specially-crafted *basis functions*, which capture the local heterogeneities (Fig. 1.4(d)) [52]. Also, in the case of MSFV, a mass-conservative flux field can be reconstructed [42], which can be used to resolve the transport (i.e. saturation) equations [54, 55].

The advent of the iterative approach [56] has opened the possibility for the use of multiscale methods as fully-converged flow solvers. To this end, the multiscale stage (aimed at damping low-frequency errors) is paired with an inexpensive



(a) Fine-scale permeability, taken from the top layer of the SPE10 case [41], 220×55 cells

(b) Example of upscaled effective permeability, 20×5 cells

(c) Fine-scale injected gas saturation, 220×55 cells

(d) Injected gas saturation obtained using the reconstructed MSFV flux [42] from the multiscale pressure solution on a coarse grid with 20×5 blocks

Figure 1.4: Comparison between the solutions obtained using fine-scale (c) and multiscale (d) methods on the top layer of the SPE10 permeability field [41] (a). Images adapted from [43].

fine-scale smoother (e.g. ILU(0) [57], to address high-frequency errors) and iterated until the solution reaches machine accuracy [58, 59]. Alternatively, only a few iterations can be employed, e.g. according to a user-specified tolerance, $\epsilon \gg \epsilon_m$, followed by the MSFV flux reconstruction procedure. This approach was shown to deliver meaningful approximate results for decision-making, at a fraction of the computational cost [60] and without sacrificing resolution or compromising the stability of the simulator.

The past two decades have seen rapid advances in the evolution of multiscale methods. Due to their increasing popularity, an Algebraic MultiScale formulation (AMS), described in terms of operators (prolongation and *restriction*) has been devised [52, 61]. This has accelerated ongoing efforts to integrate these methods into both traditional and next-generation simulators, including those that make use of parallel computing hardware [62].

The computation of the multiscale basis functions is the subject of ongoing debate. Traditionally, their local support has been ensured by imposing *localization assumptions*, according to the *wirebasket* topology of the *dual-coarse grid* [63]. Other authors have devised formulations based on numerical Green's functions [64, 65]. Recently, an iterative restricted-smoothing procedure has been proposed (MsRSB) [66], where the dual grid is replaced by interaction regions and locality is ensured via rescaling.

Initially focused exclusively on incompressible single-phase flow scenarios, AMS methods are the subject of continuous extensions in terms of the physical processes they target. Recent contributions to the literature have seen them used to capture the effects of gravity [67], complex wells [68, 69], in black-oil simulation [70] and compositional flow [71, 72]. From the total mass balance (i.e. *pressure*) equations, their scope has been extended to tackle transport (*phase saturation*) [73, 74], energy balance [75, 76], as well as the elastic deformation (*displacement*) in solids [77] and *poromechanics* (flow coupled with displacement) of fractured and faulted reservoirs [78, 79].

AMS has been successfully applied in the context of various coupling strategies between the unknowns involved in multiphase flow (pressure and phase saturation), from IMPES (Implicit Pressure, Explicit Saturation) [54], to sequentially-implicit and monolithic (i.e. fully-implicit) simulators [80], using structured, as well as unstructured grids [66, 81–84]. The transition from a single coarse grid to multiple coarse levels has enabled the use of multiscale methods in simulations featuring Dynamic Local Grid Refinement (DLGR) [85].

Enriched multiscale methods have been recently proposed to tackle media exhibiting particularly high conductivity contrasts [86–88], while ensuring monotonicity in the results [89, 90].

Finally, there is ongoing research regarding the use of multiscale-based simulators as the backbone for history matching and production optimization cycles [91] in the Closed-Loop Reservoir Management (CLRM) framework [92].

In 2014, which marked the beginning of the present PhD project, many of the developments mentioned above were in their infancy or subjects of "future work". The possibility of decoupling the formulation of the multiscale basis functions from that of the pressure equations had only recently been considered, in the context of compressible flow [43, 52, 70, 93]. The first benchmark studies on the performance of AMS for flow in unfractured heterogeneous porous media, based on processor (CPU) time, were fresh out of press [61]. More importantly, the application of AMS to flow in the presence of fractures was at its first attempts [27, 94] and multiscale for geomechanics was still a few years away. This was the context that lead to the formulation of the project's research objective and goals, as detailed in the next Section.

1.3. Research goals

The main objective of the present project is to

> *Develop a flexible and extensible framework for fractured reservoir simulation using algebraic multiscale methods.*

Towards this end, the following goals have been achieved:

- Studying the application of AMS to compressible flow through porous media.
- Modelling multiphase flow in the presence of embedded fractures and the application of multiscale methods to such cases.
- Investigating the impact of elastic rock deformation on fluid flow.

1.4. Thesis outline

This thesis is a compilation of material from published journal articles, as well as results and papers prepared for conferences, generated during the 4-year PhD program. The first two research goals constitute the subjects of individual chapters. More specifically, Chapter 2 presents the extension of AMS methods to more realistic physical cases by including the effects of pressure-driven compressibility. Then, in Chapter 3, the solution of the flow equations in the presence of fracture networks, embedded into the matrix rock, is investigated. The findings of this study have led to the development of a novel Projection-based Embedded Discrete Fracture Model (pEDFM), able to handle sub-grid features with a wide range of conductivity contrasts, which is detailed in Chapter 4. Finally, the derivation of a multiphase fully compressible poromechanics simulator, with sequentially-implicit coupling, is discussed and validated in Appendix D.

Each chapter follows a predefined structure. The beginning consists of a short topic-specific introduction, which expands on what was presented above. Then, the discussion shifts towards the governing equations, along with their discretization and linearization, the coupling between the unknowns and the approach used to compute the fine-scale solution. The latter is further expanded upon by the presentation of a suitable algebraic multiscale strategy, tailored to the specifics of the simulation scenario at hand. Finally, the results of the set of supporting numerical experiments are discussed and conclusions are drawn. Chapter 5 gathers and reviews these conclusions, in the context of the research objective and goals, and outlines directions for further research.

2

Pressure-driven fluid and rock compressibility

Up to 2014, the AMS community focused mostly on solving the linear elliptic equations arising from incompressible single-phase flow (see e.g., [61]). They are, however, not applicable in cases when gas is present in the subsurface formation and / or injected as displacing fluid. At the same time, the properties of the host rock are known to change during exploitation. Therefore, pressure-driven fluid and rock compressibility need to be taken into account when simulating realistic cases involving flow through porous media. More specifically, both the fluid density and rock porosity are allowed to vary with pore pressure, thus, introducing time dependency and non-linearities into the mass balance equations. The resulting system of parabolic Partial Differential Equations (PDEs), needs to be linearized and solved iteratively, which raises additional concerns in terms of computational expense and convergence [97].

Answering the opportunity raised by the considerations above, the present study introduces the first generic iterative AMS procedure for compressible flows in heterogeneous porous media (C-AMS), along with a thorough study of its computational efficiency (CPU time) and convergence behaviour (number of iterations). This development is crucial in extending the applicability of multiscale methods to more realistic physical scenarios.

Towards this end, a non-overlapping coarse partition of the (fine-scale) simulation grid is performed, along with its overlapping dual-coarse counterpart. Then, the *restriction* (i.e. a map from fine- to coarse-scale, or coarsening) and *prolongation* (coarse- to fine, or interpolation) operators are constructed. The latter represents a collection of locally-computed and infrequently updated *basis functions*, which capture the effect of fine-scale heterogeneities.

The material presented in this chapter has been published in the proceedings of the ECMOR XIV - 14th European Conference on the Mathematics of Oil Recovery (2014) [95] and in the Journal of Computational Physics **300**, 679 (2015) [96].

In contrast to the incompressible case, however, the construction of these basis functions for compressible flow problems is not straightforward. In the past, formulations based on the incompressible elliptic [93], compressible elliptic [52, 70] and pressure-independent parabolic [43] mass balance equations have been considered. However, the literature lacks a systematic study to reveal the benefit of using one option over the other, especially when combined with a fine-scale smoother stage. Moreover, no study of the overall computational efficiency of the multiscale methods has been done, so far, for compressible three-dimensional problems.

In order to develop an efficient prolongation operator, in this work, several formulations for basis functions are considered. These differ from each other in the amount of compressibility involved, ranging from incompressible elliptic to compressible parabolic types. In terms of the restriction operator, both MultiScale Finite Element (MSFE) [46] and MultScale Finite Volume (MSFV) [42] are employed. In addition, the possibility of applying each of the two operators in succession is studied, allowing C-AMS to benefit from the Symmetric Positive Definite (SPD) algebraic property of MSFE and the mass conservative solutions of MSFV [98].

Similar to [56], an iterative strategy is devised, during which, low-frequency errors are resolved by the global (multiscale) stage, while high-frequency errors are damped using a second-stage smoother at the fine-scale. Two options are considered for the smoother: the widely used local correction functions [43], extended, here, to account for different degrees of compressibility, as well as ILU(0) [57]. Sensitivity studies, based on CPU times, are performed w.r.t each component in the C-AMS procedure, in order to determine the most effective multiscale strategy for compressible flow through 3D heterogeneous porous media. It is important to note that the time spent during setup, linear system population and solution are measured separately - a study which was not considered in previously published compressible multiscale works.

The results in the present study show that only a few C-AMS iterations are enough in order to obtain a high-quality estimation of the fine-scale solution. As such, the method can be employed as an efficient approximate solver, in combination with a fine-scale conservative reconstruction procedure [42]. In the benchmark studies of this work, however, it is iterated until the results reach machine accuracy. Therefore, its performance as a fully converged solver is measured and compared against the industrial-grade Systems Algebraic MultiGrid method, SAMG [45]. This comparative study for compressible problems is the first of its kind, and is made possible through the presented algebraic formulation, which allows for easy integration of C-AMS in existing advanced simulation platforms. The numerical results, elicited from simulations on heterogeneous 3D domains, recommend C-AMS as an efficient method for compressible flow.

The chapter is structured as follows. First, the governing equations are briefly visited. Then, the formulation of the proposed C-AMS method is presented, while considering several options for the prolongation, restriction operators and the second-stage solver (i.e. smoother). The adaptive updating of these operators is studied (i.e. only if required at a given time-step), along with the possibility of early exit in the linear solution stage of the Newton-Raphson loop (i.e. before the residual norm reaches machine precision). The results of numerical experiments on a wide range of 3D heterogeneous test cases are subsequently discussed. Their aim is to determine the optimum multiscale solution strategy, whose performance is, ultimately, compared to SAMG, both in terms of the number of iterations and overall CPU time. The discussion of these results, finally, leads to the conclusions.

2.1. Governing equations

This section presents the mass balance equations governing isothermal single-phase flow through porous media, when gravity, capillarity and compositional effects are neglected. First, the reference case is visited, when both the fluid and the matrix rock are incompressible. These assumptions are subsequently relaxed by allowing both the rock porosity and fluid density to vary with pore pressure.

In order to illustrate the differences between the governing PDEs and their solutions in both scenarios, the resulting pressure fields, on a homogeneous 1D domain, are compared.

2.1.1. Incompressible flow

The mass balance corresponding to incompressible single-phase flow through porous media on the spatial domain Ω is governed by the following elliptic PDE,

$$\nabla \,.\, \overline{w} = q, \tag{2.1}$$

where $\nabla\,.$ is the divergence operator, q denotes volumetric source / sink terms, e.g. arising from injection / production wells, respectively, while $\overline{w}$ is the volumetric flux, defined, according to Darcy's law, as

$$\overline{w} = -\mathbf{k}\lambda \cdot \nabla p, \tag{2.2}$$

with ∇, the gradient operator, $\mathbf{k}$, the 2nd order positive-definite absolute permeability tensor, here, considered diagonal and isotropic, $\lambda = 1/\mu$, the fluid mobility, μ, the fluid viscosity and, p, the pore pressure.

In order to solve Eq. (2.2) numerically, a discretization grid is defined on Ω, composed of N *cells*, $\Omega_i, \forall i = 1, \ldots, N$. Then, by integrating over each Ω_i,

$$-\int_{\Omega_i} \nabla \,.\, (\mathbf{k}\lambda \cdot \nabla p) \;\, dV = \int_{\Omega_i} q \;\, dV, \tag{2.3}$$

the Gauss theorem can be applied,

$$-\int_{\partial\Omega_i} \mathbf{k}\lambda \cdot \nabla p \;.\, \overline{n} \;\, dS = \int_{\Omega_i} q \;\, dV, \tag{2.4}$$

where . stands for the dot product, $\partial\Omega_i$ denotes the cell's superficial area (*interface*) and $\overline{n}$ its normal *versor*.

$\mathbf{k}$ at the interface is computed by harmonically averaging the values corresponding to the two neighbouring cells. Then, under the assumption that these mobility coefficients (see Eq. (2.2)) do not vary with p, Eq. (2.4) is linear and can be written in algebraic form as

$$\mathbf{A}\,\overline{p} = \overline{q}, \tag{2.5}$$

where the $N \times N$ sparse matrix $\mathbf{A}$ contains *transmissibilities* in its off-diagonals, $\overline{p}$ is the vector of N pressure unknowns, while $\overline{q}$ contains the N Right-Hand-Side (RHS) terms.

The solution of this linear system for a 1D homogeneous reservoir (i.e. $\mathbf{k} = k$ constant over Ω) with Dirichlet boundary conditions set to non-dimensional pressure values 1 on the left and 0 on the right, is given in Fig. 2.1.

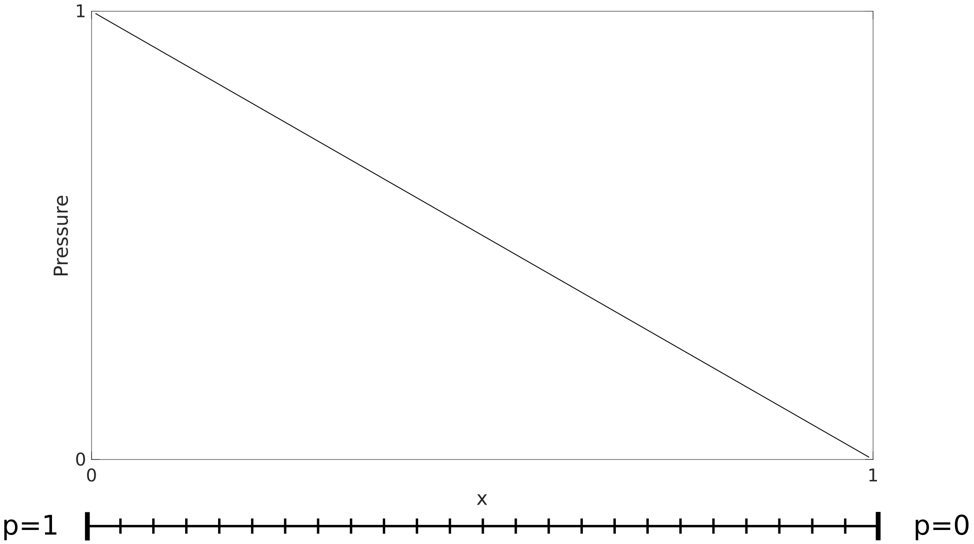


Figure 2.1: Incompressible single-phase pressure solution corresponding to a 1D homogeneous reservoir of length $L = 1$, with Dirichlet boundary conditions.

2.1.2. Compressible flow

If the rock porosity ϕ and fluid density ρ are allowed to vary with pore pressure, the PDE describing mass balance over domain Ω is

$$\frac{\partial\left(\phi\rho\right)}{\partial t} - \nabla\,.\left(\rho\boldsymbol{\lambda}\cdot\nabla p\right) = \rho q. \tag{2.6}$$

Note that, in comparison with Eq. (2.1), Eq. (2.6) contains a time-dependent *accumulation* term, $\partial\left(\phi\rho\right)/\partial t$, and the presence of ρ in the *advection* term, $-\nabla\,.(\rho\mathbf{k}\lambda\cdot\nabla p)$, makes the transmissibilities non-linear.

The semi-discrete form of this nonlinear flow equation after implicit (Euler-backward) time integration from time-step n to $n+1$ and division by $\rho^{(n+1)}$, reads

$$\frac{\phi^{(n+1)}}{\Delta t} - \frac{\phi^{(n)}}{\Delta t}\frac{\rho^{(n)}}{\rho^{(n+1)}} - \frac{1}{\rho^{(n+1)}}\nabla\,.\left(\rho^{(n+1)}\mathbf{k}\lambda\cdot\nabla p^{(n+1)}\right) = q^{(n+1)}. \tag{2.7}$$

Using first-order Taylor expansion in the accumulation term,

$$\phi^{(\nu+1)} \approx \phi^{(\nu)} + \left.\frac{\partial\phi}{\partial p}\right|^{(\nu)}\left(p^{(\nu+1)} - p^{(\nu)}\right) \tag{2.8}$$

$$\frac{1}{\rho^{(\nu+1)}} \approx \frac{1}{\rho^{(\nu)}} + \left.\frac{\partial^1\!/\!\rho}{\partial p}\right|^{(\nu)}\left(p^{(\nu+1)} - p^{(\nu)}\right) \tag{2.9}$$

leads to a Newton-Raphson loop, which, at iteration $\nu+1$, reads,

$$c^{(\nu)}\left(p^{(\nu+1)} - p^{(\nu)}\right) - \frac{1}{\rho^{(\nu)}}\nabla\,.\left(\rho^{(\nu)}\mathbf{k}\lambda\cdot\nabla p^{(\nu+1)}\right) = b^{(\nu)}, \tag{2.10}$$

where ρ was lagged in the advection term and

$$c^{(\nu)} = \frac{1}{\Delta t}\left[\left.\frac{\partial\phi}{\partial p}\right|^{(\nu)} - \phi^{(n)}\left.\frac{\partial^1\!/\!\rho}{\partial p}\right|^{(\nu)}\rho^{(n)}\right], \tag{2.11}$$

$$b^{(\nu)} = -\frac{\phi^{(\nu)}}{\Delta t} + \frac{\phi^{(n)}}{\Delta t}\frac{\rho^{(n)}}{\rho^{(\nu)}} + q^{(n+1)}. \tag{2.12}$$

Note that, as $\nu\to\infty$, $p^{(\nu+1)}\to p^{(n+1)}$, hence $\left(p^{(\nu+1)} - p^{(\nu)}\right)\to 0$ and the linearized Eq. (2.10) converges to the non-linear Eq. (2.7). Therefore, the coefficient c, which is a by-product of the linearization process, plays a role only during Newton-Raphson iterations. This fact opens up the possibility to alter c by computing it based on either $p^{(\nu)}$, resulting in $c^{(\nu)}$ from Eq. (2.11), or $p^{(n)}$, i.e. $c^{(n)}$, which is obtained by setting $\nu = n$ in the same equation. Each choice can potentially lead to a different convergence behaviour.

Eq. (2.10) can be written algebraically as

$$\mathbf{A}^{(\nu)}\,\overline{p}^{(\nu+1)} = \overline{f}^{(\nu)}, \tag{2.13}$$

where

$$\mathbf{A}^{(\nu)} = \mathbf{C}^{(\nu)} + \mathbf{T}^{(\nu)}. \tag{2.14}$$

Here, $\mathbf{C}^{(\nu)}$ is a diagonal matrix having $c\Delta V$, evaluated at cell Ω_i, as i-th entry, where ΔV denotes the cell's volume. Also, $\mathbf{T}^{(\nu)}$ contains transmissibilities in its off-diagonals. Finally, the total RHS terms are assembled in the vector $\overline{f}^{(\nu)}$,

$$\overline{f}^{(\nu)} = \overline{b}^{(\nu)} + \mathbf{C}^{(\nu)}\,\overline{p}^{(\nu)}, \tag{2.15}$$

where the vector $\overline{b}^{(\nu)}$ gathers the terms given in (2.12).

Consider the the homogeneous 1D case studied earlier, with an initial pressure $p_0(x) = 0, \forall x \in \Omega$ and the following constitutive relationships,

$$\phi(p) = \phi_0 c_r (p - p_0), \tag{2.16}$$

$$\rho(p) = \rho_0 \exp\left(c_f (p - p_0)\right), \tag{2.17}$$

where c_r and c_f are the rock and fluid compressibility coefficients, respectively. The results of solving Eq. 2.13 over several time steps is depicted in Fig. 2.2. Note the change in the curvature of the pressure field.

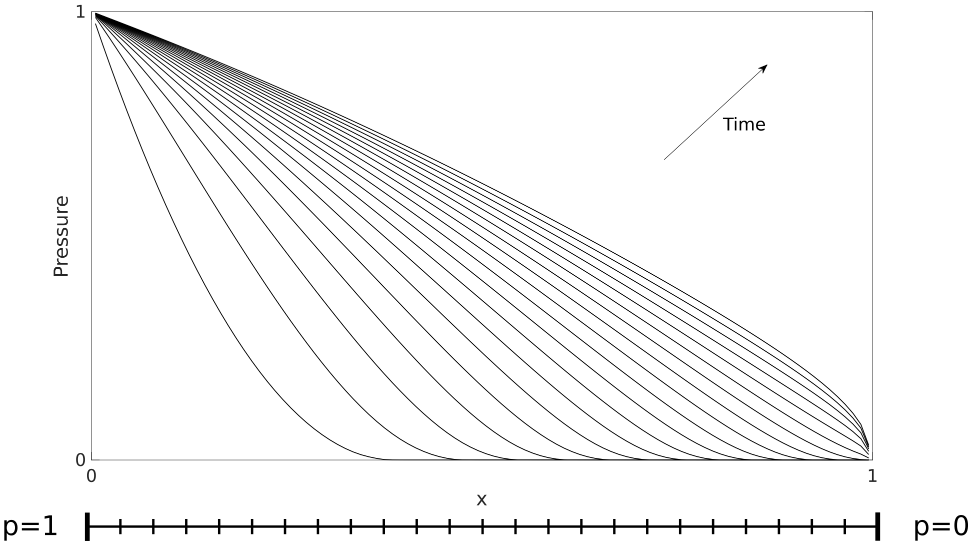


Figure 2.2: Compressible single-phase pressure solution, over several time-steps, corresponding to a 1D homogeneous reservoir of length $L = 1$, with initial pressure $p_0 = 0$ and Dirichlet boundary conditions.

2.2. Algebraic multiscale formulation (C-AMS)

This section presents the formulation of the Compressible Algebraic MultiScale method (C-AMS), as an efficient alternative to directly solving Eq. (2.13), for each time-step.

2.2.1. Primal and dual-coarse grids

C-AMS computes the solution to Eq. (2.13) on a primal-coarse domain (Fig. 2.3(b)) – a non-overlapping partition of the fine-scale grid (Fig. 2.3(a)) into N_c blocks, $\breve{\Omega}_{l,c}, \forall l = 1, \dots, N_c$.

This coarse partition, subsequently, leads to the construction of a *dual-coarse* domain, following the *wirebasket* hierarchy [63]. More specifically, first, one fine cell within each coarse block is selected and labelled as a *dual-coarse vertex* (Fig. 2.3(c)). Then, 1D domains connecting the vertices to each other, or to domain boundaries, are identified to form the *dual-coarse edges* (Fig. 2.3(d)). In similar fashion, the 2D domains connecting the edges are *dual-coarse faces* (Fig. 2.3(e)) and, finally, the 3D domains between faces constitute *dual-coarse interiors* (Fig. 2.3(f)).

On a 3D domain, the dual-coarse elements around each (3D) interior – the surrounding (2D) faces, their neighbouring (1D) edges and, subsequently, the adjacent (0D) vertices – form the corresponding dual-coarse block, $\widetilde{\Omega}_{h,d}, h = 1, \dots, N_d$. Fig. 2.4 shows the overlap between primal and dual-coarse blocks on a 2D domain – since there are no interiors in this case, the dual blocks are formed around faces, instead.

2.2.2. Multiscale operators

The transfer operators between fine-scale and coarse-scale are the multiscale Restriction ($\mathcal{R}$) and Prolongation ($\mathcal{P}$). The former is defined based on either Finite Element (MSFE), i.e., $\mathcal{R}^{FE} = \mathcal{P}^T$, or Finite Volume (MSFV), for which $\mathcal{R}^{FV}$ corresponds to the integral over primal-coarse blocks, i.e.,

$$\mathcal{R}^{FV}(i,j) = \begin{cases} 1 & \text{, if fine-cell } j \text{ is contained in primal-coarse block } i \\ 0 & \text{, otherwise.} \end{cases} \tag{2.18}$$

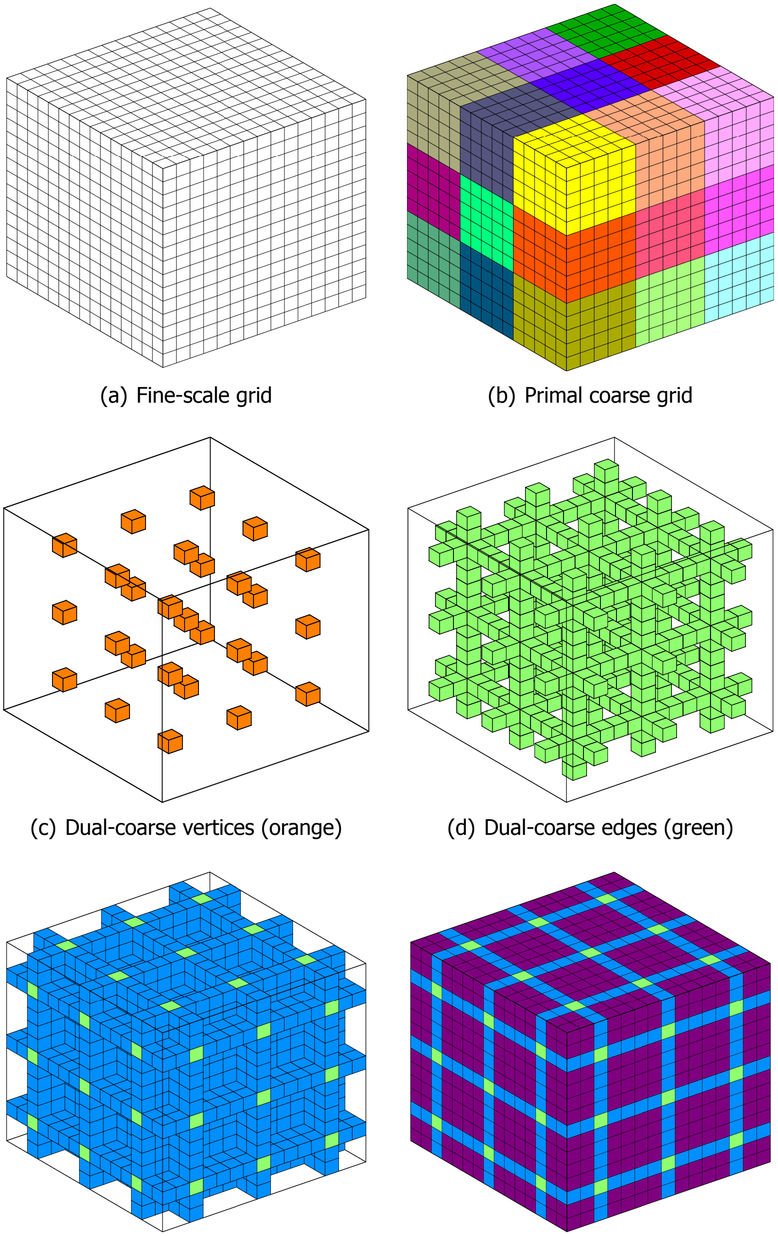
The columns of $\mathcal{P}$ are the basis functions, one for each primal coarse block, which are solutions of the mass balance equation (D.36), computed subject to simplified boundary conditions, as follows. The value of the basis function i at each vertex j is set as

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}. \tag{2.19}$$

Then the dual elements in the neighbourhood of node i are resolved, in sequence. The edge values are computed in 1D fashion, i.e. neglecting any fluxes to / from nearby face cells and using the values at the vertices as Dirichlet boundary conditions. Next, the values at the faces are obtained using the edges as boundary conditions and neglecting the connections to the interiors. Finally, the interiors can be computed, based on the values of their surrounding faces.

The neglected transmissibilities towards dual elements of inferior rank in the wirebasket hierarchy, along with the Dirichlet boundary conditions, constitute the so-called *localization assumption* (see Appendix B.4 and [61] for more details). It ensures that the support of each basis function is constrained to the neighbourhood of the corresponding dual-coarse vertex. Note that an alternative approach has recently been proposed in the literature, where the basis functions are iteratively computed and rescaled to obtain a similar effect, without the need for a dual-coarse grid [66].

The basis functions, $\Phi_l, \forall l = 1, \dots, N_c$, are assembled over dual-coarse cells $\widetilde{\Omega}_{h,d}, \forall h \in \{1, ..., N_d\}$, i.e., $\Phi_l = \bigcup_{h=1}^{N_d} \Phi_{l,h}$. In contrast to the incompressible AMS



(a) Fine-scale grid

(b) Primal coarse grid

(c) Dual-coarse vertices (orange)

(d) Dual-coarse edges (green)

(e) Dual-coarse faces (blue)

(f) Dual-coarse interiors (purple)

Figure 2.3: Multiscale primal (b) and dual-coarse partitions (c-e) of a 3D fine-scale grid (a).

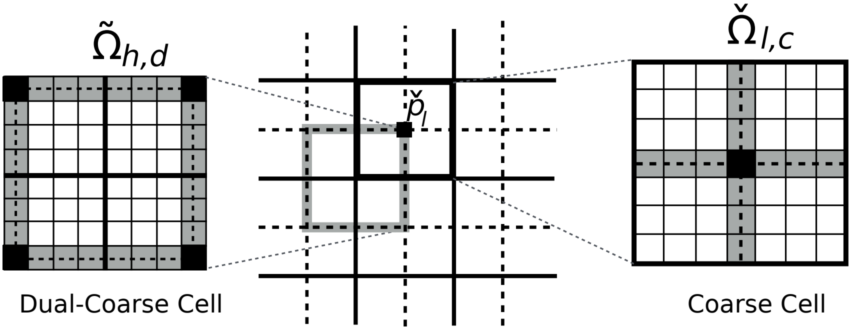


Figure 2.4: Overlap between a primal and dual-coarse block on a 2D domain. Adapted from [27].

[61], in C-AMS, the equations for these functions need not exactly coincide with the fine-scale mass balance (D.36). Instead, C-AMS can accommodate different formulations, depending on the level of compressibility involved. Two options are

$$c^{(\nu)}\Phi_{l,h}^{(\nu+1)} - \frac{1}{\rho^{(\nu)}}\nabla\,.\left(\rho^{(\nu)}\mathbf{k}\lambda\cdot\nabla\Phi_{l,h}^{(\nu+1)}\right) = 0, \tag{2.20}$$

and

$$-\frac{1}{\rho^{(\nu)}}\nabla\,.\left(\rho^{(\nu)}\mathbf{k}\lambda\cdot\nabla\Phi_{l,h}^{(\nu+1)}\right) = 0, \tag{2.21}$$

both being pressure dependent (through c and / or ρ), but different in the sense of the consideration of the accumulation term, c. Alternatively, one can also calculate basis functions using

$$c^{(n)}\Phi_{l,h} - \nabla\,.\left(\mathbf{k}\lambda\cdot\nabla\Phi_{l,h}\right) = 0, \tag{2.22}$$

or

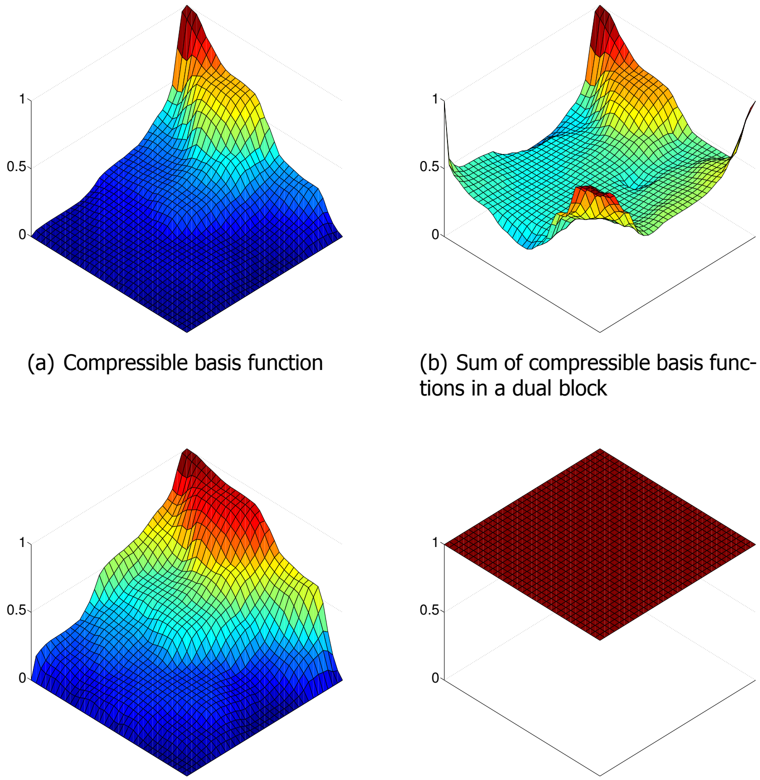
$$-\nabla\,.\left(\mathbf{k}\lambda\cdot\nabla\Phi_{l,h}\right) = 0 \tag{2.23}$$

which are both pressure independent (since c is now omitted or based on the pressure from the previous time step). Fig. 2.5 illustrates that the basis functions do not form a partition of unity when compressibility effects are included, which is the intrinsic nature of the parabolic compressible equation.

The choices formulated above need to be treated differently during time-lapse simulations. More precisely, the basis functions based on Eqs. (2.20) and (2.21) depend on pressure and, hence, need to be updated adaptively when it changes significantly. Eqs. (2.22) and (2.23) are pressure independent; thus, they only need to be computed once, at the beginning of the simulation.

2.2.3. Correction function

Note that the basis function equations ignore the influence of RHS terms and are subject to reduced-problem boundary conditions along dual-coarse block bound-



(a) Compressible basis function

(b) Sum of compressible basis functions in a dual block

(c) Incompressible bassis function

(d) Sum of incompressible basis functions in a dual block

Figure 2.5: Two choices of multiscale basis functions in a reference dual-coarse block (left), Summation of the basis functions over the dual-coarse block (right), i.e., partition of unity check.

aries $\partial\widetilde{\Omega}_{h,d}$. These are the two main sources of inaccuracies in the multiscale pressure solution. In order to mitigate them, one can employ a smoother, such as that formed from the ILU(0) decomposition [57] of $\mathbf{A}^{(\nu)}$ from (2.13), and / or compute the so-called *correction function*, Ψ (see e.g. [61]).

Ψ is computed in similar fashion to the basis functions, by plugging the RHS term, $\overline{f}^{(\nu)}$, into Eqs. (2.20)-(2.23). Thus, four different types of correction functions can be defined for compressible flow problems. After the computation is done in each dual-coarse block, the values are assembled as $\Psi = \bigcup_{h=1}^{N_d} \Psi_h^{\nu}$.

While the basis and correction functions based on Eq. (2.22) were previously used [43], the other options have not been studied in the literature, previous to this work.

2.2.4. C-AMS algorithm

The C-AMS approximates the fine-scale solution $\mathbf{p}^{\nu}$ by $\mathbf{p'}^{\nu}$ using the Prolongation operator $\mathcal{P}$, which is a matrix of size $N \times N_c$, having basis function Φ_l in its l-th column. The map between the coarse ($\breve{p}$) and fine-scale solution ($\overline{p}'$) reads

$$\overline{p}' = \mathcal{P}\breve{p}. \tag{2.24}$$

The coarse-scale system is obtained using the restriction operator, $\mathcal{R}$, as

$$\breve{\mathbf{A}}^{(\nu)}\breve{p}^{(\nu+1)} \equiv \left(\mathcal{R}\mathbf{A}^{(\nu)}\mathcal{P}\right)\breve{p}^{(\nu+1)} = \mathcal{R}\overline{f}^{(\nu)}. \tag{2.25}$$

Substituting Eq. (2.24) into (2.25) leads to

$$\overline{p}'^{(\nu+1)} = \mathcal{P}\left(\mathcal{R}\mathbf{A}^{(\nu)}\mathcal{P}\right)^{-1}\mathcal{R}\overline{f}^{(\nu)}. \tag{2.26}$$

or, in residual form,

$$\delta\overline{p}'^{(\nu+1)} = \mathcal{P}\left(\mathcal{R}\mathbf{A}^{(\nu)}\mathcal{P}\right)^{-1}\mathcal{R}\overline{r}^{(\nu)}. \tag{2.27}$$

Here $\overline{p}'^{(\nu+1)} = \overline{p}^{(\nu)} + \delta\overline{p}'^{(\nu+1)}$, while $\overline{r}^{(\nu)} = \overline{f}^{(\nu)} - \mathbf{A}^{(\nu)}\overline{p}^{(\nu)}$ is the fine-scale residual. Note that all the different options for basis functions can be considered in construction of the prolongation operator.

The C-AMS employs Eq. (2.26) as the global solver, which targets low-frequency errors. In addition to this, an efficient convergent multiscale solver needs to include a second-stage smoother at fine scale, which accounts for the remaining high-frequency errors. These can arise from the basis function localization conditions, the non-linearity of the operator and the complex RHS term. Among the choices for the smoother (block-, line-, or point-wise solvers), the correction functions (CF) and ILU(0) are considered in this work. The C-AMS procedure is finally summarized in Table 2.1.

Do until convergence ($||\overline{\epsilon}||_2^{(\nu)} < e$) achieved (see Eq. (2.31)):

1. **Initialize:** update linear system components, $\mathbf{A}^{(\nu)}$ and $\overline{f}^{(\nu)}$, based on $\overline{p}^{(\nu)}$
2. **Update residual:** $\overline{r}^{(\nu)} = \overline{f}^{(\nu)} - \mathbf{A}^{(\nu)}\overline{p}^{(\nu)}$
3. **Adaptively compute Basis Functions**: use either of Eqs. (2.20)-(2.23)
4. **Pre-smoothing Stage**: only if CF is used, apply CF on $\overline{r}^{(\nu)}$ and update residual
5. **Multiscale Stage:** Solve (2.27) for $\delta\overline{p}'^{(\nu+1/2)}$
6. **Post-smoothing Stage:** smooth $\overline{p}'^{(\nu+1/2)}$ for n_s times using a fine-scale iterative solver (here, ILU(0) is used), obtaining $\delta\overline{p}'^{(\nu+1)}$
7. **Update solution:** $\overline{p}'^{(\nu+1)} = \overline{p}^{(\nu)} + \delta\overline{p}'^{(\nu+1/2)} + \overline{p}'^{(\nu+1)}$
8. **Update error:** compute $\overline{\epsilon}^{(\nu)}$, and assign $\overline{p}^{(\nu)} \leftarrow \overline{p}^{(\nu+1)}$

Table 2.1: C-AMS iteration procedure, converging to $\overline{p}^{(n+1)}$ with tolerance e.

In the next section, numerical results for 3D heterogeneous test cases are presented, in order to provide a thorough assessment of the applicability of C-AMS to large-scale problems.

2.3. Numerical Results

The numerical experiments presented in this section are divided into: (1) finding a proper iterative procedure and multi-stage multiscale components for efficiently capturing the non-linearity within the flow equation, and (2) systematic performance study by comparing against a commercial algebraic multigrid solver, i.e., SAMG [45]. Note that the second aspect is mainly to provide the computational physics community with an accurate assessment of the convergence properties of the state-of-the-art compressible multiscale solver (i.e., C-AMS). As an advantage over many advanced linear solvers, C-AMS allows for the reconstruction of a locally conservative velocity field after any MSFV iteration [99].

The numerical experiments are conducted on sets of log-normally distributed permeability fields with spherical variograms, generated generated using sequential Gaussian simulations [100]. The variance and mean natural logarithm of the permeability, i.e., $ln(k)$, for all test cases are 4 and -1, respectively, unless otherwise mentioned. Furthermore, the fine-scale grid size and dimensionless correlation lengths, ψ_1, ψ_2 and ψ_3, are provided in Table 2.2. Each set has 20 statistically-equivalent realizations. The sets with orientation angle of 15° are referred to as the *layered fields*. Also, the grid aspect ratio α is 1, i.e., $\Delta x/\alpha = \Delta y = \Delta z = 1$ m, unless otherwise specified.

Permeability Set	1	2	3	4	5	6
Fine-scale grid	64^3	128^3	256^3	64^3	128^3	256^3
ψ_1	0.125	0.125	0.125	0.5	0.5	0.5
ψ_2	0.125	0.125	0.125	0.03	0.03	0.03
ψ_3	0.125	0.125	0.125	0.01	0.06	0.01
Angle between ψ_1 and y axis	patchy			15°		
Variance of ln(k)	4					
Mean of ln(k)	-1					

Table 2.2: Permeability sets, each with 20 statistically-equivalent realizations, used during numerical experiments. Sets 4-6 are referred to as 'layered fields', in which the orientation angle between the ψ_1 direction and the y axis is 15°.

All fluid properties and the simulation time are non-dimensional. The non-dimensional pressure and density are introduced as

$$p^* = \frac{p - p_{east}}{p_{west} - p_{east}}, \tag{2.28}$$

and

$$\rho^* = \frac{\rho}{\rho_0} = 1 + \eta\, p^*, \tag{2.29}$$

respectively, where the coefficient η is set to 1 for all subsequent test cases in this work.

The p_{west} and p_{east} values of 10^6 and 0 Pa, relative to the Standard (Atmospheric) condition, are considered. These correspond to non-dimensional pressure values of 1 and 0, which are set as Dirichlet conditions at the west and east boundaries, respectively, for all the cases, unless otherwise mentioned. Also, all the other surfaces are subject to no-flow Neumann conditions.

The non-dimensional time is defined as $t^* = t/\tau$, where

$$\tau = \frac{\mu \phi L^2}{\langle k \rangle \left(p_{\text{west}} - p_{\text{east}}\right)}. \tag{2.30}$$

Here, $\langle k \rangle$ is the domain-averaged permeability, and L is a length scale of the domain. Values of 10^6 Pa for the pressure difference, in-situ viscosity of 2×10^{-6} Pa.s, $\Delta x = 1$ m, $\phi = 0.1$, and $k = 10^{-12}$ m^2 are used for homogeneous cases, while $\tau = 128$ s for problem sizes of $L = 64$ m.

The results presented in this work were obtained using an implementation of C-AMS in the in-house single-threaded object-oriented C++ simulator, DARSim 1. CPU times were measured on an Intel Xeon E5-1620 v2 quad-core system with 64GB RAM.

2.3.1. C-AMS: sensitivity studies

The experiments in this section are meant to test the effect of using different options for the restriction operator, basis functions and smoother, during the C-AMS procedure. To this end, a set of 20 statistically-equivalent *patchy fields*, i.e., permeability Set 1 from Table 2.2, is considered. One of the realizations and its corresponding solution at $t^* = 0.4$ are shown in Fig. 2.6.

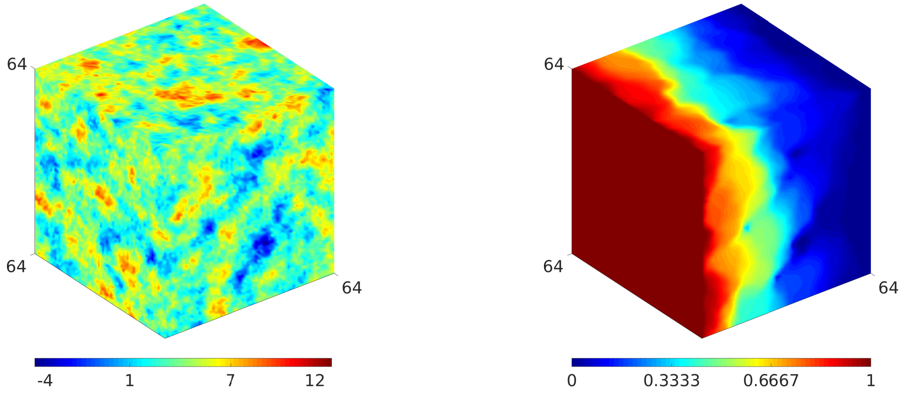


Figure 2.6: Natural-log of the permeability (left) and pressure solution after $t^* = 0.4$ (right) corresponding to one of the realization of permeability Set 1 from Table 2.2.

Nonlinear and linear level updates

In formulating a convergence criterion for the C-AMS, one can express the error of the approximate solution at step ν on the basis of either the linear or nonlinear expressions. According to Eq. (2.7), the nonlinear error in each grid cell reads

$$\epsilon^{(\nu)} = q - \frac{\phi^{(n+1)}}{\Delta t} + \frac{\phi^{(n)}}{\Delta t}\frac{\rho^{(n)}}{\rho^{(n+1)}} + \frac{1}{\rho^{(n+1)}}\nabla \cdot \left(\rho^{(n+1)}\mathbf{k}\lambda \cdot \nabla p^{(n+1)}\right), \tag{2.31}$$

and is assembled in the vector $\overline{\epsilon}^{(\nu)}$, which allows the computation of the *error norm*, $||\overline{\epsilon}^{(\nu)}||_2$. On the other hand, the linear-level error is based on the linearized equation (2.10), which leads to the computation of the *residual norm*, $||\overline{r}^{(\nu)}||_2$.

In order to determine a suitable sequence between the linear and non-linear solution stages, the same patchy domain of $64 \times 64 \times 64$ grid cells is considered (Fig. 2.6), for which the pressure equation is solved using the following strategy:

```
Do until ($||\overline{\epsilon}||_2 < 10^{-6}$) is reached:
  0. Update parameters, linear system matrix and RHS vector based on $\overline{p}^{(\nu)}$
  1. Solve linear system using the Richardson iterative scheme, preconditioned
     with one multigrid V-cycle until $||\overline{r}||_2 < 10^{-6}$
```

Table 2.3: Solution strategy used to determine a suitable stopping criterion

The error and residual norms were recorded after each iteration of the Richardson loop and are presented in Fig. 2.7. Note that the reduction of the residual norm beyond the first few iterations does not contribute to the reduction in the (non-linear) error norm. Therefore, one could ideally speed up the solution scheme by monitoring the error norm and updating the linear system after its decrease starts to stagnate. However, the computational cost of evaluating the non-linear equation is roughly the same as that of a linear system update and, thus, much more expensive than the evaluation of the residual norm.

Fig. 2.7(left) also reveals that the stagnation of the error norm happens roughly after the residual norm has been approximately reduced by $^1/_{10}$ of its initial value. Fig. 2.7(right) shows the convergence behaviour after implementing this heuristic strategy, which is more efficient, since the two norms are in agreement. Hence, in the following experiments the same strategy is employed, i.e., for linear level, $||\overline{r}^{(i)}||_2 < 10^{-1}||\overline{r}^{(0)}||_2$ after iteration i of the inner loop and, for the non-linear level, $||\overline{\epsilon}^{(\nu)}||_2 < 10^{-6}$ after iteration ν of the outer loop are set (see Table 2.3).

Adaptive updating of multiscale operators

The previous study described the first adaptive aspect considered in this work, namely, updating the linear system only after the residual norm drops by an order of magnitude. With this strategy, the basis functions remain constant during the operation of the linear solver and only get updated at the beginning of each Newton-Raphson iteration (see Fig. 2.8, top-right). The C-AMS procedure can be further

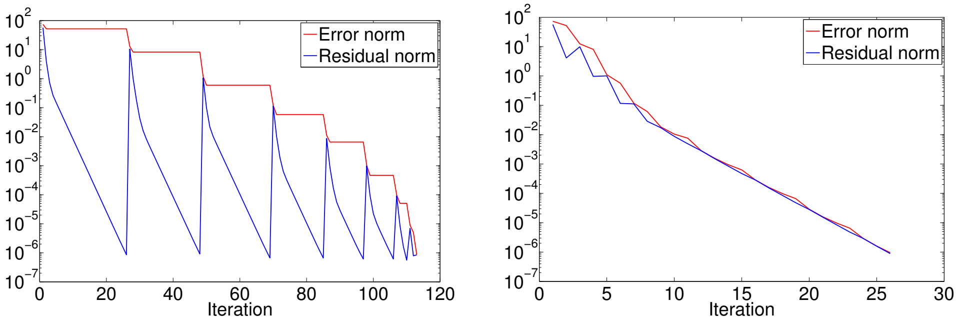


Figure 2.7: Error and residual norm histories for one of the realizations of permeability Set 1 from Table 2.2 over a single time step of $t^* = 0.4$. Shown on the left is the strategy where, at each non-linear stage, the fully converged linear solution is obtained. Shown on the right is the strategy where in each outer (Newton-Raphson) loop, the residual is reduced only by one order of magnitude.

optimized by employing adaptive updates of its multiscale components, i.e., the basis and (if considered) the correction functions. To this end, one has to monitor the changes in the entries of the transmissibility matrix $\mathbf{A}$ and RHS $\mathbf{f}$ between the iteration steps. Fig. 2.8(bottom-left) shows that the adaptive update of the C-AMS basis functions leads to a significant speed-up in terms of CPU time.

Furthermore, the two adaptivity methods (for linear system and local function updates) are combined and shown in Fig. 2.8(bottom-right). In this case, C-AMS will perform its iterations such that it exploits all adaptivity within the multiscale components, as well as the non-linearity in the flow equation. Note that, for this case, the compressible variant from Eq. (2.20) was used for both basis and correction functions. However, if the incompressible Eqs (2.22) and (2.23) are used, then the basis functions do not require any updates during iterations.

Finally, for this and all the following results (unless otherwise stated), the C-AMS coarsening ratio was chosen as $8 \times 8 \times 8$, because it was found efficient (see Subsection 2.3.1).

Choice of basis functions

The aim of this set of experiments is to study the effect of the different types of basis functions formulations on the C-AMS algorithm. The correction function is computed based on Eq. (2.20) in all cases (and, hence, updated adaptively with pressure), 20 iterations of ILU(0) are used for smoothing and all four possibilities for the basis functions, i.e., Eqs. (2.20)-(2.23), are considered. Finally, there is a single time step in the simulation, which takes the initial solution at time 0 ($p_0^* = 0$ everywhere) to the solution at time $t^* = 0.4$.

The total CPU time spent in each stage of the solver, as well as the number of iterations (given on top of each bar in Fig. 4.3), are measured. Also, the success rate of convergence is given inside parentheses beside the average number of iterations.

The results show that including compressibility in the basis functions does not

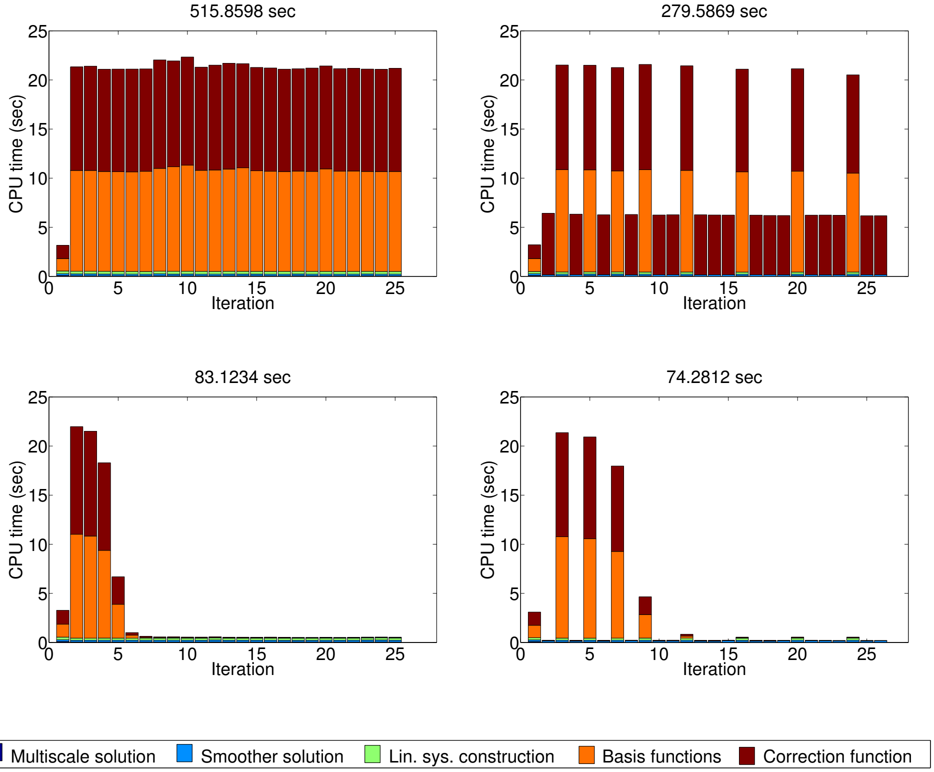


Figure 2.8: Effect of different types of adaptivity on the C-AMS performance for the permeability Set 1 from Table 2.2 after a time step of $t^* = 0.4$. Top-left: No adaptivity, top-right: Linear system update adaptivity only, bottom-left: Multiscale operator update adaptivity only, bottom-right: Fully adaptive, i.e in terms of both linear system and multiscale operator updates.

translate into faster convergence and, thus the additional CPU time required to adaptively update them is not justified. In fact, it is more efficient to use the incompressible (pressure independent) basis functions from Eqs. (2.22) and (2.23). Also, the inclusion of the accumulation term and the type of Restriction (MSFE or MSFV) does not play an important role for this patchy test case. Note that none of the choices results in 100% successful convergence, even though 20 ILU(0) smoothing iterations have been employed at each iteration. This can be attributed to the use of correction functions, as investigated in the next paragraph.

Choice of smoother

Note that none of the results from the previous test case (Fig. 4.3) has a 100% success rate. As described in [61], the CF is an independent stage in the AMS algorithm – its inclusion should be seen as an option and not a necessity for convergence. Fig. 2.10 presents the results of rerunning the previous experiment, this

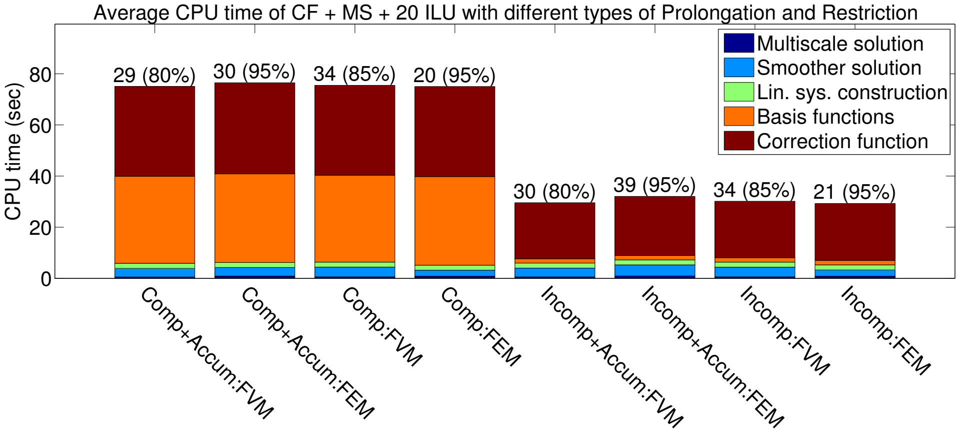


Figure 2.9: Effect of the choice of basis function on the C-AMS performance for the 64^3 grid-cell problem after a time step of $t^* = 0.4$. Results are averaged over 20 statistically-equivalent realizations. The number of iterations is shown on top of each bar. The success percentage is also shown in parentheses. Note that all simulations employ correction functions.

time varying the type of correction function. The plot confirms that eliminating the CF altogether leads to an overall speed-up, and, in addition, a convergence success rate of 100%. As described in [61], this can be explained by the sensitivity of CF to the heterogeneity of the permeability field, which leads to solver instability. Therefore, the CF should not be considered as candidate for the pre-smoothing stage in an efficient C-AMS procedure. Instead, ILU(0) is performed as post-smoother in order to resolve high-frequency errors.

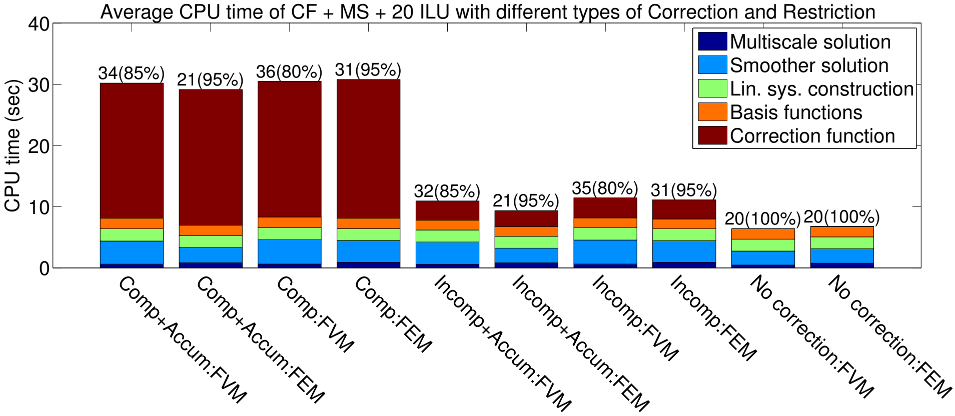


Figure 2.10: Effect of the choice of correction function on the CPU time of the multiscale solution on the permeability Set 1 from Table 2.2 after a time step of $t^* = 0.4$. The number of iterations is shown on top of each bar. Only the last 2 bars on the right correspond to runs in which no correction function was used (i.e., MS + 20 ILU).

Number of smoother iterations

Another variable in the C-AMS framework is the number of smoothing steps (here, ILU(0)) that should be applied in order to obtain the best trade-off between convergence rate and CPU time. The results of several experiments with the thus-far optimum C-AMS strategy (i.e., incompressible basis functions and no incorporation of CF) and various numbers of ILU applications are illustrated in Fig. 2.11. It is clear that with this C-AMS setup, an optimum scenario would be found with 5-10 ILU iterations per second-stage call. Note that all C-AMS runs during this experiment converged successfully.

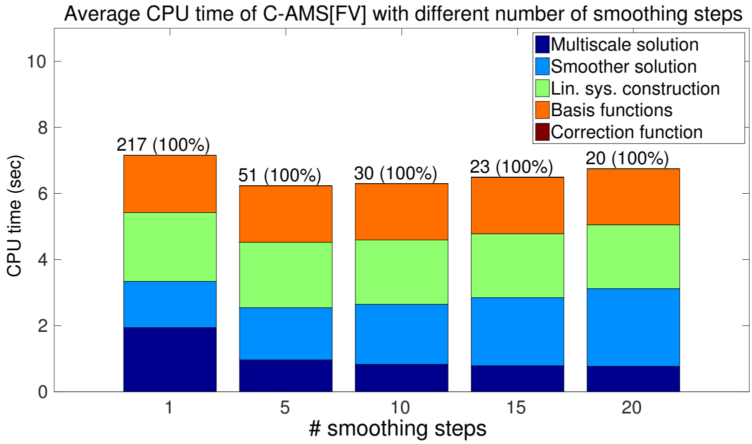


Figure 2.11: Effect of the number of ILU smoothing steps on the C-AMS[FV] performance for the permeability Set 1 from Table 2.2 (grid aspect ratio is 1) after a time step of $t^* = 0.4$. The number of iterations is shown on top of each bar, with convergence success rate inside parentheses. Note that excluding CF leads to 100% success rate for all scenarios.

Sensitivity to coarsening ratio

These experiments are meant to elicit the sensitivity of C-AMS towards the coarsening ratio used to construct the primal-coarse grid. The results are shown in Figs. 2.12-2.14 for the patchy fields of increasing domain size. Note that in both cases, the optimum overall CPU times were obtained with coarse-grid cells with the size of (approximately) the square-root of the domain length in each direction.

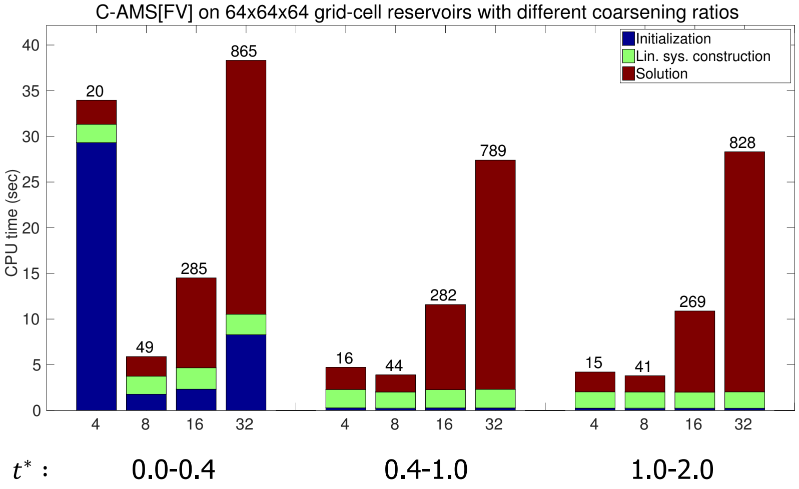


Figure 2.12: Patchy fields: Averaged CPU time (over 20 realizations) of C-AMS[FV] for different coarsening ratios for the permeability Set 1 from Table 2.2. Results support the use of coarsening ratio of 8^3. A similar behaviour was observed with the FE restriction operator.

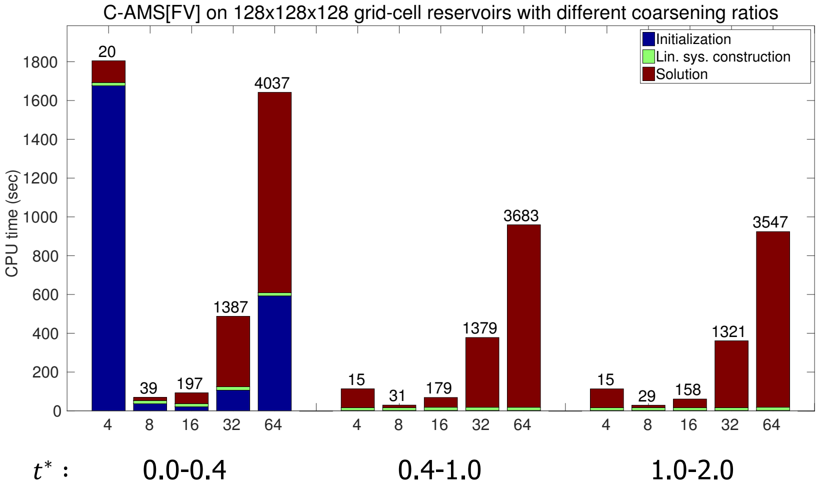


Figure 2.13: Patchy fields: Averaged CPU time (over 20 realizations) comparison of C-AMS [FV] for different coarsening ratios for the permeability Set 2 from Table 2.2. Results support the use of coarsening ratio of 8^3. A similar behaviour was observed with the FE restriction operator.

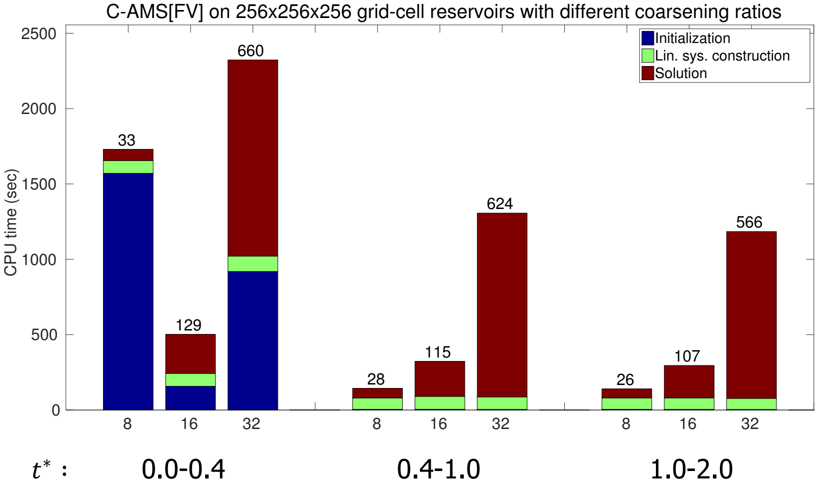


Figure 2.14: Patchy fields: Averaged CPU time (over 20 realizations) comparison of C-AMS[FV] for different coarsening ratios for the permeability Set 3 from Table 2.2. A coarsening ratio of 16^3 offers the best balance between initialization (basis function computation) and solution time, while 8^3 results in a more expensive initialization but faster convergence in subsequent time-steps. A similar behaviour was observed with the FE restriction operator.

2.3.2. C-AMS benchmark versus SAMG

Based on the previous results, the optimal C-AMS strategy includes a global multiscale stage using incompressible basis functions (Eq. (2.23)), accompanied by 5 iterations of ILU(0) for post-smoothing. In this subsection, C-AMS is compared against SAMG for three different sets of test cases: (1) the heterogeneous domains of different sizes from Table 2.2; (2): permeability Set 1 from Table 2.2 with stretched grids and line-source terms; and, (3): permeability Set 1 from Table 2.2 with different $ln(k)$ variances (i.e., permeability contrasts).

In all the presented experiments, SAMG is called to perform a single V-cycle, repeatedly in a Richardson loop. Its adaptivity is controlled manually, i.e., at the beginning of each Newton-Raphson outer iteration, SAMG is allowed to update its Galerkin operators. On the other hand, during linear iterations, SAMG is instructed to reuse its previous grids and operators. For the test cases considered here, this approach was found more efficient (by a factor in excess of 2) than the *automatic solver control* described in [45], In all other aspects, SAMG has been used as a black-box commercial solver.

Test case 1: heterogeneous domains of increasing size

In this subsection, C-AMS is compared against the SAMG algebraic multigrid solver for both patchy and layered permeability fields (Table 2.2) over 3 consecutive time steps. The time-lapse pressure solution for one patchy and one layered sample are shown in Fig. 2.15, illustrating the propagation of the pressure signal from the western face through the entire domain.

Figs. 2.16 and 2.17 show the number of iterations and CPU time at 3 consecutive non-dimensional times for problem Sets 1, 2, 4, and 5 from Table 2.2. Note that C-AMS with FV-based restriction operator did not converge in some of the test cases, while the FE-based variant achieved 100% success rate due to its SPD property. Therefore, an ideal solution strategy would use MSFE to converge to the desired level of accuracy and then employ a single MSFV sweep, in order to ensure mass conservation [99].

In addition, Figs. 2.18 illustrate CPU time (vertical axis) and the total number of iterations (on top of each column), for permeability Sets 3 and 6 from Table 2.2, with 8^3 and 16^3 coarsening ratios.

Note that, except for the first time-step, when all the basis functions are fully computed, C-AMS has a slight edge over SAMG, mainly due to its adaptivity and relatively inexpensive iterations. The initialization cost of C-AMS is particularly high in the 256^3 case, due to the large number of linear systems (solved with a direct solver) needed for the basis functions. It is clear from Fig. 2.18 that with larger primal-coarse blocks C-AMS requires less setup time, but more iterations to converge. Note that all performance studies presented in this work are for single-process computations.

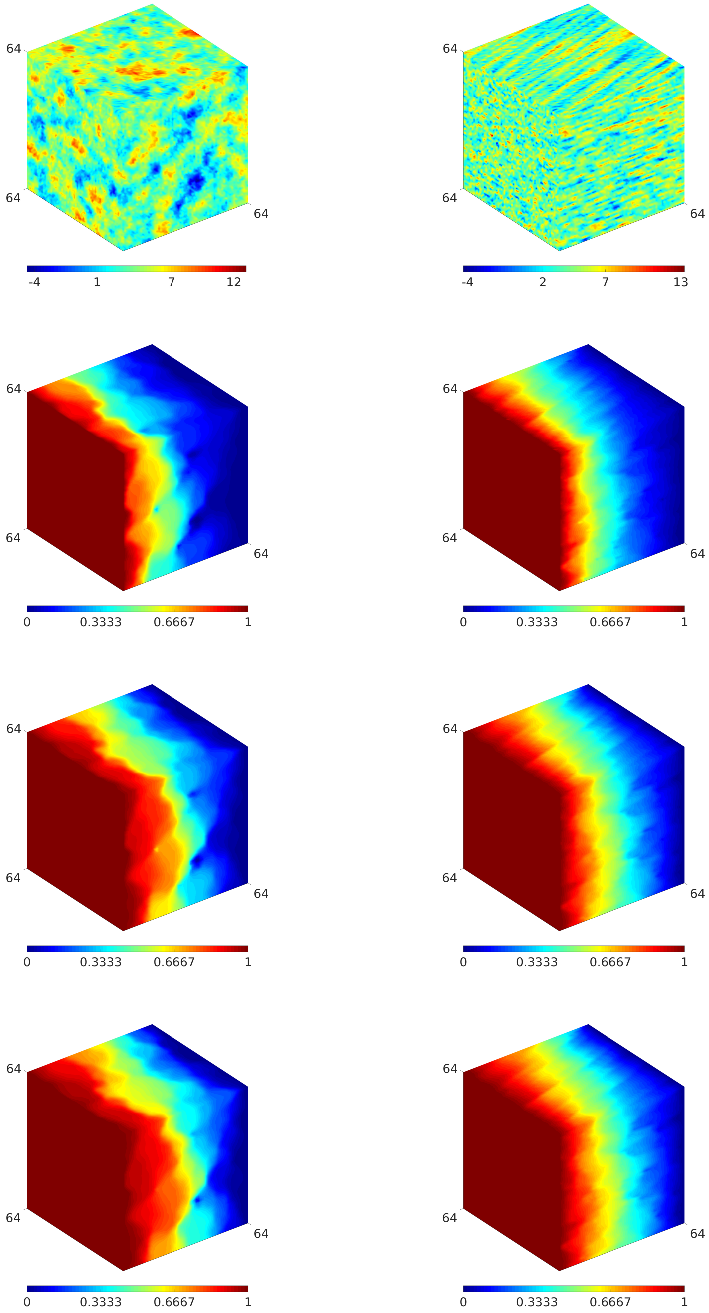


Figure 2.15: Pressure solution on one of the realizations of permeability Sets 1 (left) and 4 (right) from Table 2.2 at $t^* = 0.4, 1.0$, and 2.0 from top to bottom, respectively.

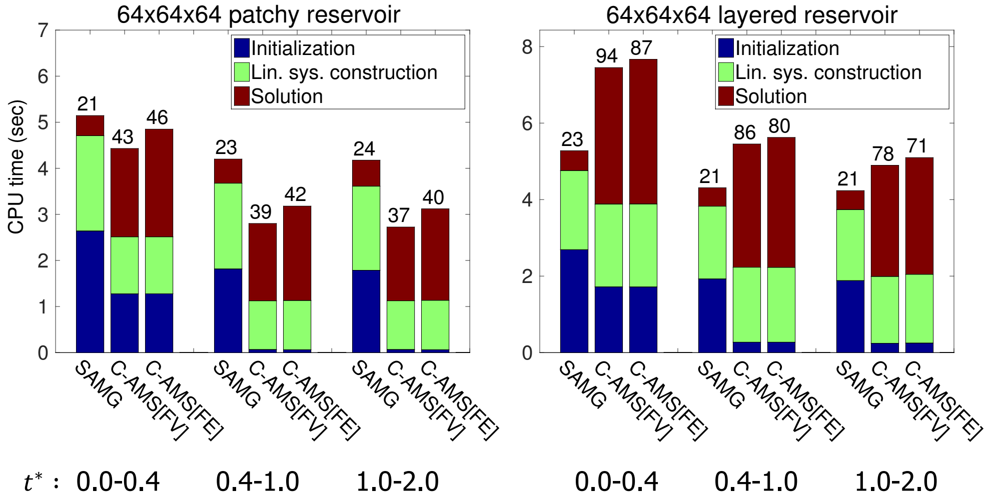


Figure 2.16: Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 1 (a) and 4 (b) from Table 2.2 over 3 successive time-steps. The coarsening ratio of C-AMS is 8^3. Moreover, C-AMS employs 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar.

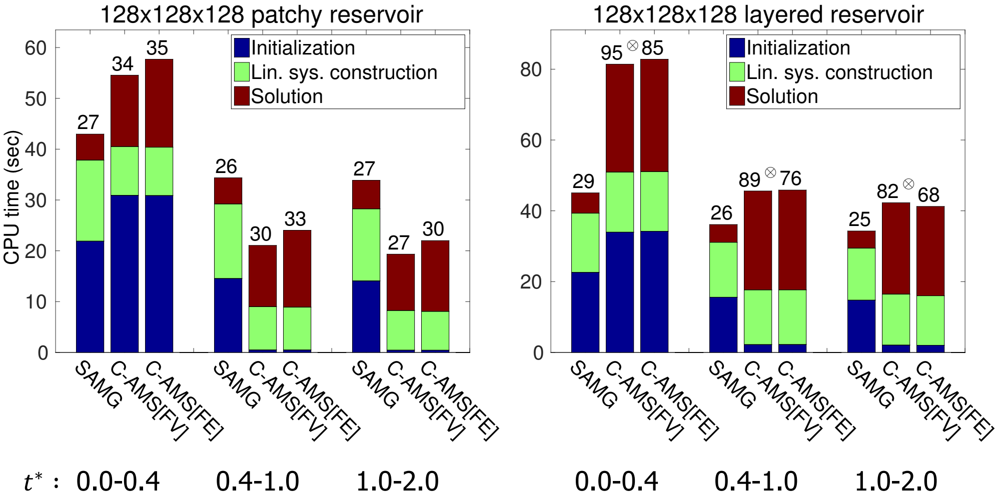


Figure 2.17: Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 2 (a) and 5 (b) from Table 2.2 over 3 successive time steps. C-AMS employs the coarsening ratio of 8^3, along with 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar. The ⊗ symbol signifies convergence success rate of 72% when FV-based restriction operator was employed.

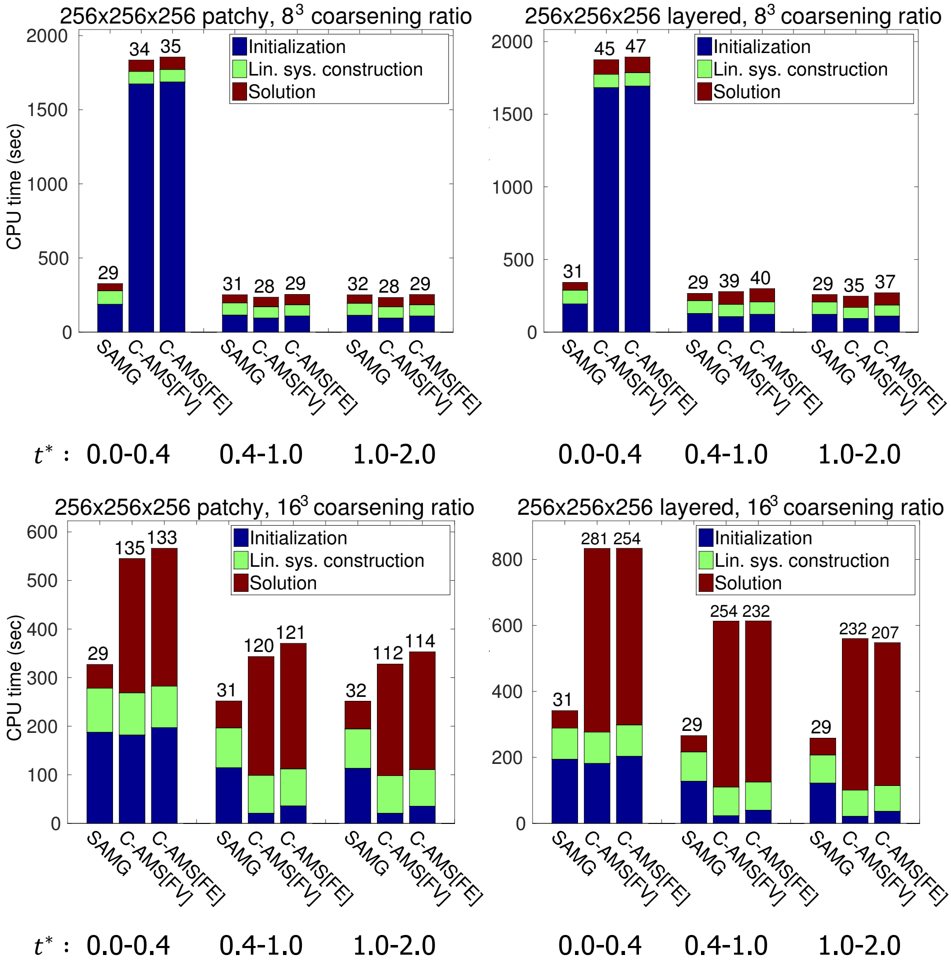


Figure 2.18: Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 3 (left column) and 6 (right column) from Table 2.2 over 3 successive time-steps. Different coarsening ratios of 8^3 (top row) and 16^3 (bottom row) are considered for C-AMS. Moreover, C-AMS employs 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar.

Since reservoir simulators are typically run for many time-steps, the high initialization time of C-AMS is outweighed by the efficiency gained in subsequent steps. Moreover, given the local support of the basis functions, this initialization can be greatly improved through parallel processing. Furthermore, only a few multiscale iterations may prove necessary to obtain an accurate approximation of the pressure solution in each time step for multi-phase flow problems.

Test Case 2: stretched grids with line-source terms

To study the effect of anisotropic permeability fields along with radial injection flow pattern, the permeability Set 1 from Table 2.2 is considered. The settings are all the same as previous test cases, except the following items. Dirichlet boundary conditions are set at the centers of two vertical sets of fine-scale grid cells: one from (1,1,1) to (1,1,64) and the other from (64,64,1) to (64,64,64) with the non-dimensional values of 1 and 0, respectively. In addition, grid aspect ratios of $\alpha = 1, 5,$ and 10 are considered (Note that $\Delta x/\alpha = \Delta y = \Delta z$). The non-dimensional time is calculated using αL as the characteristic length. Figure 2.19 illustrates the pressure solutions for one of the permeability realizations after the first time step $t^* = 0.4$.

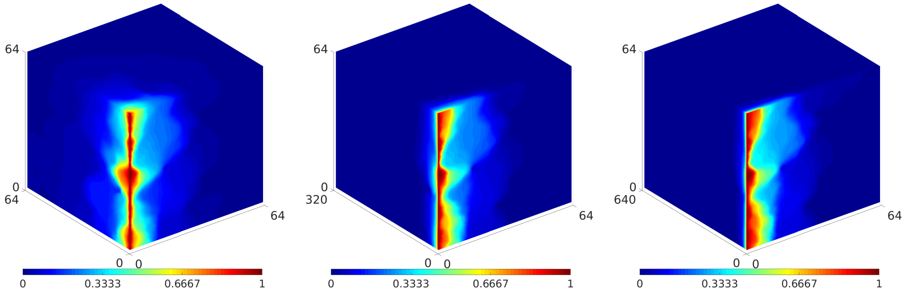


Figure 2.19: Converged pressure solution for one of the realizations of permeability Sets 1 with grid aspect ratio $\alpha = 1, 5,$ and 10, respectively from left to right, after one time step $t^* = 0.4$. Dirichlet boundary conditions are set at the centers of two vertical sets of fine-scale grid cells: one from (1,1,1) to (1,1,64) and the other from (64,64,1) to (64,64,64) with the values of 1 and 0, respectively.

The performance of C-AMS[FE] and SAMG are presented in Fig. 2.20. In contrast to C-AMS[FE], the C-AMS[FV] (not shown) did not lead to 100% convergence success. However, for those C-AMS[FV] successful runs, similar CPU times as in C-AMS[FE] were observed.

The results from in Fig. 2.20 are obtained with the C-AMS coarsening ratios of $8\times8\times8$, $2\times8\times8$, and $2\times8\times8$ for the cases of $\alpha = 1, 5$, and 10, respectively. Note that, as shown in Fig. 2.19, the anisotropic transmissibility (caused by the stretched grid effect) would further motivate the use of enhanced coarse-grid geometries for C-AMS. Such a strategy is well developed in the algebraic multigrid community, and is the subject of future studies.

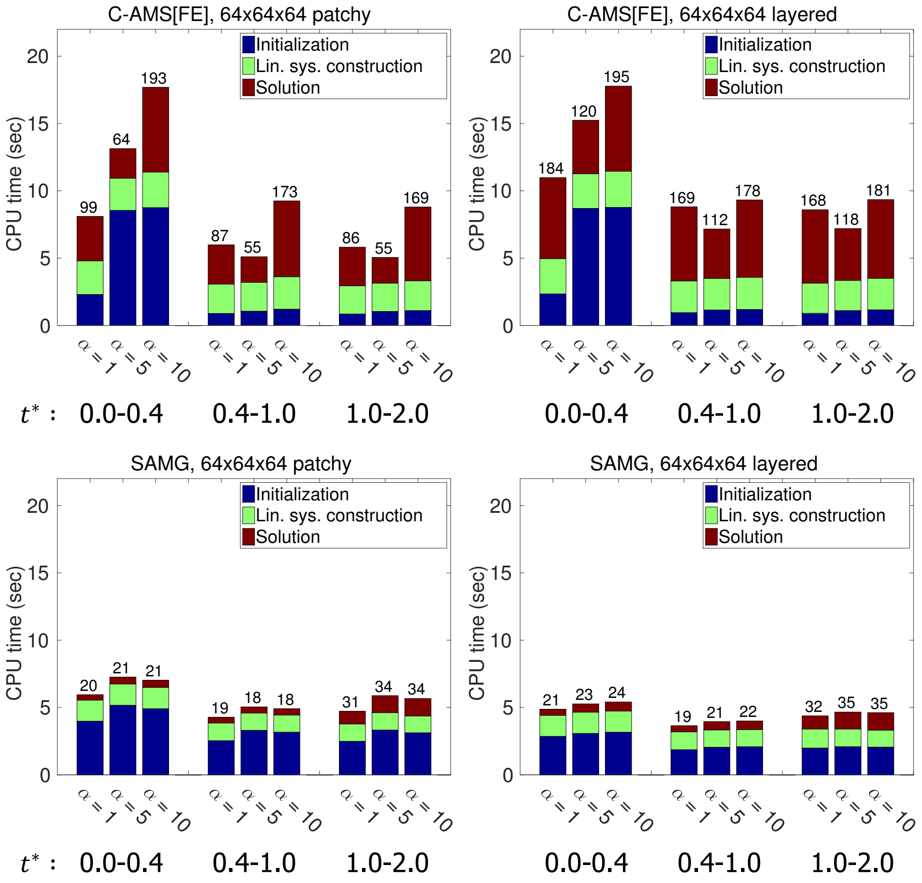


Figure 2.20: Performance of C-AMS (left column) and SAMG (right column) for permeability Set 1 from Table 2.2 for different grid aspect rations α in $\Delta x/\alpha = \Delta y = \Delta z$ for three successive time steps. Pressure solutions for the first time step is shown for one of the realizations in Fig. 2.19.

Test Case 3: effect of permeability contrast

To study the effect of permeability contrast, permeability Set 1 from Table 2.2 is considered with different $ln(k)$ variances, $\sigma = 2, 4$, and 8. Note that the so-far studied cases used a variance of 4, as described in Table 2.2.Similar settings as in the previous test cases are used, i.e., Dirichlet conditions are set at tthe east and west faces with no-flow condition everywhere else. Figure 2.21 illustrates the performance of C-AMS[FE] and SAMG. Note that the C-AMS requires more iterations when the permeability contrast is increased. To improve its performance, one can consider enriched multiscale strategies which are based on local spectral analysis [86] or a modified permeability field (with less contrast) for the calculation of basis functions [101]. Note that the success rates of C-AMS[FV] (not shown) were 90%

(patchy, $\sigma = 2$), 95% (patchy, $\sigma = 8$) and 40% (layered, $\sigma = 8$). For the successful runs, the CPU times of C-AMS[FV] were comparable with C-AMS[FE].

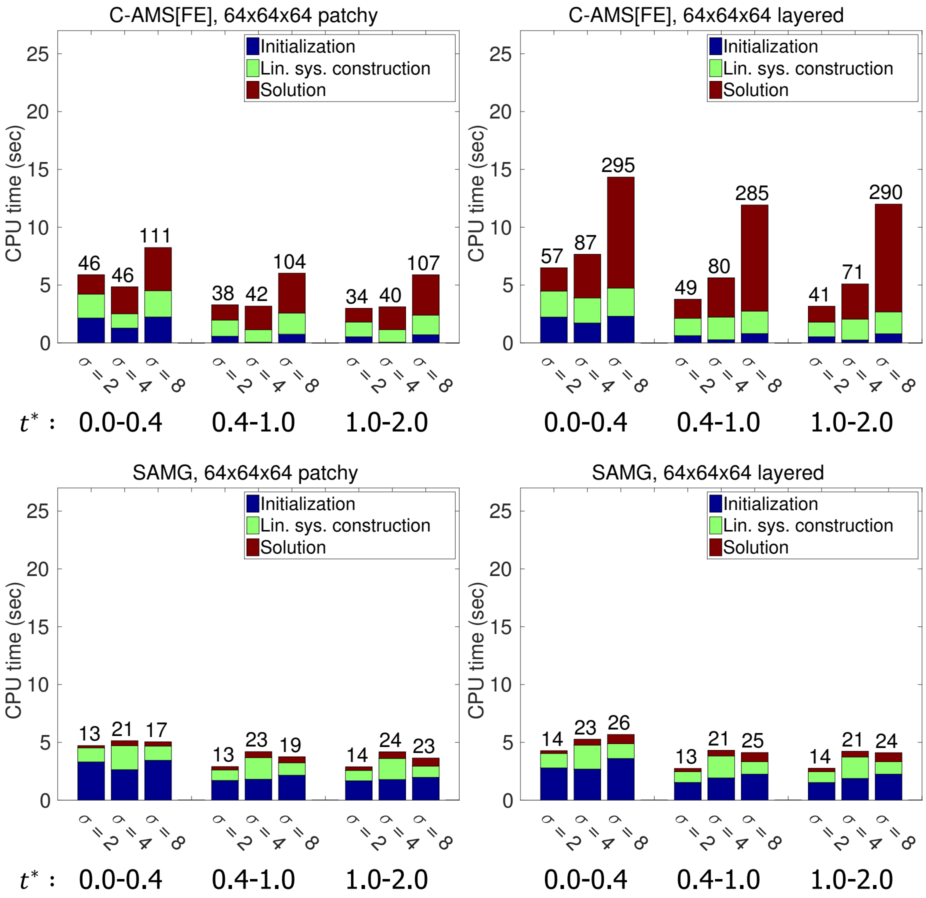


Figure 2.21: Averaged CPU time comparison between C-AMS (left) and SAMG (right) for permeability Set 1 from Table 2.2 for different $ln(k)$ variances of $\sigma = 2, 4$ and 5.

2.4. Conclusions

This work introduced an Algebraic MultiScale method for Compressible flows (C-AMS) in heterogeneous porous media. Its algebraic formulation benefits from adaptivity, both in terms of the infrequent updating of the linearized system and from the selective updating of the basis functions used to construct the prolongation operator.

The extensive numerical experiments on heterogeneous patchy and layered reservoirs revealed that the most efficient strategy is to use basis functions with incompressible advection terms, paired with 5 iterations of ILU(0) for post-smoothing.

Finally, several benchmark studies were presented, where the developed C-AMS research simulator was compared with an industrial-grade multigrid solver, i.e., SAMG. The results show that C-AMS is a competitive solver, especially in experiments that involve the simulation of a large number of time steps. The only drawback is the relatively high initialization time, which can be reduced by choosing an appropriate coarsening strategy or by running the basis function updates in parallel [62]. Moreover, due to its conservative property, C-AMS requires only a few iterations per time step to obtain a good quality approximation of the pressure solution. Systematic error estimate analyses for 3D multiphase simulations are a subject of ongoing research and, in addition, the C-AMS performance can be further extended by enrichment of the multiscale operators [86, 88, 102], and adaptive coarse grid geometries on the basis of the underlying fine-scale transmissibility.

3

Flow through fractured reservoirs

In many geoscientific applications, including hydrocarbon production and geothermal energy exploitation, the target formations are naturally fractured. More specifically, the highly heterogeneous matrix rock is crossed by several networks of lower-dimensional highly-conductive fractures at multiple length scales [105]. This raises important challenges for flow simulation, motivating the development of advanced modelling and numerical solution techniques.

Among the proposed methods, the hierarchical fracture modelling approach allows for avoiding complexities associated with the discretization and dynamic nature of fracture geometries [18]. In this approach, small-scale fractures (below the matrix grid resolution) are homogenized within the matrix rock, altering its effective permeability [20]. The remaining fractures are then represented as explicit control volumes [19, 27]. If the fracture and matrix grids are generated independently, then the formation is said to be discretized according to the Embedded Discrete Fracture Model (EDFM) [106]. Alternatively, the fracture cells can be constrained to lie at the interfaces of matrix cells, i.e. by employing Discrete Fracture Modelling methods (DFM), which often require unstructured grids [24]. Both DFM and EDFM have been applied to reservoirs with complex fracture geometries [107] and fluid physics [25, 108]. Recent developments include higher-order approximation schemes within finite-volume [109] and finite-element [110] methods.

Note that the total number of degrees of freedom (DOF), even after homogenizing small-scale fractures, is beyond the scope of classical simulation methods. This motivates the development of an efficient multiscale method for heterogeneous fractured porous media.

The material presented in this chapter has been published in the proceedings of the SPE Reservoir Simulation Symposium (2015) [103] and in the Journal of Computational Physics **321**, 819 (2016) [104].

Multiscale finite element (MSFE) [46] and finite volume (MSFV) methods [42] have been introduced and evolved mainly for heterogeneous, but non-fractured, porous media (see [98] for a comparison between the two flavours) . The first application of MSFV methods to fractured reservoirs was developed on the basis of coupling the matrix pressure to the average pressure in each fracture network, at coarse scale [27]. Based on the presented results for 2D problems, the method was efficient for media with highly conductive fracture networks which span short spatial length scales (relative to that of the domain). However, convergence was observed to degrade for test cases with significant variations in the pressure distribution along the fracture network.

In combination with streamline-based mixed formulations, multiscale methods have also been employed to 2D fractured reservoirs [94]. More recently, a multiscale approach was developed for 2D reservoirs which assigned one coarse node at each fracture intersection only, with no coarse nodes in the matrix [111]. Note that none of these methods include 3D heterogeneous reservoirs nor has their performance been benchmarked against a commercial linear solver. More importantly, the literature is lacking a multiscale method which allows for flexible coarse grids inside the matrix as well as its embedded fractures and, thus, able to accommodate heterogeneous cases with fracture networks of different length scales.

This chapter presents the development of an Algebraic MultiScale method for heterogeneous Fractured porous media (F-AMS) using EDFM. Given a partition of the fine-scale cells into primal and dual-coarse blocks for both the matrix and fracture networks, the algorithm algebraically constructs the multiscale prolongation (mapping coarse- to fine-scale) and restriction (mapping fine- to coarse-scale) operators. The columns of the prolongation operator are the local basis functions, solved on dual-coarse cells, for both matrix and fractures.

F-AMS supports four different matrix-fracture coupling strategies, at the coarse-scale. First, the Decoupled-AMS basis functions are defined by neglecting the contribution of a medium's coarse solution (e.g., fractures) in the interpolated solution in the other (e.g., matrix), thus preserving sparsity in the resulting coarse-scale system. Then, two semi-coupled (one-way) strategies, Rock-AMS and Frac-AMS, are considered. The Rock-AMS approach constructs a prolongation operator in which the matrix coarse solutions also contribute in computing the interpolated fine-scale solution in neighbouring fractures. Similarly, Frac-AMS considers the influence of the fracture coarse solution when interpolating the pressure inside the surrounding porous rock. Finally, the fully coupled strategy, Coupled-AMS, is devised, where coarse-scale solutions from both media play a role in finding the fine-scale solution of each other. This last approach, although allowing for full fracture-matrix coupling, leads to a dense coarse-scale system with additional overhead during the associated algebraic (matrix-vector, matrix-matrix) operations. As such, for practical applications, the Coupled-AMS prolongation operator may require tuning via truncation, where values below a specified threshold are algebraically deleted, followed by a rescaling step, to maintain partition of unity. This option is also investigated.

To summarize, F-AMS allows for arbitrary coarse grid resolutions in both fractures and matrix, as well as a variety of coupling strategies, at the coarse-scale.

Furthermore, once the coarse grids are defined, the F-AMS procedure is formulated and implemented in algebraic form, in line with the previously published formulations of incompressible (AMS) [61] and compressible (C-AMS) [96] flows. In the limit, if the Frac-AMS coupling strategy and only one coarse node per fracture network is used, F-AMS automatically reduces to the method previously described in [27]. However, this setup proves inefficient for many of the test cases in this work. From a bottom-up perspective, F-AMS extends the AMS prolongation operator, as previously described in [61], with additional columns. Some of these columns correspond to the enriched fracture coarse domain. The remainder represent local well basis functions that capture the effect of source terms, similar to [68].

In order to test the F-AMS method, a proof-of-concept implementation was developed, aimed at reservoirs defined on 3D structured grids with embedded vertical fracture plates. For the presented experiments, a finite-element (FE) restriction operator is employed to obtain a symmetric-positive-definite coarse system. If, for efficiency reasons, F-AMS is stopped before the solution reaches machine precision, then a finite-volume (FV) restriction operator is employed at the end, followed by a mass-conservative reconstruction of the fine-scale flux field [42]. The performance of F-AMS was tested using the in-house object oriented serial-processing simulator, DARSim 1, and both CPU times and convergence rates were measured and compared against the commercial Algebraic MultiGrid (AMG) solver, SAMG [45].

Numerical test cases are considered in order to study the effect of the different components of the algorithm, namely, the coarsening ratios and basis function coupling strategies. The results of these systematic studies show that only a few DOF per fracture network are necessary to obtain a good trade-off between convergence rate and computational expense. In conclusion, F-AMS is an efficient and scalable method for solving flow in heterogeneous and naturally fractured porous media. Its development marks an important step forward towards the integration of multiscale methods as "black-box" pressure solvers within existing reservoir simulators, with the possibility of extension to more complex physics.

The chapter is structured as follows. First, the EDFM fine-scale discrete system is described in Section 3.1. Then, the components of the F-AMS algorithm are detailed in Section 3.2. Section 3.3 consists of numerical results for both 2D and 3D test cases. Finally, conclusions and remarks are the subject of Section 3.4.

3.1. Governing equations

The mass-conservation equations for isothermal single-phase flow in fractured media, using Darcy's law and neglecting compositional effects, can be written as

$$\left[\frac{\partial(\phi\rho)}{\partial t} - \nabla\,.\,(\rho\mathbf{k}\lambda\cdot\nabla p)\right]^{m} = [\rho q]^{mw} + [\rho Q]^{m} + [\rho q]^{mf} \quad \text{on } \Omega^m \subset \mathbb{R}^n \qquad (3.1)$$

for the matrix (superscript ^{m}) and

$$\left[\frac{\partial(\phi\rho)}{\partial t} - \nabla\,.\,(\rho\mathbf{k}\lambda\cdot\nabla p)\right]^{f} = [\rho q]^{fw} + [\rho Q]^{f} + [\rho q]^{fm} \quad \text{on } \Omega^f \subset \mathbb{R}^{n-1} \qquad (3.2)$$

for the fracture (superscript ^{f}) spatial domains. Here, $\lambda = 1/\mu$ is the mobility, μ is the fluid viscosity and $\mathbf{k}^*$ denotes the fracture ($\mathbf{k}^* = k^f$) or rock ($\mathbf{k}^* = k^m$) absolute 2nd order positive definite permeability tensor, here, considered as diagonal and isotropic. Note that the latter can also account for the homogenized small-scale fractures, as described in the hierarchical fracture model [19, 20]. Also, ρ and ϕ are the fluid density and rock porosity, respectively. Here, for simplicity, fracture porosity is always considered to be 1. The q^{mw} and q^{fw} denote the matrix and fracture external source terms, respectively, i.e. from injection/production wells. For a perforated matrix control volume V, it reads

$$q_V^{mw} \;=\; \text{PI}\;\mathbf{k}^m\lambda\;(p^w - p^m)/V \;\equiv\; \beta^m\;(p^w - p^m), \tag{3.3}$$

where $\beta^m = \text{PI}\;\mathbf{k}^m\lambda/V$, $q_{tot}^{mw} = \int_V q_V^{mw}\,dV$ is the total injection rate, and PI is the productivity index [112]. Similarly, the fracture-matrix coupling terms are modelled such that for a matrix volume V intersecting with a fracture surface A one obtains

$$q_V^{mf} \;=\; \text{CI}\;\mathbf{k}^{f-m}\lambda\;(p^f - p^m)/V \;\equiv\; \eta^m\;(p^f - p^m) \tag{3.4}$$

and

$$q_A^{fm} \;=\; \text{CI}\;\mathbf{k}^{f-m}\lambda\;(p^m - p^f)/A \;\equiv\; \eta^f\;(p^m - p^f), \tag{3.5}$$

where $\eta^m = \text{CI}\;\mathbf{k}^{f-m}\lambda/V$ and $\eta^f = \text{CI}\;\mathbf{k}^{f-m}\lambda/A$. This ensures the total flux between a fracture element of area A and a matrix element of volume V is equal [27], i.e.,

$$\int_V q_V^{mf}\,dV = -\int_A q_A^{fm}\,dA. \tag{3.6}$$

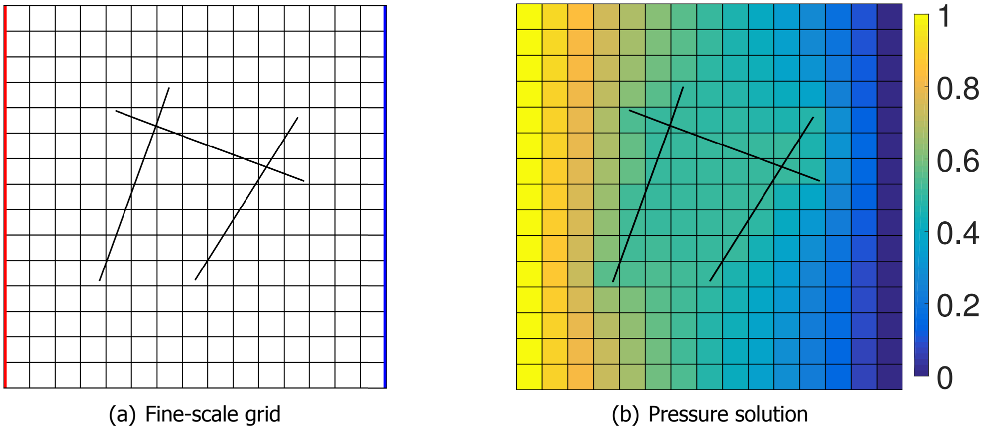
The $\mathbf{k}^{f-m}$ is the effective permeability at the interface between the fractures and their surrounding matrix (i.e. harmonically averaged). The CI is the connectivity index, defined on a discrete system as

$$\text{CI}_{ij} = \frac{A_{ij}}{\langle d\rangle_{ij}}, \tag{3.7}$$

where A_{ij} is the area fraction of fracture element i overlapping with the matrix element j, and $\langle d\rangle_{ij}$ is the average distance of the two elements (see Appendix A and [27] for its detailed computation). Finally, the Q^m and Q^f terms describe other external source terms for matrix and fractures (e.g., gravity or capillarity terms, neglected in this work).

These equations are to be solved for matrix and fracture pressures, p^m and p^f, on the matrix Ω^m and fracture Ω^f domains, as depicted in Fig. 3.1. Note that a fracture network can consist of several fractures, which are represented in a lower dimensional space, i.e. $\Omega^f \subset \mathbb{R}^{n-1}$, than the matrix (reservoir rock) $\Omega^m \subset \mathbb{R}^n$. The main advantage of this type of formulation is that the matrix and fracture grids are independent and, thus, can be freely adapted to accommodate the appropriate physics for each medium. This is especially important in highly fractured reservoirs or when fractures are generated (and closed) during simulation, e.g., in geothermal formations [113].

The incompressible single-phase pressure solution obtained using the EDFM approach for a 2D fractured reservoir model, shown in Fig. 3.1(a), is provided in Fig. 3.1(b). Two pressure-constrained wells are placed on the East and West boundaries, and the reservoir rock is homogeneous.



(a) Fine-scale grid

(b) Pressure solution

Figure 3.1: Illustration of a 2D fine-scale computational grid (a) which contains 15×15 homogeneous matrix, 21 fracture cells and two pressure-constrained wells at the West and East boundaries with values of 1 and 0, respectively. The pressure solution is plotted in (b), where fractures are 100 times more conductive than the matrix.

When non-linearities are present (e.g., compressible flows), the flow equations need to first be linearized (see Section 2.1.2), leading to the following system

$$\mathbf{A}^{(\nu)}\overline{p}^{(\nu+1)} \equiv \begin{bmatrix} \mathbf{A}^{mm} & \mathbf{A}^{mf} & \mathbf{A}^{mw} \\ \mathbf{A}^{fm} & \mathbf{A}^{ff} & \mathbf{A}^{fw} \\ \mathbf{A}^{wm} & \mathbf{A}^{wf} & \mathbf{A}^{ww} \end{bmatrix}^{(\nu)} \begin{bmatrix} \overline{p}^{m} \\ \overline{p}^{f} \\ \overline{p}^{w} \end{bmatrix}^{(\nu+1)} = \begin{bmatrix} \overline{q}^{m} \\ \overline{q}^{f} \\ \overline{q}^{w} \end{bmatrix}^{(\nu)} \equiv \overline{q}^{(\nu)}, \qquad (3.8)$$

which is solved iteratively, in a Newton-Raphson loop, until the converged solution is achieved. Note that this system (3.8) shows an implicit treatment of the coupling between fracture and matrix through the $\mathbf{A}^{fm}$ entries, and that $\mathbf{A}^{(\nu)}$ can be non-symmetric, due to the compressibility effects [43, 96].

Developing an efficient solution strategy for the linearized system (3.8) is quite challenging for several reasons. On the one hand, the size of this system can exceed several millions of unknowns for realistic test cases. On the other hand, the value of the condition numbers for the system matrix is worsened by high contrasts between reservoir properties (matrix permeability is highly heterogeneous over large scales, fractures are typically much more conductive than the matrix, etc.).

Clearly a classical upscaling method cannot be employed here due to the highly resolved fractures, which play an important role in mass transport. This creates a niche for conservative multiscale methods, which have the important advantage of solving coarse-scale problems while honouring fine-scale data [52, 56] in an iterative error reduction strategy [59, 74, 101] which allows for mass-conservative flux reconstruction at any stage [55]. Next, the development of the F-AMS method is presented.

3.2. Algebraic multiscale formulation (F-AMS)

This section describes the F-AMS procedure, an efficient multiscale solution strategy for Eq. (3.8). Given a computational domain with N_f fracture networks and N_w wells, F-AMS first superimposes two coarse grids on top of both the matrix and fracture domains. The primal-coarse grid is a non-overlapping decomposition of the domain, inside which a fine-scale cell is selected as coarse node (Fig. 3.2(a) for 2D, and 3.2(c) and 3.2(e) for 3D cases). By connecting the coarse nodes, a secondary overlapping coarse grid is obtained, which is called the dual coarse grid (Figs. 4.3(a), 3.2(d) and 3.2(f)). There exist N_{cm} and N_{dm} matrix primal-coarse and dual-coarse blocks and, similarly, each fracture network f_i contains N_{cf_i} and N_{df_i} fracture primal-coarse and dual-coarse blocks. Note that N_w (injection or production) wells exist in the domain, as driving forces for the flow.

F-AMS approximates the solution to Eq. (3.8), $\overline{p}$, as a superposition of coarse-scale solutions ($\breve{p}$) using locally computed basis functions (Φ), i.e.

$$\overline{p}^m \approx \overline{p}'^m = \sum_{i=1}^{N_{cm}} \overline{\Phi}_i^{mm} \breve{p}_i^m + \sum_{i=1}^{N_f} \sum_{j=1}^{N_{cf_i}} \overline{\Phi}_j^{mf_i} \breve{p}_j^{f_i} + \sum_{k=1}^{N_w} \overline{\Phi}_j^{mw} \breve{p}_k^w, \tag{3.9}$$

for the matrix and

$$\overline{p}^f \approx \overline{p}'^f = \sum_{i=1}^{N_{cm}} \overline{\Phi}_i^{fm} \breve{p}_i^m + \sum_{i=1}^{N_f} \sum_{j=1}^{N_{cf_i}} \overline{\Phi}_j^{ff_i} \breve{p}_j^{f_i} + \sum_{k=1}^{N_w} \overline{\Phi}_j^{fw} \breve{p}_k^w \tag{3.10}$$

for the fractures, respectively. The basis functions associated with matrix coarse cells (i.e., $\overline{\Phi}^{m*}$) are $\overline{\Phi}_i^{mm}$ for matrix-matrix effects, $\overline{\Phi}_j^{mf}$ for the matrix-fracture coupling, and $\overline{\Phi}_j^{mw}$ matrix-well interactions. These basis functions (interpolators) are employed in order to capture the effects of all the important factors (matrix, fractures, and wells) in the construction of a good approximation for the matrix pressure field p^m. Similarly for fractures, $\overline{\Phi}^{f*}$ consists of the contributions from the matrix $\overline{\Phi}^{fm}$, fractures Φ^{ff}, and wells $\overline{\Phi}^{fw}$, if present.

One of the novel aspects of this work is that the pressure field inside fractures, $\overline{p}^f$, is included explicitly in the multiscale formulation (Eq. (4.13)). This means that the fracture grid cells are also decomposed into primal and dual coarse blocks (Fig. 3.2), similar to the matrix. Their solutions are also mapped to the coarse scale and back to the original resolution, again, similar to the matrix. More specifically, each fracture network f_i is decomposed into N_{cf_i} primal-coarse grid blocks, for which sets of basis functions are calculated. One could employ the same formulation for wells, i.e., discretize them into several fine-scale cells which can then be coarsened on the superimposed multiscale grids. However, for the sake of simplicity, in the experiments presented in this work, each well is assigned a single fine-scale DOF, which is mapped to the coarse-scale using the identity restriction operator, i.e.

$$p^{w_i} = p'^{w_i} = \breve{p}^{w_i} \qquad \forall i \in \{1, \dots, N_w\}. \tag{3.11}$$

In algebraic notation, the superpositions (4.12) and (4.13) can be expressed as

$$\overline{p}^m \approx \overline{p}'^m = \mathcal{P}^m \breve{p} \equiv [\mathcal{P}^{mm} \;\; \mathcal{P}^{mf} \;\; \mathcal{P}^{mw}]\, [\breve{p}^m \;\; \breve{p}^f \;\; \breve{p}^w]^T \tag{3.12}$$

and

$$\overline{p}^f \approx \overline{p}'^f = \mathcal{P}^f \breve{p} \equiv [\mathcal{P}^{fm} \;\; \mathcal{P}^{ff} \;\; \mathcal{P}^{fw}]\, [\breve{p}^m \;\; \breve{p}^f \;\; \breve{p}^w]^T, \tag{3.13}$$

respectively.

The basis functions are assembled in the columns of the multiscale *prolongation operator*, $\mathcal{P}$, with the dimension of $N_{\text{fine}} \times N_{\text{coarse}}$, where N_{fine} and N_{coarse} are the total number of fine- and coarse-scale control volumes, respectively. The part of $\mathcal{P}$ corresponding to the matrix fine-cells is defined as

$$\mathcal{P}^m = \left[\begin{array}{c|ccc|c} \overline{\Phi}_1^{mm} \dots \overline{\Phi}_{N_{cm}}^{mm} & \overline{\Phi}_1^{mf_1} \dots \overline{\Phi}_{N_{cf_1}}^{mf_1} & \dots & \overline{\Phi}_1^{mf_{N_f}} \dots \overline{\Phi}_{N_{cf_{N_f}}}^{mf_{N_f}} & \overline{\Phi}_1^{mw} \dots \overline{\Phi}_{N_w}^{mw} \end{array}\right]. \tag{3.14}$$

Note the three sub-blocks which represent matrix-matrix, matrix-fracture, and matrix-well coupling. Similarly, the prolongation operator for fractures can be stated as

$$\mathcal{P}^f = \left[\begin{array}{c|ccc|c} \overline{\Phi}_1^{fm} \dots \overline{\Phi}_{N_{cm}}^{fm} & \overline{\Phi}_1^{ff_1} \dots \overline{\Phi}_{N_{cf_1}}^{ff_1} & \dots & \overline{\Phi}_1^{ff_{N_f}} \dots \overline{\Phi}_{N_{cf_{N_f}}}^{ff_{N_f}} & \overline{\Phi}_1^{fw} \dots \overline{\Phi}_{N_w}^{fw} \end{array}\right]. \tag{3.15}$$

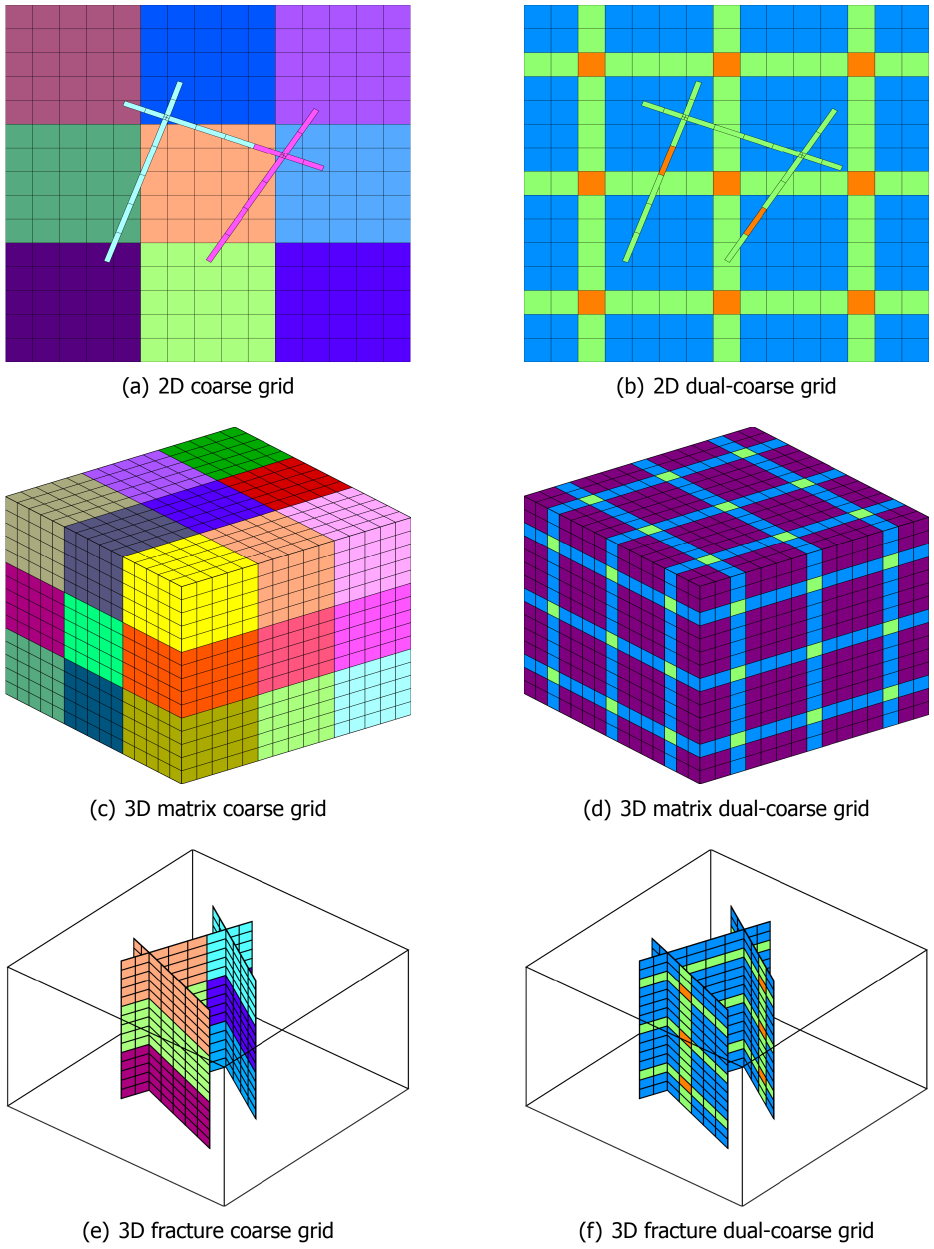
Algebraically, the complete F-AMS prolongation operator reads

$$\mathcal{P} = \begin{bmatrix} \mathcal{P}^m \\ \mathcal{P}^f \\ \mathcal{P}^w \end{bmatrix} = \begin{bmatrix} \mathcal{P}^{mm} & \mathcal{P}^{mf} & \mathcal{P}^{mw} \\ \mathcal{P}^{fm} & \mathcal{P}^{ff} & \mathcal{P}^{fw} \\ \mathcal{P}^{wm} & \mathcal{P}^{wf} & \mathcal{P}^{ww} \end{bmatrix}, \tag{3.16}$$

where $\mathcal{P}^{wm}$ and $\mathcal{P}^{wf}$ are set to zero, while $\mathcal{P}^{ww}$ is the identity matrix.

Note that the prolongation operator, as described in Eq. (3.16), allows full flexibility in consideration of the fracture-matrix coupling in the interpolated solution, i.e. via the values in $\mathcal{P}^{mf}$ and $\mathcal{P}^{fm}$. This leads to the definition of four operators, differentiated by the coupling strategy they employ:

1. **Decoupled-AMS:** $\mathcal{P}^{mf} = 0$ and $\mathcal{P}^{fm} = 0$
2. **Frac-AMS:** only $\mathcal{P}^{fm} = 0$.
3. **Rock-AMS:** only $\mathcal{P}^{mf} = 0$.
4. **Coupled-AMS:** $\mathcal{P}^{mf}$ and $\mathcal{P}^{fm}$ both non-zero.



(a) 2D coarse grid

(b) 2D dual-coarse grid

(c) 3D matrix coarse grid

(d) 3D matrix dual-coarse grid

(e) 3D fracture coarse grid

(f) 3D fracture dual-coarse grid

Figure 3.2: F-AMS coarse grids defined on 2D (top) and 3D domains (middle and bottom). The primal grid (left) consists of non-overlapping coarse blocks, each shown in a different colour. The dual grid (right) is split into nodes, shown in orange, 1D blocks (edges) in green, 2D blocks (faces) in blue and 3D blocks (interiors) in purple. Note that the fracture aperture in (a) and (b) is magnified for clarity.

The first option, i.e., Decoupled-AMS, constructs the most sparse $\mathcal{P}$ and thus has an efficient setup phase. The fourth option, i.e., Coupled-AMS, can lead to more accurate multiscale simulations, however, it can severely increase the density of the operators. In such cases, one may be able to obtain a trade-off between the quality of the prolongation operator and its sparsity via truncation, followed by a rescaling of the rows to ensure partition of unity. A CPU-based study considering the overhead introduced by the density of $\mathcal{P}$ is presented in Section 3.3.

In order to construct the coarse-scale system, F-AMS also needs the specification of a restriction operator, which is a map from fine- to coarse-scale (dimension $N_{\text{coarse}} \times N_{\text{fine}}$). Due to its algebraic formulation, F-AMS can accommodate multiscale finite volume (MSFV), multiscale finite element (MSFE) or even a hybrid multiscale finite element and volume restriction (MSMIX). More specifically, the MSMIX employs a FV-based restriction for part of the domain (e.g., fractures or wells), and FE for the rest (e.g., matrix rock). It is important to note that, after any MSFV stage, it is possible to construct a mass-conservative flux field for both matrix and fractures. As such, in multiphase simulations, if iterations for pressure Eq. (3.8) are stopped before full convergence is achieved, MSFV needs to be employed before solving transport equations. On the other hand, MSFE leads to a symmetric-positive-definite (SPD) coarse system if the fine-scale system matrix is also SPD [61, 96], and is the option used during all numerical experiments presented in this work. Note that MSMIX can be tweaked to achieve the desired compromise between MSFV and MSFE.

In the following sub-sections, first, the formulation of the local basis functions is explained. Then, the F-AMS system, and finally the simulation strategy is described in detail.

3.2.1. Basis function formulations

As stated before, F-AMS constructs a non-overlapping partition on the given fine-scale computational domain for both matrix rock and fracture cells, i.e. the primal-coarse grid. Then, by connecting the coarse nodes, the overlapping decomposition of the domain, i.e., dual-coarse grid, is obtained. Following the original description of the MSFV basis functions [42, 55] and its algebraic description [53, 61, 96], local basis functions are calculated for each coarse node i by respecting the *wirebasket* hierarchy [63]. First the value of basis function i in each coarse node j (also called vertices, shown in orange in Fig. 4.3(a)) is set according to the Kroenecker delta, δ_{ij}, i.e. 1 when $i = j$ and 0 otherwise. Then the dual blocks in the neighbourhood of node i are resolved, in sequence, as follows: first all the 1D dual blocks (or edges, shown in green in Fig. 4.3(a)), followed by the 2D (or faces, shown in blue in Fig. 4.3(a)) and, finally, the 3D dual blocks (or interiors, shown in purple in Fig. 3.2(d)). The fact that each dual element (e.g. edge) neglects the transmissibilities to neighbouring cells belonging to elements of inferior rank in the wirebasket hierarchy (i.e. faces and interiors), constitutes the *localization assumption* [61], which ensures that each basis function has a limited support.

The values obtained in the manner described above, for each coarse node i, are assembled in column i of the prolongation operator to form basis function $\overline{\Phi}_i$.

It is important to note that by having independent fine-scale grids for each media, a matrix cell (say from a face block) can be directly connected to fracture cells belonging to dual blocks of any type (vertex, edge or face). This is an important difference from unfractured media, where the two-point flux approximation (TPFA) stencil on structured grids ensured that any dual grid element would have connections only to blocks of directly superior or inferior rank in the wirebasket hierarchy (e.g., the external neighbours of a face cell are either edge or interior cells). As such, the multscale localization assumption needs to be extended to account for the connection between the two media. In the F-AMS framework, this leads to the formulation of basis functions which account for different degrees of coupling between the matrix and its perforating fractures.

In order to provide a compact definition of the various basis functions (matrix, fracture, well), paired with one of the four different coupling strategies considered (Decoupled-AMS, Rock-AMS, Frac-AMS, Coupled-AMS), the following "skeleton" is introduced

$$-\nabla \,.\, (\mathbf{k}^*\lambda \cdot \nabla\Phi^{*\bullet}) + \sum_{j\in \text{conn}^*_{mf}} \eta^*_j \;\xi(\Phi^{*\bullet}) + \sum_{j\in \text{perf}^*_w} \beta^*_j \;(\Phi^{*\bullet} - \Phi^{w\bullet}) = 0, \tag{3.17}$$

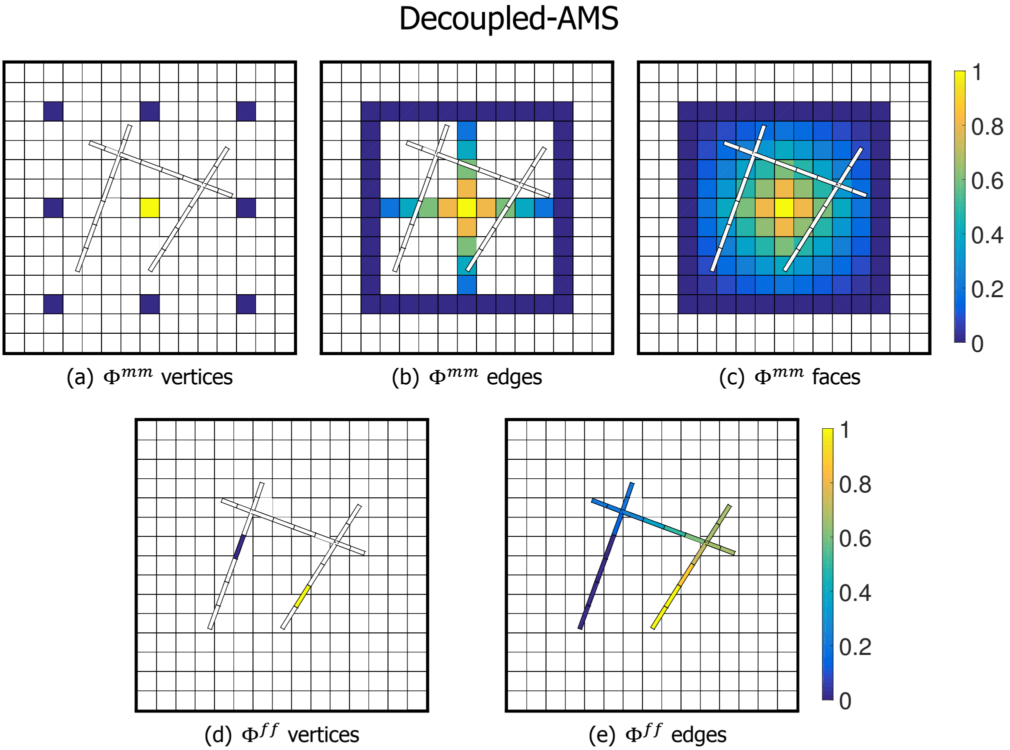
which is solved for all basis functions $\Phi^{*\bullet} \in \{\Phi^{mm}, \Phi^{mf}, \Phi^{mw}, \Phi^{fm}, \Phi^{ff}, \Phi^{fw}\}$, subject to the localization assumption within each domain. Recall that $\mathcal{P}^{wm}$ and $\mathcal{P}^{wf}$ are zero, while $\mathcal{P}^{ww}$ is the identity matrix. The skeleton expression (3.17) is based on the incompressible pressure equation, since it was found computationally efficient, even when compressibility is involved [96]. In its definition, perf^*_w represents the set of (matrix or fracture) cells perforated by wells. Moreover, conn^*_{mf} is the set of all cells with cross-media connectivities from the corresponding (matrix or fracture) domain. Finally, the $\xi(\Phi^{*\bullet})$ function gives the type of matrix-fracture coupling captured by the basis function, and will be specified separately for each strategy, as follows:

1. **Decoupled-AMS:** all basis functions have no-flow boundary conditions between the matrix and fracture domains, i.e.,

   $$\xi(\Phi^{*\bullet}) = 0 \qquad \forall\Phi^{*\bullet} \in \{\Phi^{mm}, \Phi^{mf}, \Phi^{mw}, \Phi^{fm}, \Phi^{ff}, \Phi^{fw}\}, \tag{3.18}$$

   which means that the fracture-matrix coupling term is completely omitted in Eq (3.17). The prolongation is solved algebraically, as described in Appendix B.1. Figure 3.3 illustrates the step-by-step procedure for a fracture and matrix basis function belonging to the 2D reservoir from Fig. 3.1. Note that the support of each of the interpolators is restricted to their containing medium. Finally, the Decoupled-AMS approach can be seen as applying the original AMS to separate sub-domains (i.e., matrix and fractures), having them coupled only at the coarse-scale system.

2. **Frac-AMS:** the fracture basis functions in the fracture domain, Φ^{ff}, are first computed subject to no-flow conditions towards the matrix, the same as in



(a) Φ^{mm} vertices (b) Φ^{mm} edges (c) Φ^{mm} faces

(d) Φ^{ff} vertices (e) Φ^{ff} edges

Figure 3.3: Step-by-step construction of a Decoupled-AMS matrix basis function (top) and fracture function (bottom). In this strategy, Φ^{mf} and Φ^{fm} are explicitly set to 0 and the two media have no-flow boundary conditions towards each other. Note that the fracture aperture is magnified for clarity.

the Decoupled-AMS (Fig. 3.3(d) and 3.3(e)), i.e. by substituting

$$\xi(\Phi^{ff}) = 0 \tag{3.19}$$

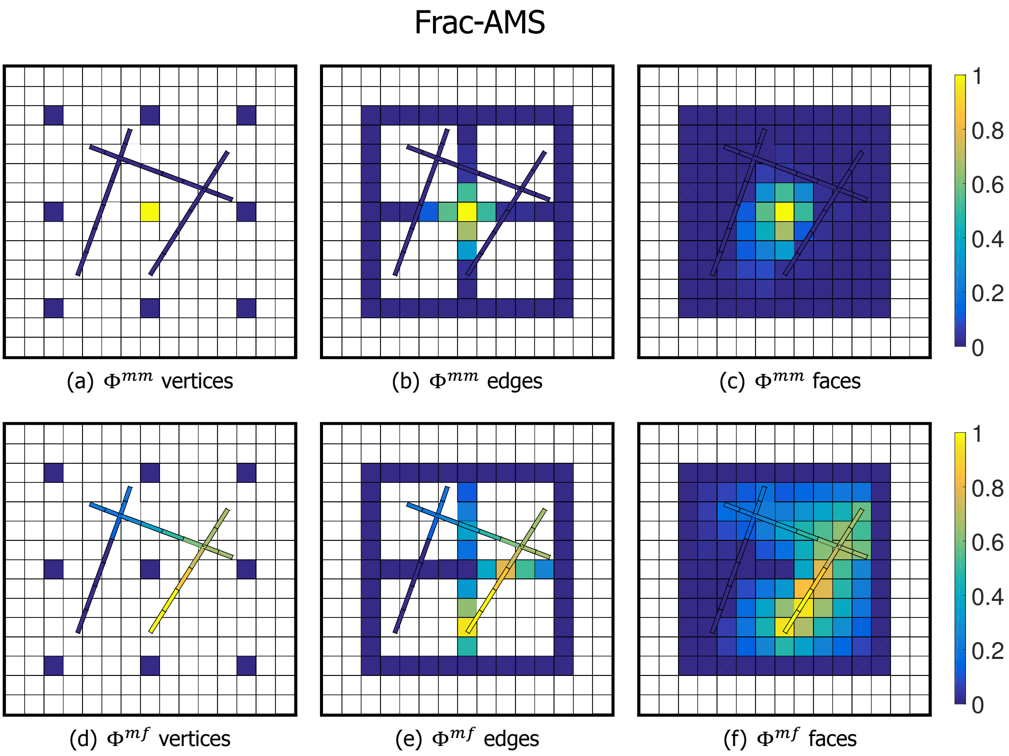
in Eq. (3.17). Then, the obtained values are fixed and used as Dirichlet boundary conditions while computing Φ^{mf}, for which the matrix-fracture transmissibility is taken into account, i.e.,

$$\xi(\Phi^{mf}) = \Phi^{mf} - \Phi^{ff}. \tag{3.20}$$

On the other hand, the matrix basis functions, Φ^{mm}, are solved by setting $\Phi^{fm} = 0$ as Dirichlet boundary condition, i.e.,

$$\xi(\Phi^{mm}) = \Phi^{mm}. \tag{3.21}$$

This procedure is performed algebraically, as described in Appendix B.2. Note from Fig. 3.4 that, after this computation, the fracture functions have non-zero values in the matrix, while the support of the matrix basis functions is restricted to the rock domain.



(a) Φ^{mm} vertices (b) Φ^{mm} edges (c) Φ^{mm} faces

(d) Φ^{mf} vertices (e) Φ^{mf} edges (f) Φ^{mf} faces

Figure 3.4: Step-by-step construction of a Frac-AMS matrix basis function (top) and fracture function (bottom). In this strategy, $\Phi^{fm} = 0$ and used as Dirichlet boundary condition while solving Φ^{mm}. Then, Φ^{ff} is obtained similar as in Decoupled-AMS (Figs. 3.3(d)-3.3(e)) and used as Dirichlet boundary condition for Φ^{mf}.

3. **Rock-AMS:** First, Φ^{mm} is computed with no-flow to the fractures, as with Decoupled-AMS (Figs. 3.3(a)-3.3(c)), i.e.,

$$\xi(\Phi^{mm}) = 0. \tag{3.22}$$

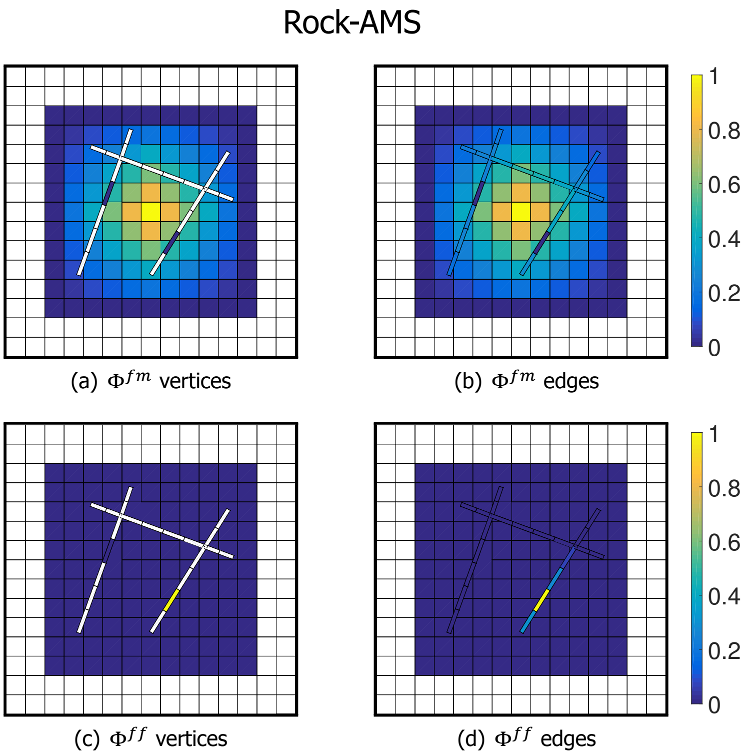
Then, the values are fixed and used as Dirichlet boundaries while solving Φ^{fm}, for which the fracture-matrix connections are taken into account in Eq. 3.17, i.e.,

$$\xi(\Phi^{fm}) = \Phi^{fm} - \Phi^{mm}. \tag{3.23}$$

For the fracture functions, $\Phi^{mf} = 0$ which is used as Dirichlet condition to compute Φ^{ff}, i.e.,

$$\xi(\Phi^{ff}) = \Phi^{ff}. \tag{3.24}$$

Appendix B.3 presents the algebraic procedure corresponding to this coupling strategy. Note from Fig. 3.5 that, in Rock-AMS, the matrix basis functions have non-zero values in the fractures, while the opposite does not hold.



(a) Φ^{fm} vertices (b) Φ^{fm} edges

(c) Φ^{ff} vertices (d) Φ^{ff} edges

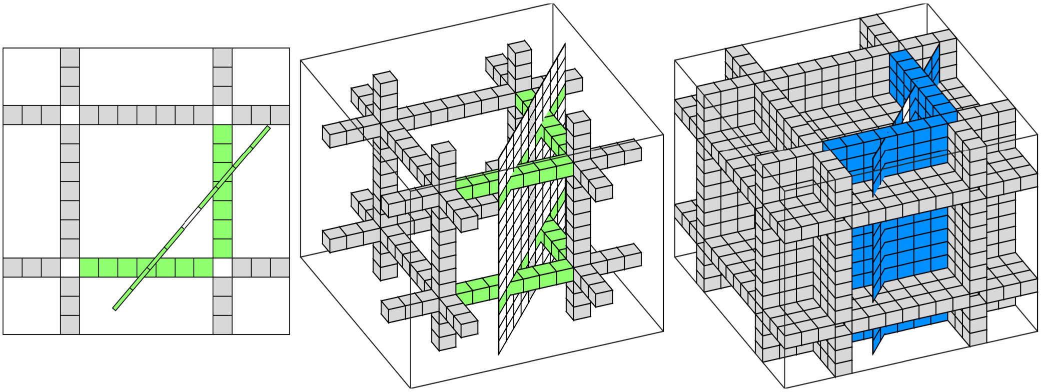
Figure 3.5: Step-by-step construction of a Rock-AMS matrix basis function (top) and fracture function (bottom). In this strategy, the Φ^{mm} are obtained similar to Decoupled-AMS (Figs. 3.3(a)-3.3(c)) and used as Dirichlet boundary conditions while solving Φ^{fm}. Then, $\Phi^{mf} = 0$ and used as Dirichlet boundary conditions for Φ^{ff}.

4. **Coupled-AMS:** In order to preserve the two-way coupling between fractures and matrix, adjacent dual blocks of the same type are merged (e.g. fracture edges with the matrix edges they perforate), as shown in Fig. 3.6. To clarify, two blocks are considered adjacent if there is a non-zero transmissibility between a cell from one of them and a cell from the other.

   On the resulting dual-coarse grid, computation of the basis functions follow the usual wirebasket hierarchy, with full consideration of the coupling, i.e., using

$$\begin{aligned} \xi(\Phi^{mm}) &= \Phi^{mm} - \Phi^{fm}, & \quad & \xi(\Phi^{fm}) = \Phi^{fm} - \Phi^{mm}, \\ \xi(\Phi^{ff}) &= \Phi^{ff} - \Phi^{mf} & \quad \text{and} \quad & \xi(\Phi^{mf}) = \Phi^{mf} - \Phi^{ff}, \end{aligned} \tag{3.25}$$

   for matrix and fracture cells, respectively (see Appendix B.4 for the detailed algebraic procedure). Figure 3.7 presents an illustration of the Coupled-AMS basis functions. Note that fracture functions have non-zero values in the matrix, and vice versa, the matrix basis functions also have non-zero values



(a) Merged edges in 2D (b) Merged edges in 3D (c) Merged faces in 3D

Figure 3.6: The merging of adjacent dual blocks of the same type in 2D (a) and 3D (b and c), in order to preserve the full coupling between fractures and matrix in the Coupled-AMS basis functions.

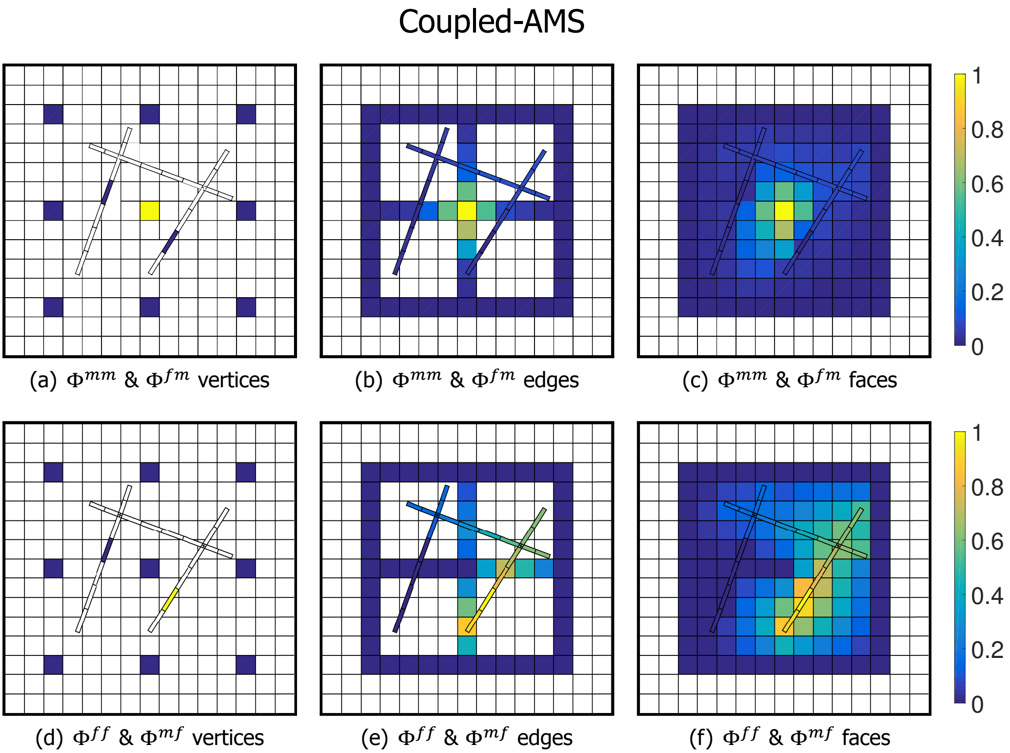
inside fractures.

It is worth mentioning that, by construction, all four coupling strategies result in basis functions which form a partition of unity.

The consideration of wells is similar to what was described in the literature for 2D problems [68], but extended here to 3D problems and integrated within the F-AMS framework. Each well is represented as a single coarse node and a well function is computed locally. The resulting values are assembled in the designated column of the prolongation operator. Note that, even for well perforations in the matrix, the corresponding well functions can have non-zero values inside fractures as well if either Coupled-AMS or Rock-AMS are employed.

As mentioned before, the basis functions have local support. However, in the Coupled-AMS case, the merging of dual blocks (Fig. 3.6) can increase this support substantially in cases with a high density of interconnected (or long) fractures. This, in turn, can lead to a dense prolongation operator, with a severe impact on computational performance. To overcome this, one can impose a limiting criterion on the merging of the dual blocks, for example a maximum number of fine grid cells. Alternatively, or in combination with the previous method, one can discard the non-zeros from $\mathcal{P}$ which lie below a specified threshold, and rescale the rows accordingly to preserve the partition of unity. The latter choice is studied in detail in Section 3.3.

It is important to note that F-AMS basis functions are computed at the beginning of a time-dependent simulation, and adaptively updated only if the fine-scale properties change beyond a threshold value [59]. Next, the F-AMS solution algorithm is described.



(a) Φ^{mm} & Φ^{fm} vertices (b) Φ^{mm} & Φ^{fm} edges (c) Φ^{mm} & Φ^{fm} faces

(d) Φ^{ff} & Φ^{mf} vertices (e) Φ^{ff} & Φ^{mf} edges (f) Φ^{ff} & Φ^{mf} faces

Figure 3.7: Step-by-step construction of a Coupled-AMS matrix basis function (top) and fracture function (bottom). In this strategy, both media preserve their connectivity towards each other.

3.2.2. The F-AMS solution algorithm

In addition to the prolongation, in order to obtain the coarse-scale pressure system, the restriction operator $\mathcal{R}$ (i.e., map from fine to coarse scale) is now defined. As previously described, F-AMS identifies three distinct types of features in the domain, i.e., matrix, fractures, and wells, therefore, $\mathcal{R}$ can be defined in a much more general form than in the previous studies [53, 61, 96].

The first option is to apply a FV-based restriction to all domains, i.e. the MSFV restriction operator, $\mathcal{R}^{FV}$, where the entry at row i and column j is 1 only if the fine-scale cell j (either from the matrix, fractures, or wells) belongs to primal-coarse block i. MSFV ensures mass conservation, at the coarse-scale, thus allowing the reconstruction of a fine-scale conservative flux field. However, in previous works [61, 90], it has been found sensitive to the heterogeneity contrast in the domain. Alternatively, one can construct a Galerkin-FE-based restriction operator, as $\mathcal{R}^{FE} = \mathcal{P}^T$, traditionally called MSFE, which leads to a symmetric-positive-definite (SPD) coarse linear system, if the fine-scale system (3.8) is also SPD. Finally, one can consider a third option, where some of the features (e.g., matrix) are restricted according to FE, while, for the rest, FV is used. This will be referred to as the MSMIX restriction operator. Note that, for multiphase flow problems, if the pressure system is not solved to machine accuracy, a final iteration with $\mathcal{R}^{FV}$ needs to be employed, followed by a conservative flux reconstruction stage, in order to facilitate the solution of the transport equations [43, 73].

Although F-AMS can be used as a single-sweep multiscale solver, where the approximate solution, $\overline{p}'$, is used with no iterations, previous studies have shown that an iterative procedure is needed for highly-heterogeneous reservoirs [56]. To this end, one needs to pair the F-AMS multiscale step with a fine-scale smoother, which ensures error reduction to any desired level. The F-AMS algorithm can now be compiled, as shown in Table 3.1.

Repeat the following steps until convergence to the desired accuracy is reached:

1. **Initialize:** $\overline{p}^{(\nu)} \leftarrow \overline{p}^{(\nu+1)}$
2. **Update linear system entries:** $\mathbf{A}^{(\nu)}$ and $\overline{q}^{(\nu)}$ in (3.8)
3. **Update residual:** $\overline{r}^{(\nu)} = \overline{q}^{(\nu)} - \mathbf{A}^{(\nu)}\overline{p}^{(\nu)}$
4. **Compute (or adaptively update)** $\mathcal{P}$: follow either coupling strategy from Subsection 3.2.1.
5. **Multiscale Stage:** $\delta\overline{p}^{(\nu+1/2)} = \mathcal{P}\left(\mathcal{R}\mathbf{A}^{(\nu)}\mathcal{P}\right)^{-1}\mathcal{R}\overline{r}^{(\nu)}$
6. **Update residual** $\overline{r}^{(\nu+1/2)} = \overline{r}^{(\nu)} - \mathbf{A}^{(\nu)}\delta\overline{p}^{(\nu+1/2)}$
7. **Smoothing Stage:** $\delta\overline{p}^{(\nu+2/2)} = \mathbf{M}_S^{-1}\overline{r}^{(\nu+1/2)}$
8. **Update solution:** $\overline{p}^{(\nu+1)} = \overline{p}^{(\nu)} + \delta\overline{p}^{(\nu+1/2)} + \delta\overline{p}^{(\nu+2/2)}$

Table 3.1: F-AMS solution algorithm

The smoothing operator, $\mathbf{M}_S^{-1}$, approximates the inverse of the complete fine-scale linear operator, $\mathbf{A}^{(\nu)}$, via ILU(0) decomposition [57]. Note that the contrast between the matrix and fracture transmissibility values is usually severe, leading to a high condition number in Eq. (3.8). In such cases, F-AMS can easily be extended to include another smoothing stage, which employs iterations on the sub-block systems corresponding to each media, i.e. $\mathbf{A}^{mm}$ and $\mathbf{A}^{ff}$ from (3.8). A detailed study of the impact of such a smoothing stage is beyond the scope of this work, and makes the object of future research.

Next, numerical results are presented in order to study the effect of each component on the performance of the F-AMS algorithm. Then, the scalability of F-AMS is studied, as linear solver, in a CPU benchmark, where the SAMG commercial solver [45] is used as reference.

3.3. Numerical Results

The aim of this section is to investigate the performance of F-AMS while simulating flow through fractured porous media. First, a 2D reservoir with heterogeneous matrix rock and a relatively complex fracture network is considered. A distance-based graph algorithm is used, in order to automate the fracture coarsening process (see Appendix C). The convergence behaviour of F-AMS is studied, considering each of the four different coupling strategies introduced in the previous section. Then, the same fracture network is extruded along the Z axis and embedded in a 3D heterogeneous domain, for which simulations are performed considering different coarsening strategies, fracture-matrix conductivity ratios, fracture densities and domain sizes. CPU times are measured in detail for both the setup and solution stages in all test cases, and compared to those obtained using the industrial-grade SAMG solver [45]. Finally, the same 3D reservoir is used to investigate the effect of heterogeneous fracture conductivities, spanning several orders of magnitude.

During the upcoming simulations, special attention is given to the conductivity contrast between the matrix and the fracture domains. The transmissibility ratio T_{ratio} is introduced as

$$T_{\text{ratio}} = \frac{\langle T \rangle_{\text{frac}}}{\langle T \rangle_{\text{rock}}}, \tag{3.26}$$

i.e. the ratio between the average fracture $\langle T \rangle_{\text{frac}}$ and matrix $\langle T \rangle_{\text{rock}}$ transmissibility values, respectively.

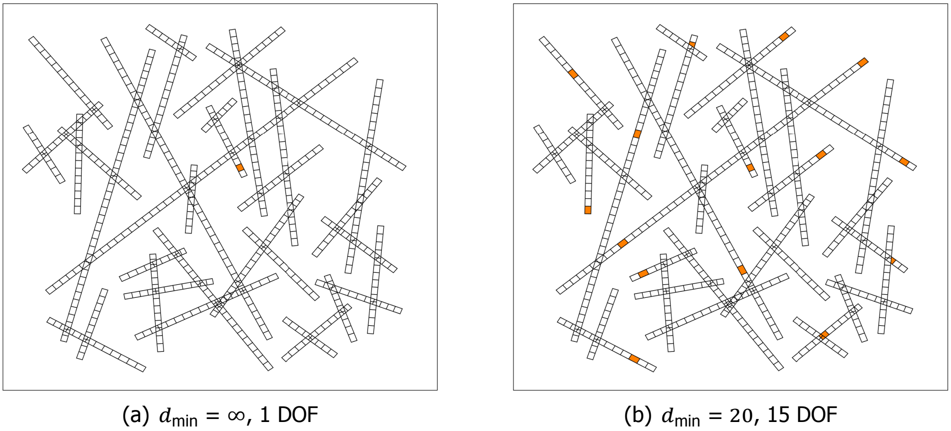
It is important to note that, in all test cases, F-AMS employs a FE restriction operator. Furthermore, the coarse-scale linear system and the basis functions in each dual block are all solved using a direct solver, based on LU decomposition, from the PETSc package [114].

For some experiments, a detailed breakdown of the CPU time spent in each stage of the F-AMS algorithm is presented. In the legends of the corresponding bar plots, the "Initialization" refers to the time spent on allocation of memory for the various data structures, the "Operators" represents the computation of basis functions and construction of the restriction and prolongation matrices (Step 4 in Table 3.1). Also, "Fine linsys. constr." denotes computation of the transmissibility values and fine-scale linear system assembly. In addition, the matrix multiplications resulting in the coarse-scale system are labelled as "Coarse linsys, constr.", while "Solution" stands for the solution of the coarse system followed by the interpolation (Step 5 in Table 3.1). Finally, "Smoother" accounts for Step 7.

3.3.1. F-AMS convergence

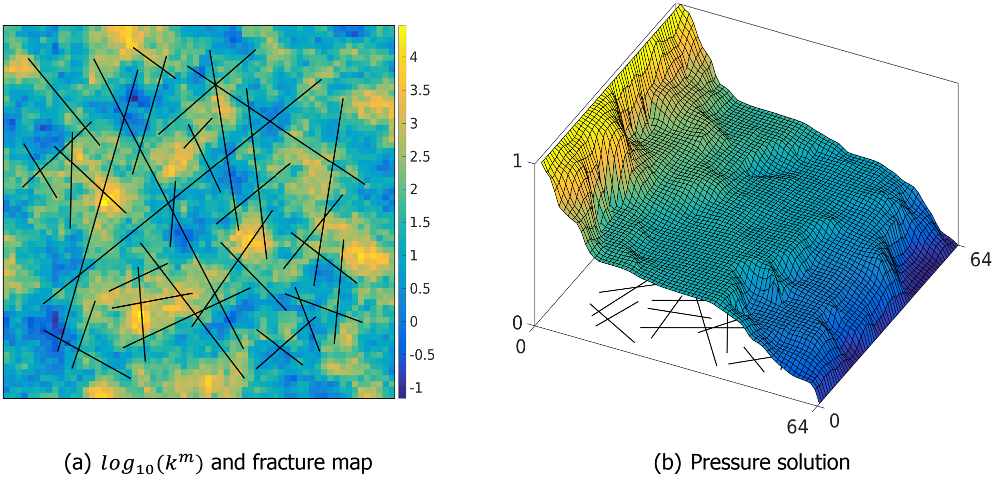
The fracture network from Fig. 3.8 was embedded into a heterogeneous (patchy) matrix rock with two pressure-constrained wells added on the West and East boundaries. This 2D test case, depicted in Fig. 3.9, was used to study the convergence properties of F-AMS, with the four coupling strategies presented before.

In order to test the accuracy of the basis functions as pressure interpolators, F-AMS was stopped after Step 5 of its first iteration (see Table 3.1). The pressure solutions, obtained while changing the number of DOF in the fracture network, are shown in Fig. 3.10, for each basis function coupling strategy. Using a single fracture coarse DOF leads to a poor approximation of the pressure distribution, especially for Decoupled- and Frac-AMS (left side of Figure 3.10). In this setup, F-AMS treats fractures similar to [27]. Note that, due to the large length scale of the network,


(a) $d_{\text{min}} = \infty$, 1 DOF

(b) $d_{\text{min}} = 20$, 15 DOF

Figure 3.8: Coarse nodes on a network with 575 fine-scale cells. Note that the aperture is magnified for clarity.

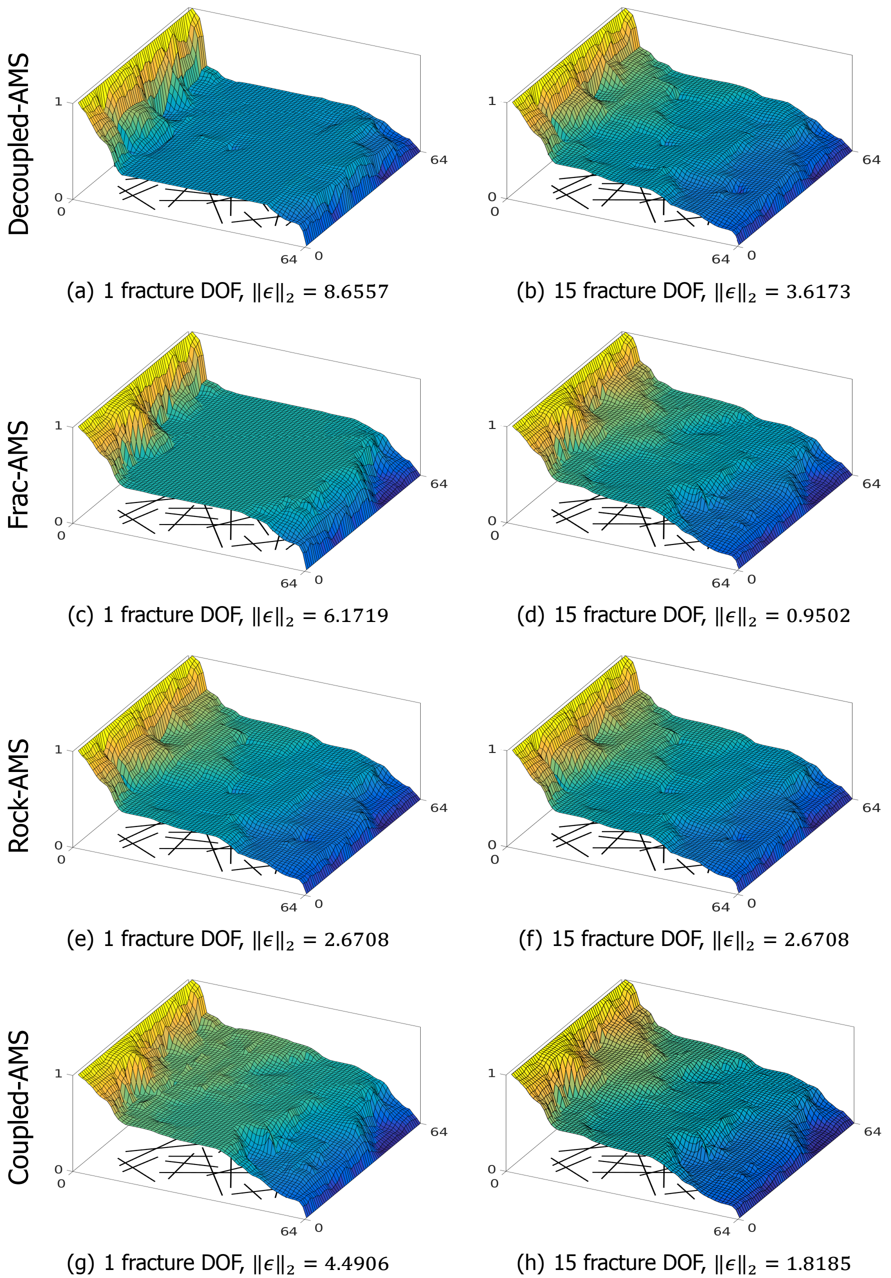


(a) $log_{10}(k^m)$ and fracture map

(b) Pressure solution

Figure 3.9: 2D Test Case: heterogeneous matrix rock with 64×64 matrix and 575 fracture grid cells (a). Two pressure-constrained wells at the West and East boundaries are placed, resulting in the pressure solution shown in (b) for $T_{\text{ratio}} = 100$.

using a constant interpolator for the fracture pressure (as is the case in Decoupled-AMS and Frac-AMS with 1 fracture DOF) results in an initial solution which lacks a lot of the fine-scale features. In contrast, the Rock-AMS and Coupled-AMS place a lot more emphasis on the matrix basis functions and, since, in this test case, the rock heterogeneity is the main source of approximation error, their results are more accurate. It may seem unexpected that the Coupled-AMS performs slightly worse than Rock-AMS. This can be attributed to the fact that a single fracture DOF



(a) 1 fracture DOF, $\|\epsilon\|_2 = 8.6557$

(b) 15 fracture DOF, $\|\epsilon\|_2 = 3.6173$

(c) 1 fracture DOF, $\|\epsilon\|_2 = 6.1719$

(d) 15 fracture DOF, $\|\epsilon\|_2 = 0.9502$

(e) 1 fracture DOF, $\|\epsilon\|_2 = 2.6708$

(f) 15 fracture DOF, $\|\epsilon\|_2 = 2.6708$

(g) 1 fracture DOF, $\|\epsilon\|_2 = 4.4906$

(h) 15 fracture DOF, $\|\epsilon\|_2 = 1.8185$

Figure 3.10: Pressure solutions after a single multiscale iteration with 1 (left) and 15 (right) fracture coarse cells and 4 different prolongation coupling strategies. In all cases, the matrix coarse grid contains 8×8 blocks and the 2-norm of the error is indicated.

is not sufficient to accurately capture the pressure distribution in the large fracture network, especially under the localization assumption. However, when only few additional coarse DOF are added in the fracture domain (as shown on the right side of Fig. 3.10), the situation improves dramatically for Decoupled-AMS, Frac-AMS and Coupled-AMS. Note that Rock-AMS, on the other hand, is insensitive to this change.

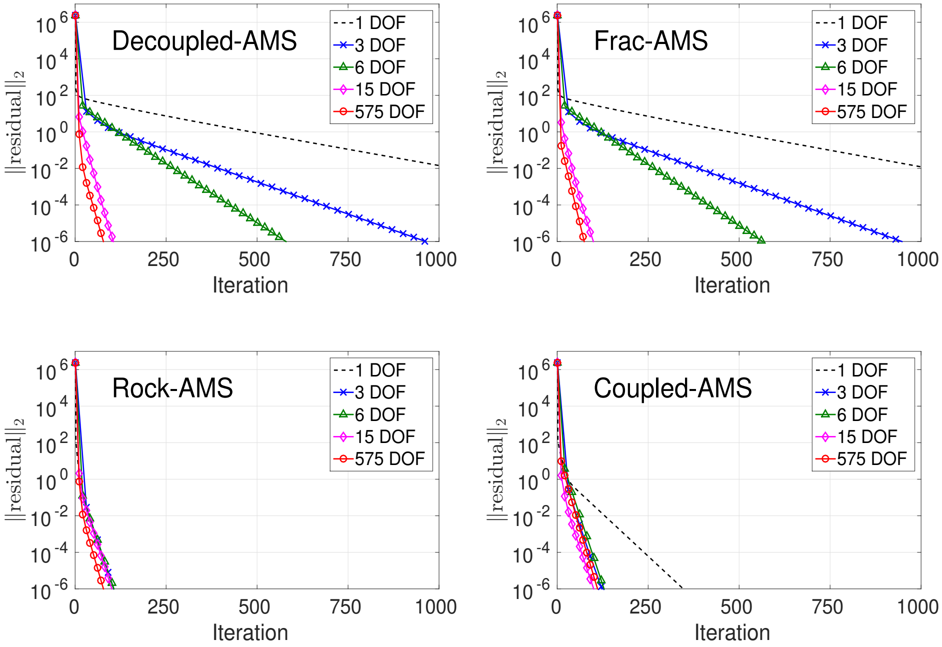
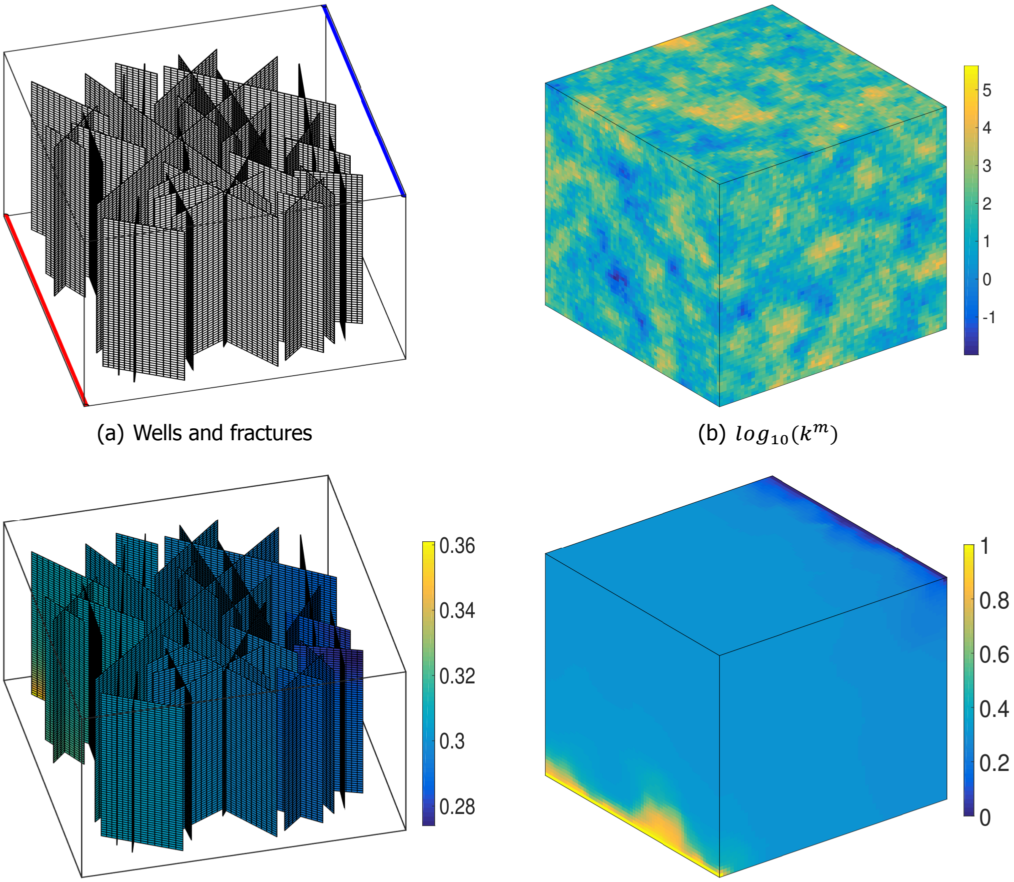


Figure 3.11: Convergence history of F-AMS for the 2D test case with four basis function coupling strategies and different number of coarse DOF in the fracture network.

For the results in Fig. 4.18, as well as all subsequent experiments in this work, F-AMS was iterated until converged to a residual 2-norm of 10^{-6}. It is clear that Rock-AMS shows a good convergence rate on this particular 2D test case, regardless of the fracture coarsening factor. Also, the other strategies reach a similar behaviour when the fracture network is enriched with only few additional coarse-scale DOF.

3.3.2. Basis function truncation

In order to get an idea of the performance of F-AMS on realistic fractured reservoirs, Fig. 3.12 introduces a 3D scenario, where the fracture network from Fig. 3.9 was extruded and discretized along the Z axis. Two pressure-constrained horizontal wells are placed on opposite edges of the domain boundary. Figure 4.20 shows the fine-scale pressure solution obtained on the heterogeneous (patchy) matrix permeability field shown in Fig. 3.12(b). Note that, even though the matrix-fracture conductivity contrast is of only two orders of magnitude, this is enough to make the pressure distribution in the fracture network insensitive to the matrix heterogeneity (see the approximately constant pressure in Fig. 3.12(c)).



(a) Wells and fractures

(b) $log_{10}(k^m)$

(c) Fracture pressure

(d) Matrix pressure

Figure 3.12: Illustration of the 3D test case with 575×64 fracture and $64 \times 64 \times 64$ matrix grid cells. The logarithm of the heterogeneous permeability map is provided in (b). Two pressure-constrained wells are placed on opposite edges, as shown in (a). The pressure solution is shown for $T_{\text{ratio}} = 100$ for fracture (c) and matrix (d). F-AMS employs the coarsening ratios of $8 \times 8 \times 8$ for matrix and 8×8 for fractures.

The procedure described in Appendix C is followed to determine the fracture coarse nodes along the projection of the network on the X-Y plane. Then, the resulting coarse grids are extruded along the Z-axis uniformly, with the vertical distribution of the coarse nodes honouring the user-specified coarsening ratio (see Fig. 3.2).

The increased number of cells in both the matrix ($64 \times 64 \times 64$), as well as the fracture network (575×64), compared to the 2D case, can lead to a larger number

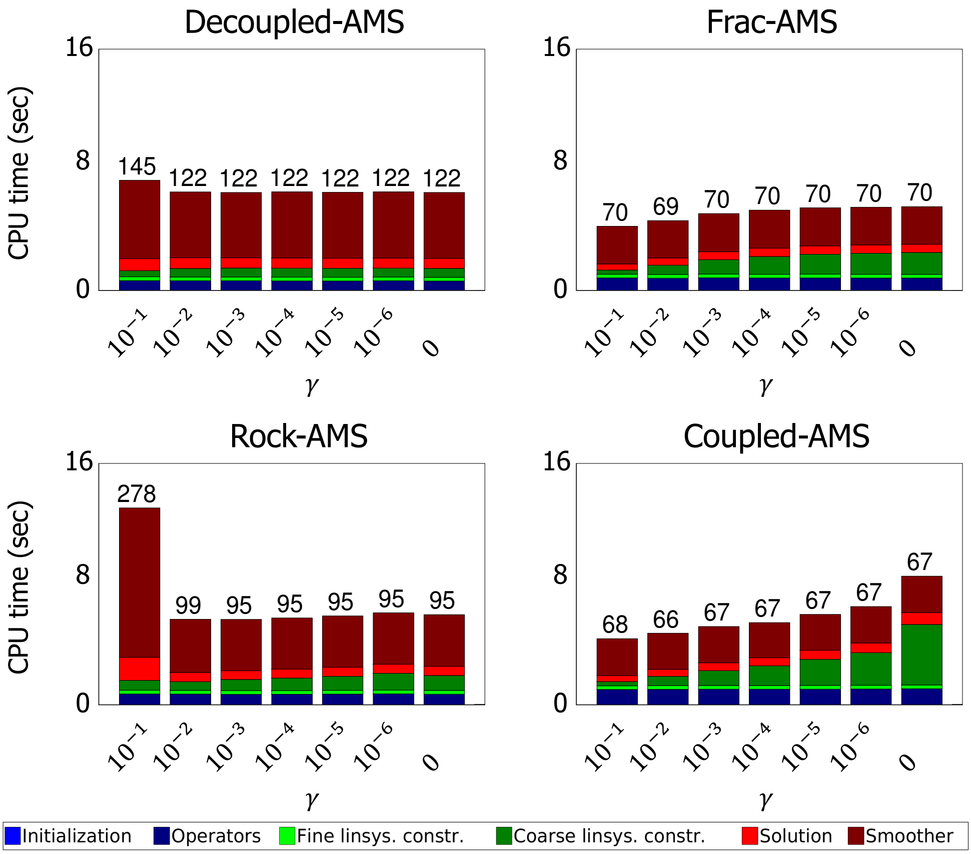


Figure 3.13: F-AMS performance for different values of the basis function truncation threshold, γ. Choosing $\gamma = 0$ invalidates the truncation procedure. The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar.

of non-zeros for the basis functions which take into account the coupling between the two media. This can be particularly severe for Coupled-AMS (Fig. 3.7), since the high density of fractures can cause a large number of dual blocks to be merged. The resulting basis functions have a wider support and can potentially lead to more accurate interpolations, however, the added density to the prolongation operator will also increase the computational effort necessary to construct and solve the coarse-scale linear system (i.e., Step 5 in Table 3.1).

One can limit the density of $\mathcal{P}$ by truncating basis function values below a specified threshold, $\gamma \in [0, 1)$. However, in order to preserve partition of unity, the affected rows in $\mathcal{P}$ need to be rescaled by dividing the remaining values by the row sum. Figure 3.13 shows the CPU time spent by F-AMS on the 3D test case, while varying the value of γ. Notice that the very restrictive value of $\gamma = 10^{-1}$ leads to an increase in the number of overall iterations, because the smoother needs

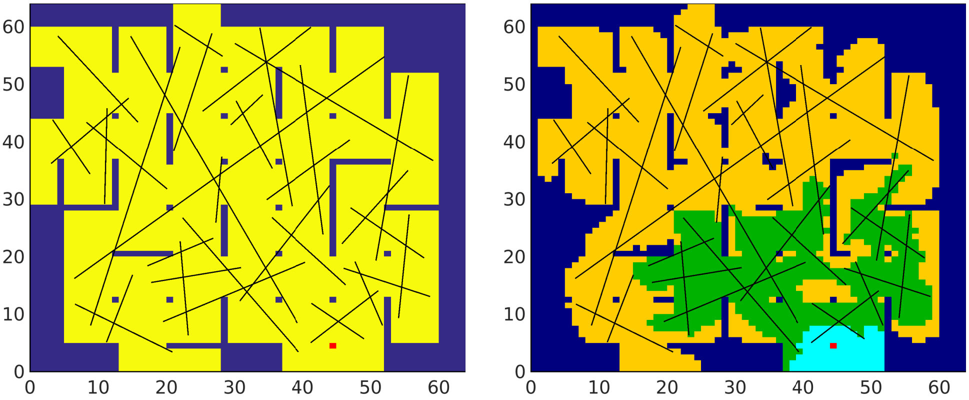


Figure 3.14: Coupled-AMS basis function support before (left) and after truncation (right). The coarse node (red) is located in a low permeable region of the 2D test case. The fracture network contains only 1 coarse-scale DOF. The colors on the right plot correspond to $\gamma = 10^{-5}$ (orange), 10^{-3} (green) and 10^{-2} (light blue).

to compensate for the induced inaccuracy of the basis functions. However, starting with $\gamma = 10^{-2}$, the convergence is no longer much affected and the algorithm gains efficiency from the reduced number of computations (FLOPS) necessary to perform $\mathcal{R}\mathbf{A}^{\nu}\mathcal{P}$ and invert the result. The truncation has the biggest impact on the Coupled-AMS strategy, which experiences a speed-up factor of 2, compared to the un-truncated case (last bar in Fig. 3.13(d), where $\gamma = 0$). Figure 3.14 shows that, for this coupling strategy, when only a single DOF is considered for fracture network, the support of basis functions can be as big as the span of the fracture network. Also, this figure shows that after the truncation stage the locality of the basis function support can be maintained. As conclusion to this study, the subsequent experiments will use a value of $\gamma = 10^{-2}$, regardless of the choice of basis function coupling strategy.

3.3.3. Sensitivity to the coarsening factor

The sensitivity of F-AMS to the number of coarse DOF in the fracture network, as well as the matrix coarsening ratio is studied for the 3D test case shown in Fig. 3.12. The coarsening factor is defined as the average number of fine cells contained in one (matrix or fracture) primal-coarse block, along each axis. Recall that, along the fracture length, this is given by the d_{min} parameter in Table C.1. The experiment designed for this purpose is focused on "isotropic" coarsening factors, mainly due to the point-wise nature of ILU(0), which was chosen as global smoothing stage for the implementation of F-AMS used here (Table 3.1).

Figure 4.17 shows the F-AMS CPU times obtained with three different coarsening factors for the matrix, as well as the fracture network. From this figure, the Coupled-AMS is found to automatically adapt itself to the coarsening ratio. Its con-

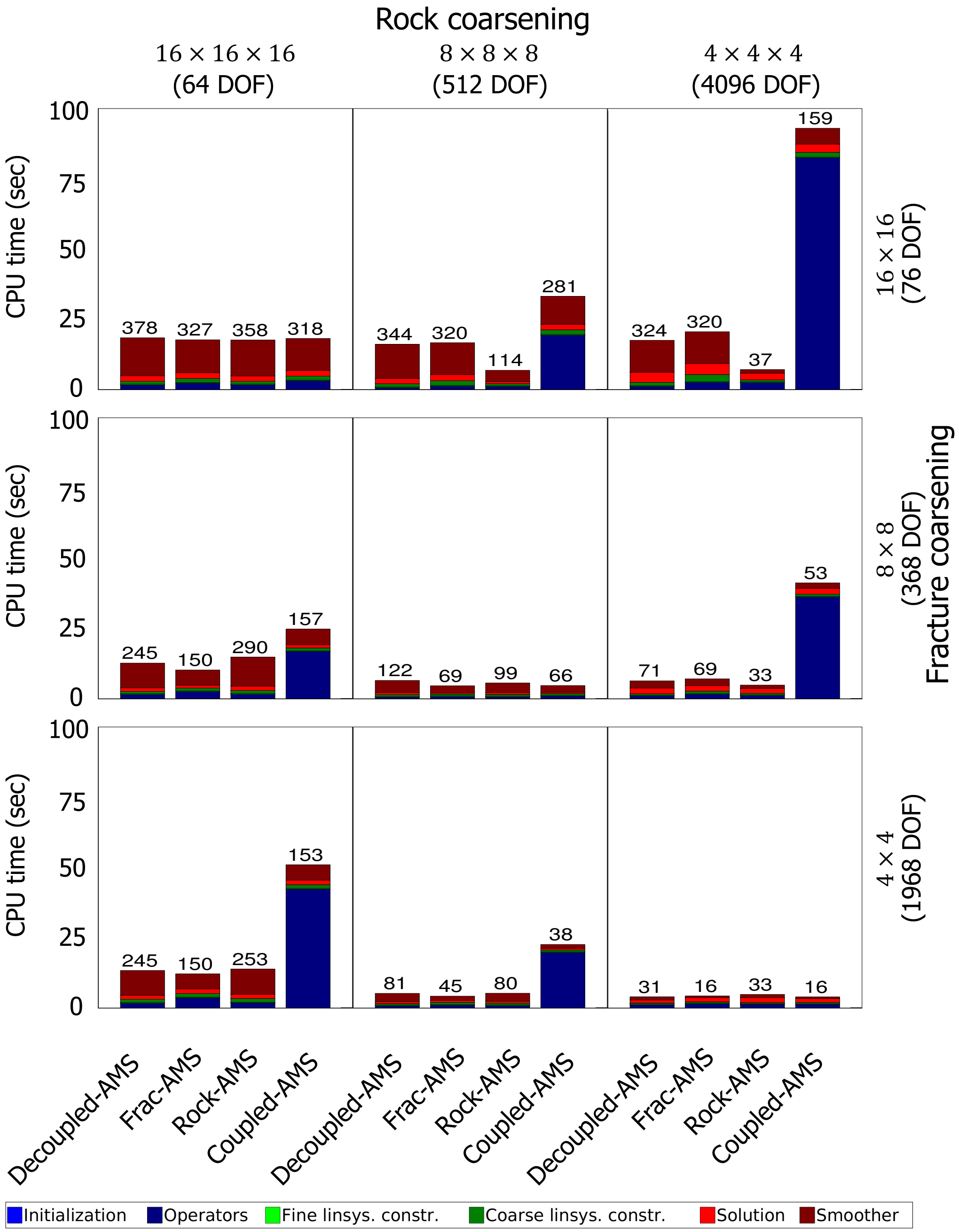


Figure 3.15: F-AMS performance for different matrix (decreasing from left to right) and fracture coarsening ratios (decreasing from top to bottom). The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar.

vergence rate is surpassed only in cases where there is a large discrepancy between the rock and fracture coarsening ratios. However, this comes with the additional computational cost of having basis functions with wider support. Still, due to the truncation factor $\gamma = 10^{-2}$, the construction and solution of coarse system for Coupled-AMS is not significantly higher than that of the alternative strategies. In addition, in each row, the optimum simulation results are obtained when fracture and matrix coarsening ratios are similar.

Based on this study and unless otherwise stated, the coarsening ratio of 8 in each direction for both matrix and fracture media is employed in the experiments presented in the following subsections. Note that this option leads to more efficient coarse-scale systems than the alternative option of using coarsening ratios of 4. In addition, the number of linear iterations can be significantly reduced when F-AMS is employed as preconditioner for GMRES [57].

3.3.4. Sensitivity to the transmissibility ratio

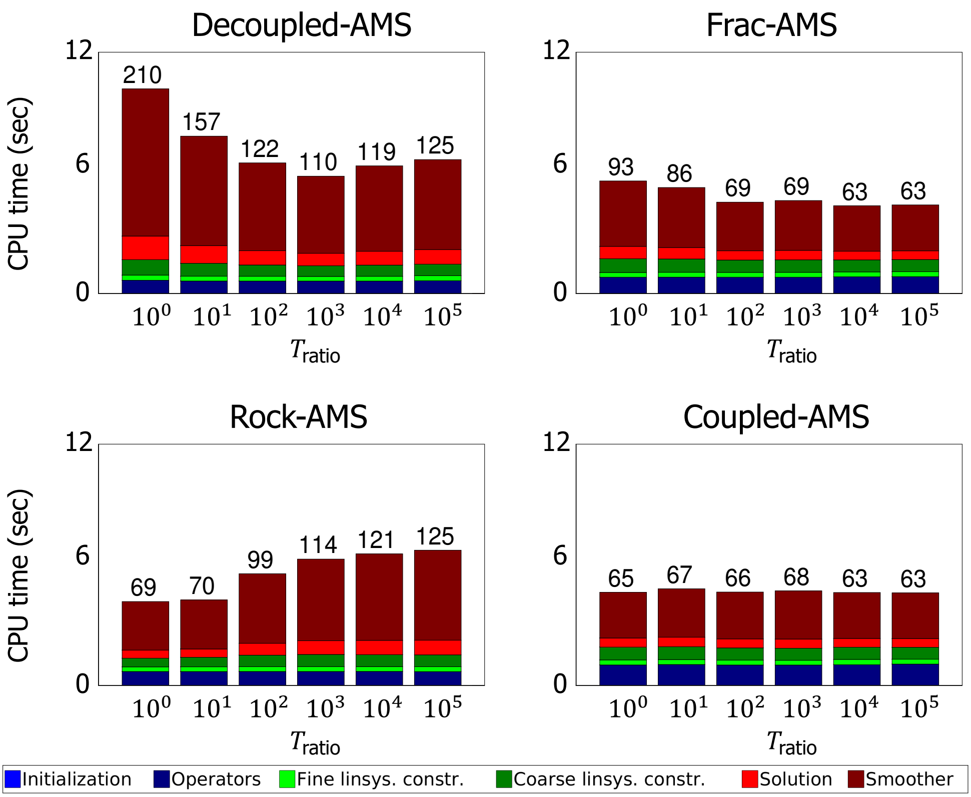


Figure 3.16: F-AMS performance for different matrix-fracture transmissibility ratios. The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar.

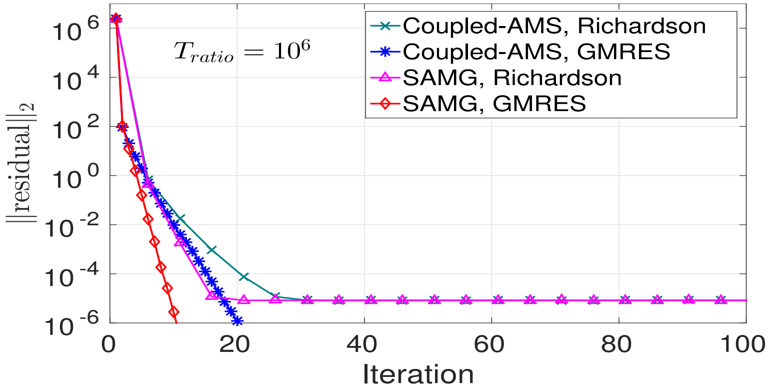


Figure 3.17: Convergence history of F-AMS and SAMG on the 3D test case with a fracture/matrix transmissibility ratio of $T_{\text{ratio}} = 10^6$. Notice that neither method can converge when iterated in a Richardson's loop. Instead, as preconditioners to GMRES, both methods converge after a few iterations.

The next set of experiments aims to investigate the sensitivity of F-AMS to the conductivity contrast between the matrix and fractures. The transmissibility ratio T_{ratio}, as defined in Eq. (3.26), is varied over several orders of magnitude (Fig. 3.16) while measuring CPU times and number of linear iterations performed by the Richardson loop. A coarsening factor of 8 was chosen for both media, based on the results from the previous subsection. As the network becomes more conductive, the influence of the matrix heterogeneity on the fracture pressure decreases (see Figs. 3.12(c) and 4.20). As such, Rock-AMS exhibits a degradation in performance for higher T_{ratio} (Fig. 3.16(c)), while the reverse is true for Frac-AMS (Fig. 3.16(b)). On the other hand, by automatically adapting to the change, the Coupled-AMS strategy remains relatively insensitive to T_{ratio}, as shown in Fig. 3.16(d). Finally, Decoupled-AMS requires the most number of iterations when the fracture and matrix transmissibility values are close (Fig. 3.16(a)), since in this case the two-way coupling between the media is the most pronounced.

Note that the solver could not converge to the chosen tolerance, of 10^{-6} residual norm, when F-AMS was iterated in a Richardson's loop for $T_{\text{ratio}} \geq 10^6$ and the same holds for SAMG [45]. However, as shown in Fig. 3.17, both methods converge successfully when employed as preconditioners for GMRES [57].

3.3.5. CPU benchmark study

This final subsection presents the results of a benchmark study between F-AMS and SAMG [45] on 3D heterogeneous fractured reservoirs. Both methods are employed as preconditioners to GMRES [57] and iterated until converged with a residual 2-norm below 10^{-6}.

Unlike the Richardson loop, similar performance was observed for all experiments when F-AMS is used as preconditioner to GMRES, regardless of which coupling strategy was chosen. Therefore, the presentation of the results is restricted to Decoupled-AMS, for conciseness. Note that the Coupled-AMS strategy provides a more general framework, however, with a more complex implementation.

At each GMRES iteration, SAMG employs a single V-cycle. It is important to note that SAMG is a commercial black-box package. Thus, it is not possible to measure its CPU breakdown as accurately as for F-AMS. Instead, the time spent on its first iteration is considered as "Initialization", while subsequent iterations were labelled as "Solution". Finally, for both SAMG and F-AMS, the setup and construction of the operators are performed only once, at the beginning of the iteration procedure.

This study is aimed only to demonstrate the scalability of the F-AMS method. Note that a unique advantage of F-AMS over SAMG is that a fine-scale mass conservative flux field can be reconstructed after any iteration stage, once the coarse-scale system with $\mathcal{R}^{FV}$ restriction operator is solved.

Transmissibility contrast

The test case from Fig. 3.12 is used, with different values of fracture-matrix transmissibility contrasts, i.e., T_{ratio} in Eq. (3.26). As can be seen in Fig. 3.18, the number of iterations and the CPU time for both F-AMS and SAMG is insensitive to the contrast. This is a significant achievement for F-AMS, compared to [27].

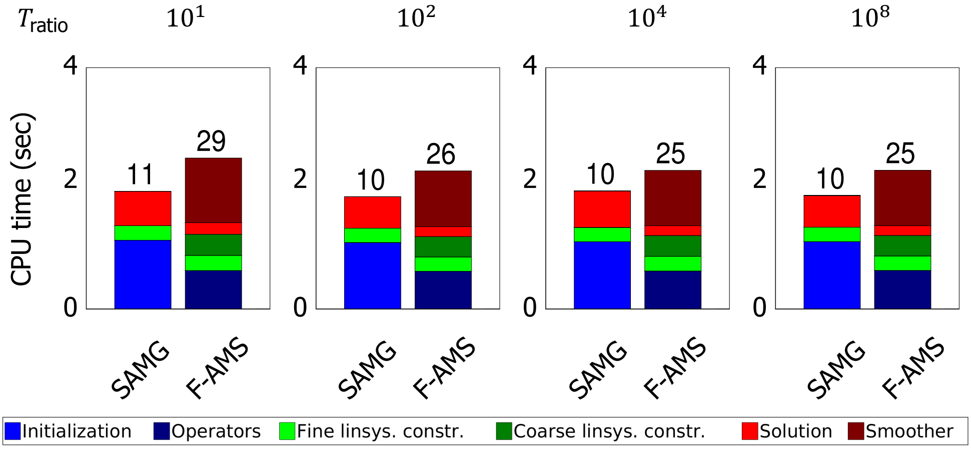


Figure 3.18: F-AMS (Decoupled-AMS) performance compared with SAMG for different matrix-fracture transmissibility ratios. The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar. For these experiments, both methods were used as preconditioners for GMRES. Similar performance was observed for other F-AMS coupling strategies.

Fracture density

What follows is a study of the scalability of F-AMS when faced with a dynamic fracture network, where the number of fracture plates is increased step by step. The 3D fracture map shown in Fig. 3.12(a) is considered, where the network is now created through 4 phases, as presented in Fig. 3.19. Note that, as new fractures are added, not only the number of DOF increases, but also the pressure variation along the network can increase. The detailed description of the CPU times obtained using F-AMS in these four cases are depicted in Fig. 4.21. It is clear that, by maintaining the prescribed fracture coarsening factor of 8×8, F-AMS maintains virtually the same convergence rate. The slight increase in CPU time is mainly due to computation of extra fracture basis functions, as well as the construction and solution of a slightly larger coarse-scale linear system.

In consequence, by having multiple coarse-scale DOF in each fracture network, F-AMS can automatically scale with fracture length and density. This is in contrast to [27], where the use of a single fracture basis function would lead to a drastic deterioration of the multiscale convergence for test cases containing fracture networks with large length scales.

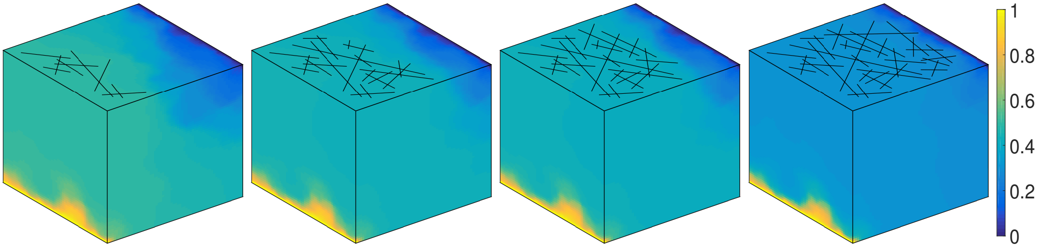


Figure 3.19: Pressure solution obtained for different fracture densities. The left-most contains 27 fracture plates (136×64 cells), followed by one with 76 fracture plates (334×64 cells), the next has 96 fracture plates (438×64 cells), and, finally, the right-most is perforated by 127 fracture plates (575×64 cells).

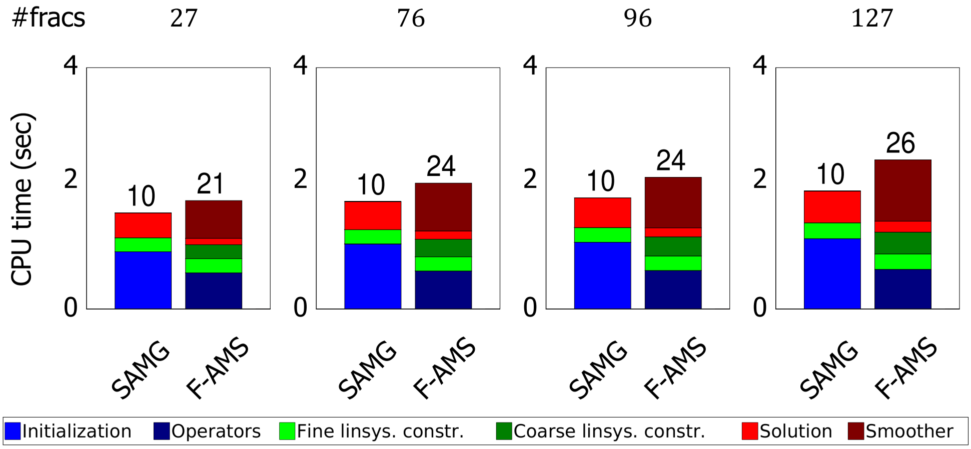


Figure 3.20: F-AMS (Decoupled-AMS) performance for reservoirs with different number of fractures. The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar. For these experiments, both methods were employed as preconditioners for GMRES. Similar performance was observed for other F-AMS coupling strategies.

Domain scale

The scalability of F-AMS, benchmarked with SAMG, is investigated for heterogeneous (patchy) reservoir of increasing size. To this end, both the matrix and fracture fine-scale grid resolution is varied from 32^3 matrix and 320×32 fractures (smallest) up to 256^3 matrix and 2117×256 fracture cells (see Fig. 3.21). The transmissibility ratio between the two media is $T_{\text{ratio}} = 10^2$. Figure 3.22 shows the obtained CPU times. F-AMS and SAMG both maintain their convergence rates and experience a similar level of scalability, in terms of CPU time, i.e., they grow linearly with the problem size. Note that SAMG uses adaptive coarsening at each level in its V-cycles.

During the simulations carried out for this experiment, as well as those from Subsection 3.3.3, it became clear that the performance of F-AMS is highly dependent on the coarsening ratios used. Only the optimum configuration was featured in the plots from Figure 3.22. Table 3.2 lists the performance and convergence of F-AMS, when using primal grids with more refined and more coarse blocks, respectively, for comparison purposes.

Scale	Coarsening ratio (matrix, fracs)	Total CPU time (sec)	# iterations
32^3	$2 \times 2 \times 2,\ 2 \times 2$	1.904	11
	$\mathbf{6 \times 6 \times 6,\ 6 \times 6}$	**0.330**	**23**
	$8 \times 8 \times 8,\ 8 \times 8$	0.351	29
64^3	$4 \times 4 \times 4,\ 4 \times 4$	3.270	13
	$\mathbf{6 \times 6 \times 6,\ 6 \times 6}$	**2.186**	**19**
	$9 \times 9 \times 9,\ 9 \times 9$	2.371	27
128^3	$6 \times 6 \times 6,\ 6 \times 6$	21.790	17
	$\mathbf{8 \times 8 \times 8,\ 8 \times 8}$	**17.620**	**22**
	$11 \times 11 \times 11,\ 11 \times 11$	21.350	31
256^3	$8 \times 8 \times 8,\ 8 \times 8$	164.600	18
	$\mathbf{10 \times 10 \times 10,\ 10 \times 10}$	**150.400**	**23**
	$17 \times 17 \times 17,\ 17 \times 17$	252.100	40

Table 3.2: Performance of F-AMS during the scale sensitivity test cases, when using different coarsening factors. The middle row for each test (shown in bold) is the optimum configuration, whose results were presented in the body of the manuscript.

Heterogeneous fractures

Finally, the sensitivity of the F-AMS method (benchmarked with SAMG) to strongly heterogeneous fracture properties is investigated. The permeability of the 127 fracture plates from Fig. 3.12(a) is randomly perturbed to a span 6 orders of magnitude. Figure 3.23 shows that, if an appropriate coarsening ratio is chosen (in this case

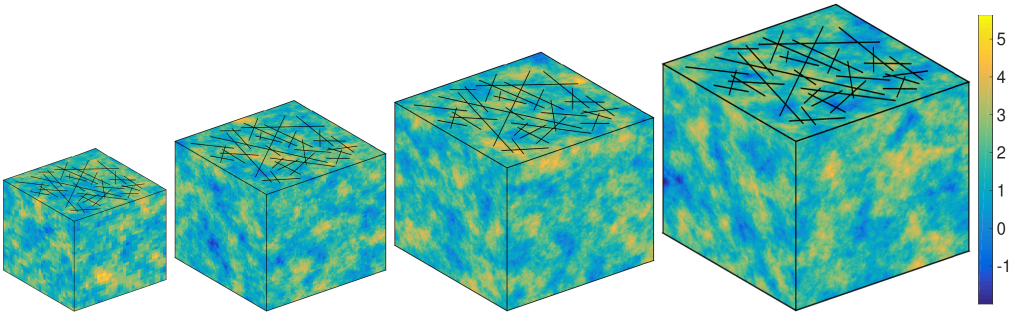


Figure 3.21: Logarithm of the permeability ($log_{10}(k^m)$) and fracture distribution in four reservoirs of increasing size. The left-most has 32^3 matrix and 320×32 fracture cells, followed by 64^3 matrix and 575×64 fracture cells, then 128^3 matrix and 1087×128 fracture cells, and, finally, 256^3 matrix and 2117×256 fracture cells.

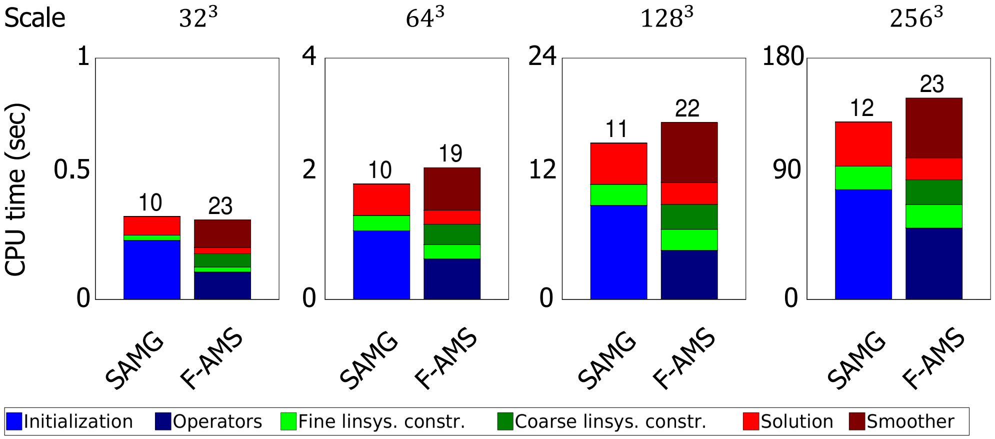


Figure 3.22: F-AMS (Decoupled-AMS) performance compared with SAMG for different domain sizes. The coarsening factors are 6, 8, 8 and 10 respectively. Also, a transmissibility ratio of $T_{\text{ratio}} = 10^2$ was considered. The number of iterations to reach 10^{-6} residual 2-norm is given on top of each bar. For these experiments, both methods were employed as preconditioners for GMRES. Similar performance was observed for other F-AMS coupling strategies.

$6 \times 6 \times 8$ in the matrix and 4×8 in the fractures), then the F-AMS and SAMG performances are comparable. In addition to the coarsening ratio, multiscale methods are also sensitive to the heterogeneity contrasts (here, in both fractures and matrix). Improvements can be achieved by adapting the coarse grid geometry to follow the fracture and matrix conductivity distribution, or by enriching the prolongation operator with additional basis functions [86–88]. These are subjects of future studies.

The results of the experiments presented in this section show that the performance and scalability of F-AMS is comparable to that of SAMG. As such, even in its current proof of concept implementation, F-AMS is found a promising multiscale

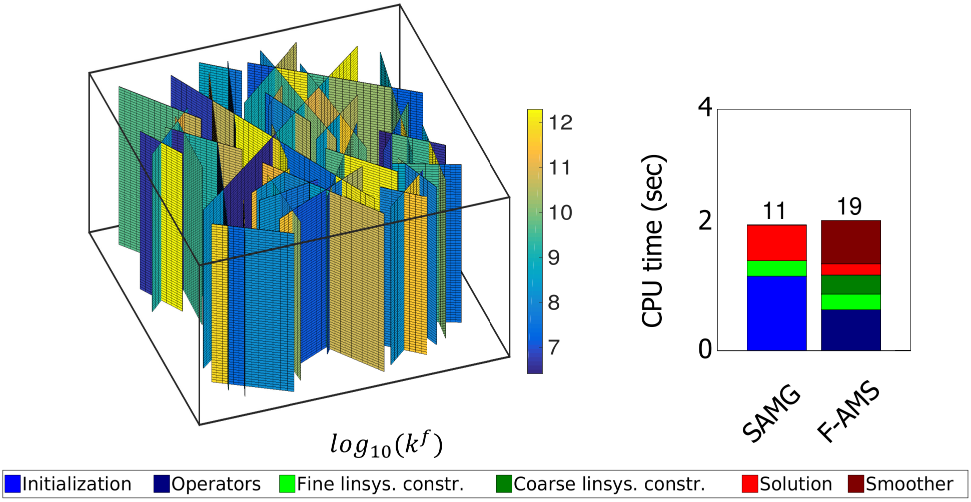


Figure 3.23: F-AMS (Decoupled-AMS) performance compared with SAMG on a test case with heterogeneous fracture permeability. The logarithm of the fracture permeability is shown on the left. The number of iterations to reach 10^{-6} residual 2-norm is given on top of each bar. For these experiments, both methods were employed as preconditioners for GMRES.

method for naturally fractured porous media. Note that, for realistic multiphase test cases, simulations can be further accelerated by employing only few iterations of F-AMS, followed by a flux reconstruction stage, leading to efficient approximate solutions [99].

3.4. Conclusions

This chapter introduced a novel general multiscale framework, F-AMS, devised for efficient and accurate simulation of flow through heterogeneous porous media with embedded fractures of various length scales. The possibility to prescribe an arbitrary coarse grid in each fracture network was presented. Then, for each coarse node (from both matrix and fractures), a locally-supported basis function was defined, by considering one of the four cross-media coupling strategies (Decoupled-AMS, Frac-AMS, Rock-AMS or Coupled-AMS). All of these flexibilities allow the user to tweak the trade-off between the computational budget of the setup stage and the convergence rate.

Aligned with the EDFM approach of having independent grids for fracture and matrix [18], this work used a distance-based automatic coarsening algorithm for the fracture domain. It allows the user to specify the desired (uniform) coarsening factor for the fracture networks, in similar fashion to the matrix. In addition, the effect of truncating small non-zeros from the prolongation operator was studied, in order to maintain efficiency, especially for the Coupled-AMS strategy. For all test cases considered, a truncation value of $\gamma = 10^{-2}$ was found to be optimal.

The numerical results illustrate that F-AMS (similar as SAMG) is insensitive to the fracture-matrix conductivity contrast, and - importantly - shares the same scalability with respect to the fracture density, domain scale and heterogeneous properties. However, the performance of F-AMS was found to dramatically degrade if a sub-optimal coarsening strategy is chosen for challenging test cases. The method can be further extended to address this, e.g. by considering adaptive coarsening strategies, different choices for the smoother or employment of enriched prolongation operators [86, 87].

Another important finding of this work was that all basis function coupling strategies perform similar when F-AMS is used as preconditioner to GMRES. This recommends the Decoupled-AMS approach for commercial reservoir simulation, due to its convenient implementation, an attractive feature for real-field applications.

In summary, it is concluded that F-AMS is an important multiscale development for flow in heterogeneous media with embedded fractures. It was shown that only few fracture coarse nodes are required to deliver good approximate pressure solutions, at the original fine-scale resolution. Future developments of F-AMS will include consideration of unstructured grids (see [83, 84, 115]) or the inclusion complex physics such as geomechanics, capillarity and gravity effects for multiphase flow simulations.

4

Projection-based Embedded Discrete Fracture Model

Accurate and efficient simulation of flow through subsurface formations is essential for effective engineering operations (including production, storage optimization and safety assessments). Alongside their intrinsic heterogeneous properties, the target geological formations often contain complex networks of naturally-formed or artificially-induced fractures, with a wide range of conductivity properties. Given their significant impact on flow patterns, the accurate representation of these lower-dimensional structural features is paramount for the quality of the simulation results [105].

Discrete Fracture Models (DFM) reduce the dimensionality of the problem by constraining the fractures, as well as any inhibiting flow barriers, to lie at the interfaces between matrix rock cells [24–26]. Then, local grid refinements are applied, where a higher level of detail is necessary, leading to a discrete representation of the flow equations on, sometimes complex, unstructured grids [32, 39, 107, 118, 119]. Although the DFM approach has been extended to include complex fluids and rock physics – e.g. compositional displacements [120, 121] and geomechanical effects [122] – its reliance on complex computational grids may raise important challenges in real-field applications.

This has led to the emergence of models which make use of non-conforming grids w.r.t. fracture-matrix connections, such as eXtended Finite Element Methods (XFEM) [123] and Embedded Discrete Fracture Models (EDFM) [18, 19]. This work focuses on the latter, which are especially appealing due to their intrinsic ability to deliver mass-conservative flux fields. To this end, the lower-dimensional structural features with relatively small lengths (i.e. fully contained in a single fine-scale matrix cell) are first homogenized, by altering the effective permeability of their

The material presented in this chapter has been published in the proceedings of the ECMOR XV - 15th European Conference on the Mathematics of Oil Recovery (2016) [116] and in Advances in Water Resources **105**, 205 (2017) [117].

support rock [20]. Then, the remaining fracture networks are discretized on separate numerical grids, defined independently from that of the matrix [113, 124]. A comprehensive comparison between DFM and EDFM, along with other fracture models, is performed in [125].

The EDFM has been applied to reservoirs containing highly-conductive fractures with complex geometrical configurations, while considering compositional fluid physics [108] and plastic and elastic deformation [126]. It has seen used as an upscaling technique [40, 127] and was successfully paired with multiscale methods for efficient flow simulation [27, 104, 115, 116]. However, the experiments presented in this work show that, in its current formulation, the model is not suitable in cases when the fracture permeability lies below that of the matrix. In addition, even when fractures coincide with the interfaces of matrix cells, the existing EDFM formulation still allows for independent flow leakage (i.e. disregarding the properties of the fracture placed between neighbouring matrix cells). To resolve these important limitations, this work proposes a new formulation for embedded fracture approaches, namely, the projection-based EDFM (pEDFM). The pEDFM is shown to successfully accommodate to lower-dimensional structural features with a wide range of permeability contrasts towards the matrix. This includes highly conductive fractures and flow barriers with small apertures, relative to the reservoir scale, which allows their representation as 2D plates. For the remainder of the chapter, they will be referred to, simply, as *fractures*, regardless of their conductive properties.

The devised pEDFM formulation retains the geometric flexibility of the classic EDFM procedure. More specifically, once the fracture and matrix grids are independently defined, and the cross-media communication points are identified, pEDFM adjusts the matrix-matrix and fracture-matrix transmissibilities in the vicinity of fracture networks. This ensures that the conductivity of the fracture networks, which can be several orders of magnitude below or above that of the matrix, are automatically taken into account when constructing the flow patterns. Finally, when fractures are explicitly placed at the interfaces of matrix cells, pEDFM automatically provides identical results to DFM.

The chapter is structured as follows. First, the governing equations are presented and discretized according to the pEDFM approach. Then, a series of numerical experiments are presented, targeted at validating pEDFM by comparing it to DFM (i.e. using unstructured grids with fractures being confined at the interfaces) and EDFM. The sensitivity of pEDFM with respect to fracture position and orientation, grid resolution and the conductivity contrast towards the matrix is studied extensively. Finally, the results are summarized, conclusions are drawn and possible directions for future work are discussed.

4.1. pEDFM formulation

In order to accommodate fractures with a wide range of conductivity contrasts towards the matrix, pEDFM extends the classic EDFM discretization of the governing flow equations by automatically scaling the matrix-matrix connections in the vicinity of fracture networks. At the same time, additional fracture-matrix connections are added in order to keep the system of equations well-posed in all possible scenarios. This is explained in detail in the following subsections.

4.1.1. Governing equations

The mass-conservation equations for isothermal multi-phase Darcy flow in fractured media, without compositional effects, can be written as

$$\left[\frac{\partial(\phi\rho_\alpha s_\alpha)}{\partial t} - \nabla\,.\,(\rho_\alpha \mathbf{k}\lambda_\alpha \cdot \nabla p)\right]^m = Q^m + [\rho_\alpha q]^{mw} + [\rho_\alpha q]^{mf} \quad \text{on } \Omega^m \subset \mathbb{R}^n \quad (4.1)$$

for the matrix (superscript ^{m}) and

$$\left[\frac{\partial(\phi\rho_\alpha s_\alpha)}{\partial t} - \nabla\,.\,(\rho_\alpha \mathbf{k}\lambda_\alpha \cdot \nabla p)\right]^f = Q^f + [\rho_\alpha q]^{fw} + [\rho_\alpha q]^{fm} \quad \text{on } \Omega^f \subset \mathbb{R}^{n-1} \quad (4.2)$$

for the fracture (superscript ^{f}) spatial domains. Here, ϕ is the rock porosity, p the pore pressure, $\mathbf{k}$ the 2nd order positive-definite absolute permeability tensor, here, considered diagonal and isotropic, while s_α, $\lambda_\alpha = k_{r,\alpha}/\mu_\alpha$, $k_{r,\alpha}$, μ_α and ρ_α are the phase saturation, mobility, relative permeability, viscosity and density, respectively. The q^{mw} and q^{fw} give the fluxes from / towards wells, q^{mf} and q^{fm} stand for the cross-media connections, while Q^m and Q^f are other source terms, e.g. due to capillary and gravity effects (neglected in this work), in the matrix and fracture domain, respectively.

4.1.2. Discretization

In order to solve the coupled system of Eqs. (4.1)-(4.2), independent grids are generated for the rock and fracture domains (See Fig. 4.1). This approach alleviates complexities related to grid generation, since, unlike in DFM, fractures do not need to be confined to the interfaces between matrix grid cells.

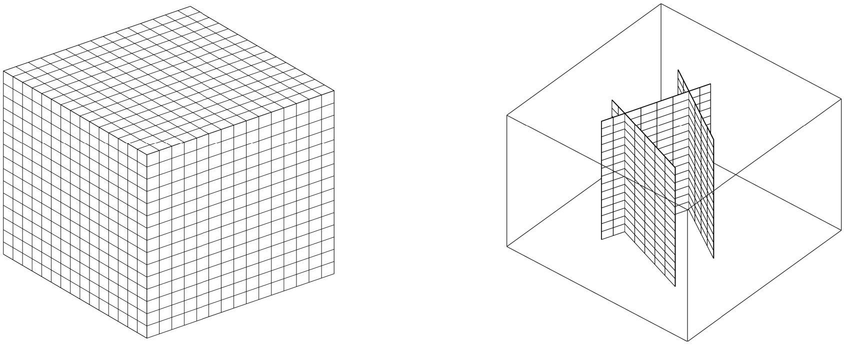
The advection term from Eqs. (4.1)-(4.2) is defined for each (matrix-matrix and fracture-fracture) grid interface, following the two-point-flux approximation (TPFA) finite volume discretization of the flux F_{ij} between each pair of neighbouring cells i and j as

$$F_{ij} \;= T_{ij}\;(p_i - p_j). \quad (4.3)$$

Here, $T_{ij} = \frac{A_{ij}}{d_{ij}} k_{ij}\lambda_{ij}$ is the transmissibility, A_{ij} is the interfacial area, d_{ij} is the distance between the cell centers, k_{ij} and λ_{ij} are the effective absolute permeability and fluid mobility at the interface between i and j, respectively (rock properties are harmonically averaged, while fluid properties are upwinded [128]).

The Peaceman well model [112] gives the source / sink terms between control volume Ω_l in the (matrix or fracture) domain and any perforating injection / production wells, respectively, as

$$F_{lw} = \int_{\Omega_l} q_{lw}\; dV = T_{lw}\;(p_l - p_w), \quad (4.4)$$



(a) Matrix grid

(b) Fracture grid

Figure 4.1: In pEDFM, independent grids are defined separately for the matrix and fracture domains.

where $T_{lw} = \mathrm{PI}_{lw}\ k_{lw}\lambda_{lw}$, PI is the well productivity index, while k_{lw} and λ_{lw} are the effective absolute permeability (harmonically averaged) and fluid mobility (upwinded) between the domain and the well, respectively.

The fracture-matrix coupling terms are modelled similar to [19, 27], i.e., for matrix cell i (with volume V_i) connected to a fracture cell f (of area A_f),

$$F_{if} = \int_{V_i} q_{if}^{mf}\, dV = T_{if}\ (p_f - p_i) \tag{4.5}$$

and

$$F_{fi} = \int_{A_f} q_{fi}^{fm}\, dA = T_{fi}\ (p_i - p_f), \tag{4.6}$$

where $T_{if} = \mathrm{CI}_{if}\ k_{if}\lambda_{if} = T_{fi}$ is the cross-media transmissibility. Once again, k_{if} and λ_{if} are the effective absolute permeability (harmonically averaged) and fluid mobility (upwinded) at the interface between matrix and fractures, respectively, while the CI_{if} is the conductivity index, defined as

$$\mathrm{CI}_{if} = \frac{S_{if}}{\langle d \rangle_{if}}, \tag{4.7}$$

where S_{if} is the surface area of the connection (to be further specified below) and $\langle d \rangle_{if}$ is the average distance between the points contained in the rock control volume V_i and the fracture surface A_f [19, 27], i.e.,

$$\langle d \rangle_{if} = \frac{1}{V_i} \int_{\Omega_i} d_{if}\ dv_i, \tag{4.8}$$

where d_{if} stands for the distance between finite volume dv_i and fracture plate. Appendix A gives an analytical method for its computation on 2D structured grids.

EDFM

Consider the fractured medium from Fig. 4.2, which is discretized on a structured grid. Let A_{if} be the area of intersection between fracture cell f and matrix volume i (highlighted in yellow in Fig. 4.2). The classical EDFM formulation [19, 27] defines the transmissibility as

$$T_{if} = \frac{2A_{if}}{\langle d \rangle_{if}} k_{if} \lambda_{if} \tag{4.9}$$

where, in this case, $S_{if} = 2A_{if}$ for computing CI in Eq. (4.7), while k_{if} and λ_{if} are the effective cross-media absolute permeability and mobility, respectively. The transmissibility of the matrix-matrix connections in the neighbourhood of the fracture (between control volumes i and j, k, respectively) are left unmodified from their TPFA finite volumes form, i.e

$$T_{ij} = \frac{A_{ij}}{\Delta x} k_{ij} \lambda_{ij} \qquad T_{ik} = \frac{A_{ik}}{\Delta y} k_{ik} \lambda_{ik}. \tag{4.10}$$

where A_{ij}, A_{ik} are the areas and k_{ij}, k_{ik}, $\lambda_{ij}, \lambda_{ik}$ the effective absolute permeabilities and mobilities are the corresponding matrix interfaces.

pEDFM

This work proposes an extension to the EDFM formulation, by modifying the matrix-matrix and fracture-matrix in the vicinity of fractures. This enables the development of a general embedded discrete fracture modeling approach (pEDFM), applicable in cases with any conductivity contrast between fractures and matrix. To this end, first a set of matrix-matrix interfaces is selected, such that they define a continuous projection path of each fracture network on the matrix domain (highlighted in red on the right side of Fig. 4.2). While, devising a generic algorithm for the construction of these paths are out-of-scope, it is important to ensure their continuity for each fracture network.

Consider fracture cell f intersecting matrix volume i on an n-dimensional structured grid over a surface area, A_{if}. Let $A_{if\perp x_e}$ be its corresponding projections on the path, along each dimension, $e = 1, \dots, n$ (depicted in red on the left side of Fig. 4.2). Also, let i_e be the matrix control volumes which reside on the opposite side of the interfaces affected by the fracture cell projections (highlighted in orange in Fig. 4.2). Then, the following transmissibilities are defined

$$T_{if} = \frac{A_{if}}{\langle d \rangle_{if}} k_{if} \lambda_{if}, \qquad T_{i_ef} = \frac{A_{if\perp x_e}}{\langle d \rangle_{i_ef}} k_{i_ef} \lambda_{i_ef} \quad \text{and} \quad T_{ii_e} = \frac{A_{ii_e} - A_{if\perp x_e}}{\Delta x_e} k_{ii_e} \lambda_{ii_e}, \tag{4.11}$$

where A_{ii_e} are the areas of the matrix interfaces hosting the fracture cell projections and k_{if}, k_{i_ef}, k_{ii_e}, λ_{if}, λ_{i_ef}, λ_{ii_e} are effective absolute permeabilities and fluid mobilities between the corresponding cells. Notice that the projected areas, $A_{if\perp x_e}$, are eliminated from the matrix-matrix transmissibilities and, instead, make the object of stand-alone connections between the fracture and the *non-neighbouring*

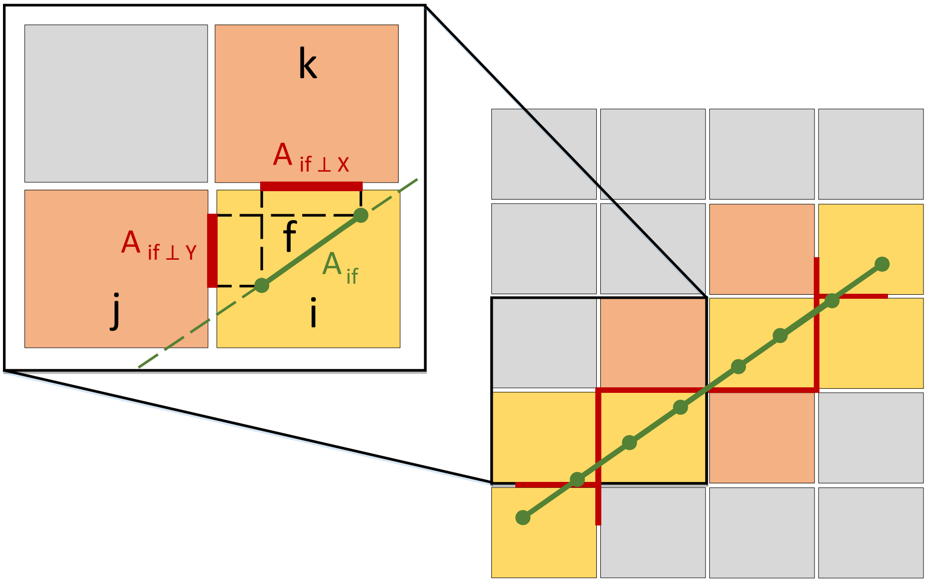


Figure 4.2: Illustration of pEDFM on a 2D structured grid. The matrix cells highlighted in yellow are connected directly to the fracture, as defined in the classic EDFM. The cells highlighted in orange take part in the additional non-neighbouring connections between fracture and matrix grid cells, as required by pEDFM.

(i.e. not directly intersected) matrix cells i_e. Also, the matrix-matrix connectivity T_{ii_e} will be eventually zero if the fracture elements (belonging to one or multiple fractures) cross through the entire matrix cell i.

Finally, note that, for fractures that are explicitly confined to lie along the interfaces between matrix cells, the pEDFM formulation, as given in Eq. (4.11), naturally reduces to the DFM approach on unstructured grids, while the EDFM does not.

Given the above TPFA finite-volume discretization of the advection and source terms from Eqs. (4.1)-(4.2), after applying backward Euler time integration, the coupled system is linearized with the Newton-Raphson scheme and solved iteratively.

4.2. Algebraic multiscale formulation (F-AMS-pEDFM)

This section briefly describes the application of the F-AMS procedure [104] to the pEDFM model. As previously discussed, given a computational domain with N_f fracture networks and N_w wells, F-AMS first superimposes two coarse grids on top of both the matrix and fracture fine-scale. The primal-coarse grid is a non-overlapping decomposition of the domain, inside which a fine-scale cell is selected as coarse node (Figs. 3.2(c) and 3.2(e)). By connecting the coarse nodes, a secondary overlapping coarse grid is obtained, which is called the dual-coarse grid (Figs. 3.2(d) and 3.2(f)). There exist N_{cm} and N_{dm} matrix primal-coarse and dual-coarse blocks and, similarly, each fracture network f_i contains N_{cf_i} and N_{df_i} fracture primal-coarse and dual-coarse blocks. Note that N_w (injection or production) wells exist in the domain, as driving forces for the flow.

F-AMS approximates the solution to the coupled system Eqs. (4.1)-(4.2), p, as a superposition of coarse-scale solutions ($\breve{p}$) using locally computed basis functions (Φ), i.e.

$$\overline{p}^m \approx \overline{p}'^m = \sum_{i=1}^{N_{cm}} \overline{\Phi}_i^{mm}\breve{p}_i^m + \sum_{i=1}^{N_f}\sum_{j=1}^{N_{cf_i}} \overline{\Phi}_j^{mf_i}\breve{p}_j^{f_i} + \sum_{k=1}^{N_w} \overline{\Phi}^{mw_k}\breve{p}^{w_k}, \tag{4.12}$$

for the matrix and

$$\overline{p}^f \approx \overline{p}'^f = \sum_{i=1}^{N_{cm}} \overline{\Phi}_i^{fm}\breve{p}_i^m + \sum_{i=1}^{N_f}\sum_{j=1}^{N_{cf_i}} \overline{\Phi}_j^{ff_i}\breve{p}_j^{f_i} + \sum_{k=1}^{N_w} \overline{\Phi}^{fw_k}\breve{p}^{w_k} \tag{4.13}$$

for the fractures, respectively. Since each well is assigned a single pressure value, their fine-scale and coarse-scale representations will be equivalent, i.e.

$$p^{w_i} = p'^{w_i} = \breve{p}^{w_i} \qquad \forall i \in \{1, \cdots, N_w\}. \tag{4.14}$$

Here, the basis functions associated with matrix coarse cells (i.e., Φ^{m*}) are Φ_i^{mm} for matrix-matrix effects, $\Phi_j^{mf_i}$ for the matrix-fracture coupling, and Φ^{mw_k} matrix-well interactions. These basis functions (interpolators) are employed in order to capture the effects of all the important factors (matrix, fractures, and wells) in the construction of a good approximation for the matrix pressure field $\overline{p}^m$. Similarly for fractures, Φ^{f*} consists of the contributions from the matrix Φ_i^{fm}, fractures $\Phi_j^{ff_i}$, and wells Φ^{fw_k}, if present.

In order to write the above in algebraic form, the values of the basis functions are assembled into the columns of the multiscale *prolongation operator*, $\mathcal{P}$ (of dimension $N_{\text{fine}} \times N_{\text{coarse}}$, where N_{fine} and N_{coarse} are the total number of fine- and coarse-scale control volumes, respectively), which reads

$$\mathcal{P} = \begin{bmatrix} \mathbf{\Phi}^{mm} & \mathbf{\Phi}^{mf} & \mathbf{\Phi}^{mw} \\ \mathbf{\Phi}^{fm} & \mathbf{\Phi}^{ff} & \mathbf{\Phi}^{fw} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}^{ww} \end{bmatrix}, \tag{4.15}$$

where $\mathbf{I}$ is the identity matrix.

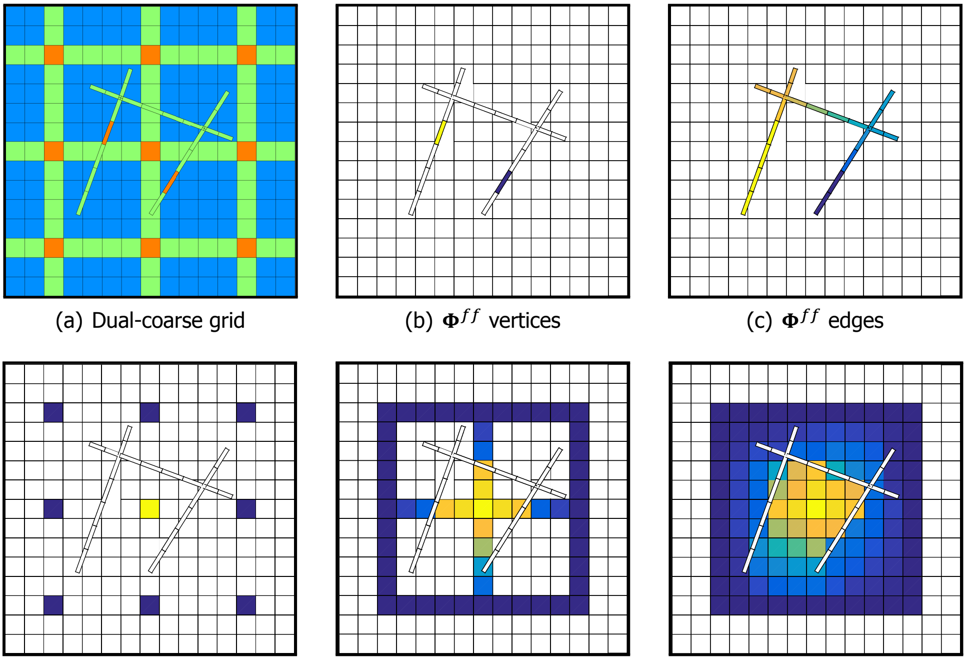
The basis function of each (matrix, fracture or well) coarse node i is computed with local support by following the wirebasket hierarchy [61, 63, 129]. This ensures that each basis function has a localized support around its corresponding coarse node.

For the purposes of the present work, the off-diagonal blocks accounting for the fracture-matrix coupling in the prolongation are neglected (i.e. $\mathbf{\Phi}^{mf}$ and $\mathbf{\Phi}^{fm}$

set to zero in Eq. (4.15)). This results in the Decoupled-AMS strategy from [104], where the basis functions corresponding to coarse nodes from one medium (e.g. matrix) are solved with no-flow boundary conditions to the other (i.e. fractures). More specifically, the incompressible pressure equation (see [96]) with no fracture-matrix coupling is written

$$-\overline{\nabla}\,.\,(\mathbf{k}^m\lambda^m\cdot\nabla\Phi_c^{mm}) + \sum_{j\in\text{perf}_w^m} \text{PI}_j\mathbf{k}^m\lambda^m\,(\Phi_c^{mm} - \underbrace{\Phi_c^{wm}}_{=0}) = 0, \tag{4.16}$$

for the matrix basis functions ($\mathbf{\Phi}^{mm}$), where the divergence operator $\overline{\nabla}\,.$ was modified to neglect the connections removed by the localization assumption. Here, c denotes the coarse node (i.e. the column in $\mathcal{P}$) to which the basis function belongs to, and perf_w^m is the set of wells perforating the matrix control volume under consideration. A step-by-step illustration of this computation is shown at the bottom of Fig. 4.3.


(a) Dual-coarse grid

(b) $\mathbf{\Phi}^{ff}$ vertices

(c) $\mathbf{\Phi}^{ff}$ edges

(d) $\mathbf{\Phi}^{mm}$ vertices

(e) $\mathbf{\Phi}^{mm}$ edges

(f) $\mathbf{\Phi}^{mm} faces$

Figure 4.3: Illustration of a step-by-step basis function calculation using the Decoupled-AMS strategy on the pEDFM model.

Note that, in the pEDFM formulation, when a matrix cell is fully crossed by fractures, the transmissibility between itself and one or more of its neighbours can become 0. This can lead to rank deficient linear systems when computing basis functions under the localization assumption (i.e. along the edges, faces or interiors), if the coupling approach presented here – which corresponds to the Decoupled-AMS approach of [104] – is followed. In order to address this issue, an upper and lower threshold can be imposed on the transmissibilities when computing basis functions (for clarity, the fine-scale system remains unaffected), as previously suggested in the literature [90]. Let T_{avg} be the average of the transmissibilities of all interfaces present in the reservoir. Then, the basis function transmissibilities are not allowed to lie below $T_{\text{min}} = T_{\text{avg}}/T_{\text{thresh}}$ or above $T_{\text{max}} = T_{\text{avg}} * T_{\text{thresh}}$. In the experiments presented in this work, values of $T_{\text{thresh}} = 10$ and 10^2 are used, which limit the transmissibility contrast captured by the basis functions to roughly 10^2 and 10^4, respectively. Note that, if the Coupled-AMS or Frac-AMS coupling strategies are followed within F-AMS [104], this issue does not exist. Consideration of the effect of these coupling approaches is the subject of ongoing research.

Similar to the matrix,

$$-\overline{\nabla}\,.\left(\mathbf{k}^f\lambda^f\cdot\nabla\Phi_c^{ff}\right)+\sum_{j\in\text{perf}_w^f}\text{PI}_j\mathbf{k}^f\lambda^f\,(\Phi_c^{ff}-\underbrace{\Phi_c^{wf}}_{=0})=0, \tag{4.17}$$

is solved for the fracture basis functions ($\mathbf{\Phi}^{ff}$), as shown in the top of Fig. 4.3, where perf_w^f denotes the set of wells perforating the fracture cell for which Eq. (4.17) is written.

F-AMS extends the consideration of wells in the multiscale solution [68] to 3D fractured porous media. To this end, each well is represented as a single vertex on the dual-coarse grid and a well function is computed locally by solving

$$-\overline{\nabla}\,.(\mathbf{k}^m\lambda^m\cdot\nabla\Phi_c^{mw})+\sum_{j\in\text{perf}_w^m}\text{PI}_j\mathbf{k}^m\lambda^m\,(\Phi_c^{mw}-\underbrace{\Phi_c^{ww}}_{=\delta_{ij}})=0 \tag{4.18}$$

or

$$-\overline{\nabla}\,.\left(\mathbf{k}^f\lambda^f\cdot\nabla\Phi_c^{fw}\right)+\sum_{j\in\text{perf}_w^f}\text{PI}_j\mathbf{k}^f\lambda^f\,(\Phi_c^{fw}-\underbrace{\Phi_c^{ww}}_{=\delta_{ij}})=0, \tag{4.19}$$

depending on the domain to which its neighbouring dual blocks belong to. Finally, the values of all well functions are assembled into the $\mathbf{\Phi}^{fw}$ (for the fracture domain) and $\mathbf{\Phi}^{mw}$ (for the matrix domain) sub-blocks of the prolongation operator, respectively.

Alongside prolongation, in order to obtain the coarse-scale pressure system, the restriction operator $\mathcal{R}$ (i.e., map from fine to coarse scale, with dimension $N_{\text{coarse}} \times N_{\text{fine}}$) needs to be defined. Among the options supported by F-AMS [104], this work focuses on the MSFV restriction operator, $\mathcal{R}^{FV}$, where the entry at row i and column j is 1 if the fine-scale cell j (either from the matrix, fractures, or wells) belongs to primal-coarse block i, or 0 otherwise. This choice allows a conservative flux reconstruction stage, in order to facilitate the solution of the transport equations [43, 73] when F-AMS is used as approximate solver.

Although this allows F-AMS to be used as a single-sweep multiscale solver, where the approximate solution, p', is used with no iterations, previous studies have shown that an iterative procedure is needed for highly-heterogeneous reservoirs [56]. To this end, one needs to pair the F-AMS multiscale step with a fine-scale smoother, which ensures error reduction to any desired level. During the experiments that follow, the smoothing operator is an approximation of the inverse of the fine-scale linear operator, $\mathbf{A}^{(\nu)}$, via ILU(0) decomposition [57].

4.3. Numerical results

This section presents the results of numerical experiments of single- and two-phase incompressible flow through two- and three-dimensional fractured media. Their aim is to validate pEDFM, whose formulation was presented in the previous section, and study its sensitivity to fracture position, grid resolution and fracture-matrix conductivity contrast, respectively. The reference solution for these studies is obtained on a *fully resolved* grid, i.e. where the size of each cell is equal to the fracture aperture. This allows the following model error measurement,

$$\frac{\|\epsilon\|_1}{N_{\text{coarse}}} = \frac{\sum_1^N |p'_{\text{fine}} - p_{\text{coarse}}|}{N_{\text{coarse}}} \tag{4.20}$$

where N_{coarse} is the number of grid cells used by pEDFM and p'_{fine} is the corresponding fully-resolved pressure, interpolated to the coarse scale, if necessary. Some of the experiments were repeated for the classic EDFM, as well as unstructured DFM, for comparison purposes.

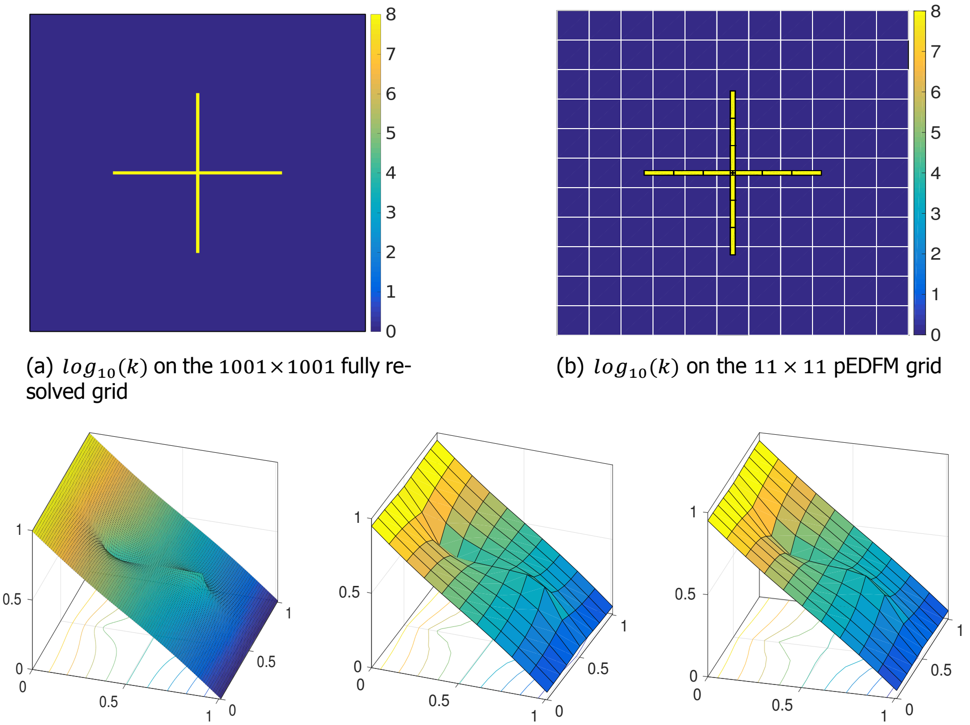
For simplicity, but without loss of generality, the flow in these experiments is driven by Dirichlet boundary conditions, instead of injection and production wells, while capillary and gravity effects are neglected.

Finally, the simulations were performed using the DARSim 1 in-house simulator, using a sequentially implicit strategy for the multiphase flow cases.

4.3.1. pEDFM validation

In order to validate pEDFM as a fine-scale model suitable to accommodate fractures with a wide range of permeabilities, a 2D homogeneous domain ($k^m = 1$) is considered, having a +-shaped fracture network, located in the middle. In order to drive the incompressible single-phase flow, Dirichlet boundary conditions with non-dimensional pressure values of $p = 1$ and $p = 0$ are imposed on the left and right boundaries of the domain, respectively, while the top and bottom sides are subject to no-flow conditions.

As shown in Fig. 4.4, the study is first conducted in a scenario where the fractures are 8 orders of magnitude more conductive than the matrix. The reference solution, in this case, is computed on a 1001×1001 structured Cartesian grid. From the bottom of Fig. 4.4, it is clear that both EDFM and pEDFM, on a coarser 11×11 domain, can reproduce the behaviour of the flow as dictated by the highly-conductive embedded fracture network.



(a) $log_{10}(k)$ on the 1001×1001 fully resolved grid

(b) $log_{10}(k)$ on the 11×11 pEDFM grid

(c) Fully resolved pressure

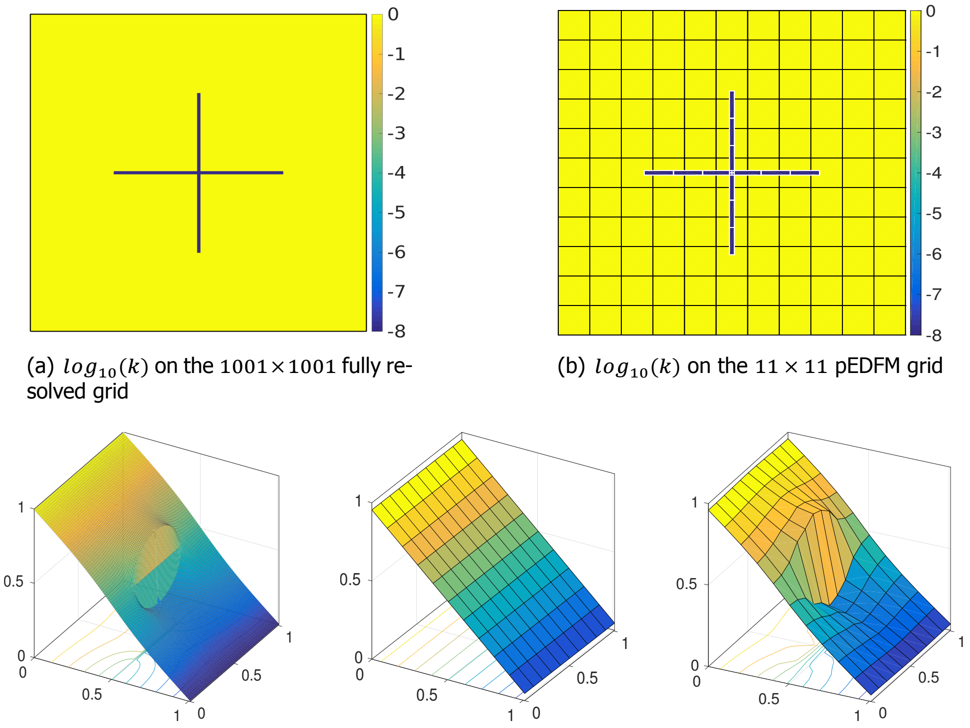
(d) EDFM pressure

(e) pEDFM pressure

Figure 4.4: Fully-resolved (c) EDFM (d) and pEDFM (e) pressure solutions in a homogeneous reservoir containing a +-shaped highly-conductive fracture network (top).

As shown in Fig. 4.5, the same experiment was rerun for the case where the fracture permeability lies 8 orders of magnitude below that of the host matrix. The results expose the limitations of EDFM, where the impermeable fractures are simply by-passed by the flow through the (unaltered) matrix, resulting in a pressure field corresponding to a reservoir with homogeneous (non-fractured) permeability. On the other hand, through its new formulation, pEDFM is able to reproduce the effect of the inhibiting flow barrier (see bottom of Fig. 4.5), confirming its applicability to this case.

These experiments confirm that pEDFM is a suitable extension of EDFM to a wider range of geological scenarios, being able to reproduce the correct flow behaviour in the presence of both high and low permeable fractures, embedded in the porous matrix.



(a) $log_{10}(k)$ on the 1001×1001 fully resolved grid

(b) $log_{10}(k)$ on the 11×11 pEDFM grid

(c) Fully resolved pressure

(d) EDFM pressure

(e) pEDFM pressure

Figure 4.5: Fully-resolved (c) EDFM (d) and pEDFM (e) pressure solutions in a homogeneous reservoir containing a +-shaped nearly impermeable flow barrier (top).

4.3.2. Sensitivity to the fracture position within the grid cell

Given that pEDFM typically operates on much coarser grids than the fully resolved case, it is of interest to elicit its sensitivity to the fracture position within the host grid cell. To this end, the +-shaped fracture configuration is considered; the reference solution is computed on a $3^7\times3^7$ (i.e., 2187×2187) cell grid, while pEDFM grid operates at 10×10 resolution.

From Fig. 4.4, it appears that in the case when the fracture network is highly conductive, the horizontal fracture is the one that dictates the flow. Consequently, successive simulations are conducted for both EDFM and pEDFM, while moving the horizontal fracture from top to bottom, as shown in Fig. 4.6. Their accuracy is measured using Eq. (4.20).

The results show that EDFM is more accurate when fractures are placed at the cell center, rather than when they are close to the interface. However, once the fracture coincides with the interface, EDFM connects it to both matrix cells (each, with a CI calculated using $S_{if}=A_{if}$ in Eq. (4.7), instead of $2A_{if}$ as was the case in Eq. (4.9)), thus explaining the abrupt dip in error. In contrast, the pEDFM error

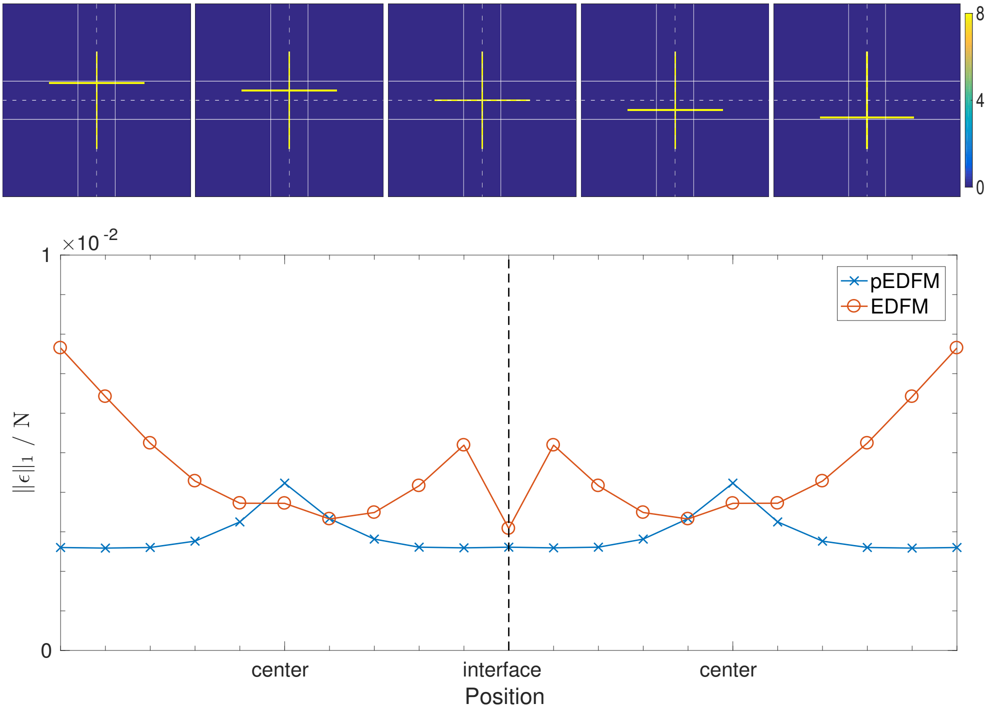


Figure 4.6: Sensitivity of pEDFM to the position of highly conductive fractures, embedded within the matrix grid cells. To this end, the horizontal fracture of the +-shaped network is successively moved from top to bottom over a 2 grid cell window (top), while monitoring the pressure mismatch towards the corresponding fully resolved simulation (bottom).

attains its peak when fractures are placed at the cell centers and does not exhibit any jumps over the interface. The error of both methods lies within similar bounds (still pEDFM is more accurate) showing that they applicable to the case when fractures are highly conductive. The consistent aspect of pEDFM is that, its results for the case when fractures coincide with the matrix interfaces, its results are identical to the DFM method, while –as explained before– this is not the case for EDFM.

When the network is nearly impermeable, the location of the vertical fracture is critical to the flow (Fig. 4.5). As such, for the purposes of the current experiment, it will be shifted from left to right, as shown in Fig. 4.7. The resulting error plot shows a dramatic increase for EDFM, when compared to Fig. 4.7, due to its inability to handle fractures with conductivities that lie below that of the matrix. pEDFM, on the other hand shows a similar behaviour and error range as was observed in the case with highly conductive fractures, i.e., it retains its high accuracy.

These results show a promising trend for pEDFM, which is able to maintain reasonable representation accuracy of the effect of the embedded fractures. The slight increase in error for fractures placed near the matrix cell centers may be mitigated by employing moderate local grid refinements.

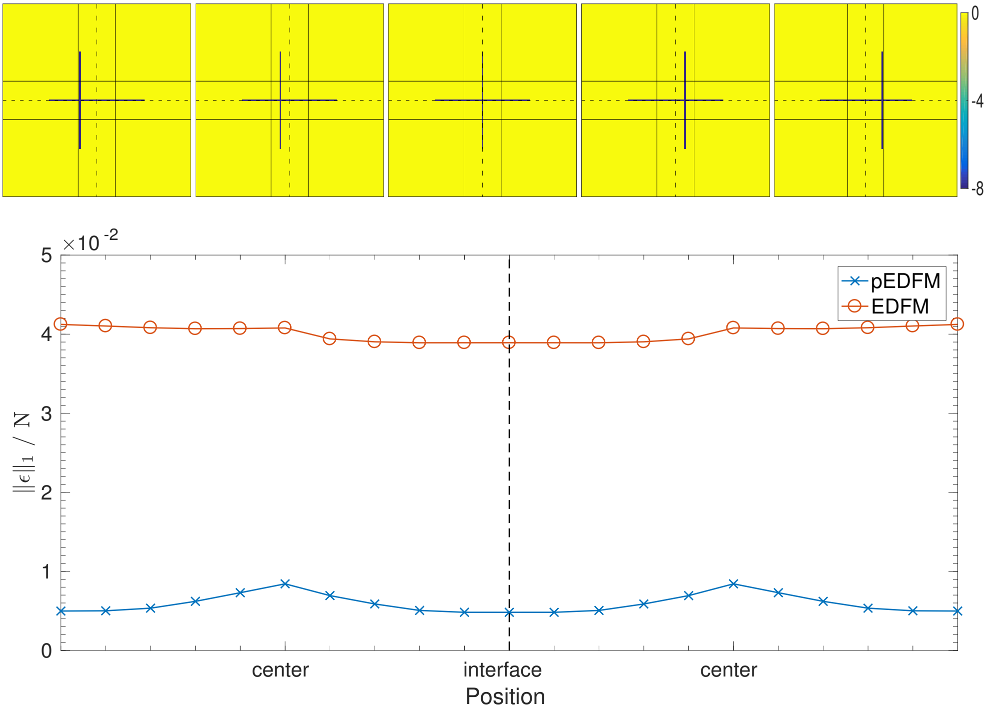


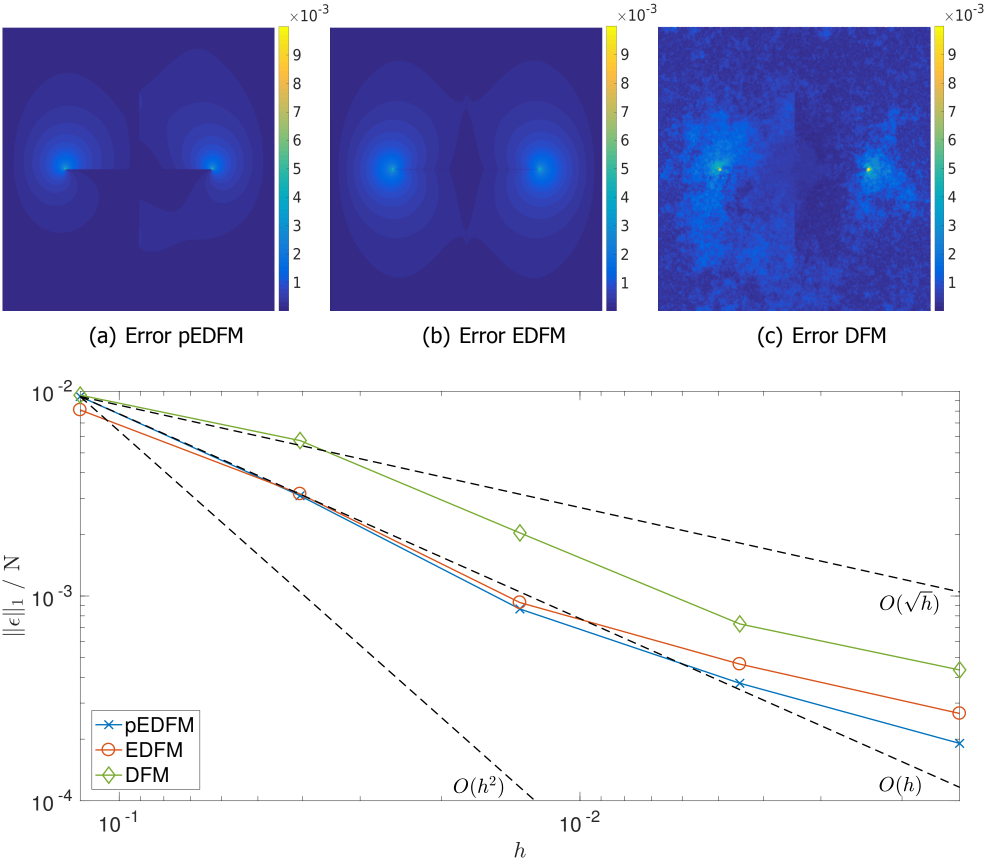
Figure 4.7: Sensitivity of pEDFM to the position of nearly impermeable fractures, embedded within the matrix grid cells. The vertical fracture of the +-shaped network is successively moved from left to right over a 2 grid cell window (top), while monitoring the pressure mismatch towards the corresponding fully resolved simulation (bottom).

4.3.3. Sensitivity to the grid resolution

Another important factor in assessing the quality of an embedded fracture model is its order of accuracy with respect to the grid resolution. A series of nested matrix grids for the +-shaped fracture test case of Figs. 4.4 and 4.5 was constructed. The number of cells over each axis is gradually increased using $N_x = N_y = 3^i$ formula, where $i = 2, 3, \dots$, up to a reference grid resolution, where $i = 7$. At the same time, the fracture grid is also refined accordingly such that its step size approximately matches the one in the matrix, $h = \Delta x = \Delta y$. The measure of accuracy for this case is similar to Eq. 4.20, where, this time, no interpolation is necessary, since the cell centroids are inherited from one level to another in the nested grid hierarchy.

For a better comparison, alongside pEDFM and EDFM, the same sequence of geological scenarios was simulated using DFM on a 2D unstructured grid [24], where the number of triangles was tweaked to match $N = N_x \times N_y$ as closely as possible and without imposing any preferential grid refinement around the fractures.

The results of this study, in the case when the fractures are highly conductive, are depicted in Fig. 4.8. It follows that all three methods experience a linear de-



(a) Error pEDFM

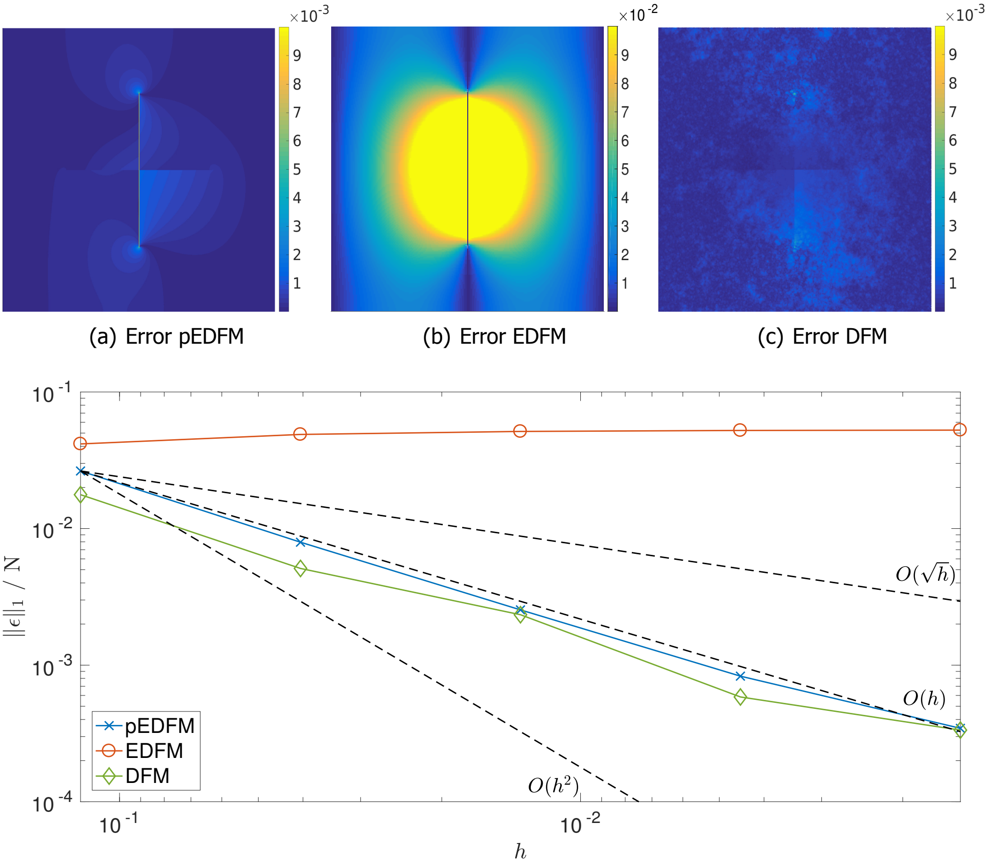
(b) Error EDFM

(c) Error DFM

Figure 4.8: Grid resolution sensitivity of pEDFM, EDFM and DFM on the case with highly conductive fractures. The sequence of plots on the top shows pressure error maps for the three methods, when $N_x = N_y = 3^6$ (or $h = 0.0015$) holds for pEDFM and EDFM, and DFM employs comparable total number of elements.

cay in error with increasing grid resolution. The three error snapshots, which were taken when $N_x = N_y = 3^6$ (or $h = 0.0015$), show that the pressure mismatch is mainly concentrated around the tips of the horizontal fracture, which represent the network's inflow and outflow points, respectively. For EDFM, the error decays radially for points further away from these fracture tips. For pEDFM, the contour curves are slightly skewed, depending on the choice between upper and lower matrix interfaces for the fracture projection (both are equally probable since the horizontal fracture crosses the grid cell centroids). Finally, for DFM the error distribution shows some heterogeneity, which is a consequence of using unstructured grids in a medium which, except for the neighbourhood of the fractures, is homogeneous.

The scenario when the fracture network is considered almost impermeable can not be properly handled by EDFM, regardless of which grid resolution is used (Fig.



(a) Error pEDFM (b) Error EDFM (c) Error DFM

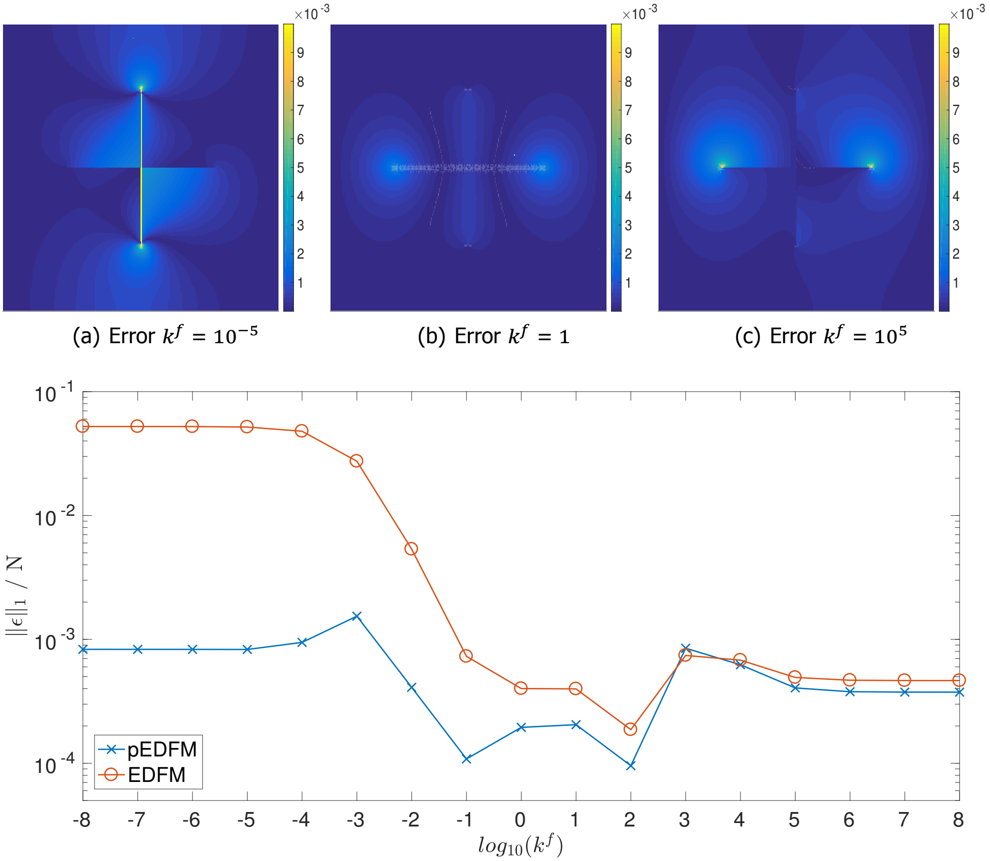
Figure 4.9: Grid resolution sensitivity of pEDFM, EDFM and DFM on the case with nearly impermeable fractures. The sequence of plots on the top shows pressure error maps for the three methods, when $N_x = N_y = 3^6$ (or $h = 0.0015$) holds for pEDFM and EDFM, and DFM employs comparable total number of elements.

4.9). This serious limitation is, once again, successfully overcome by using pEDFM, which, similar to DFM, maintains its linear scalability with grid refinement on this challenging test case. The error snapshots depict that, this time, the pressure is inaccurate around the tips, as well as the body, of the vertical barrier. This can be explained by the fact that an embedded model on a coarse grid can have difficulty in placing the sharp discontinuity in the pressure field at exactly the right location. Still, the pressure mismatch decays with increasing grid resolution, suggesting that local grid refinements around highly contrasting fractures can benefit pEDFM, in a similar manner to DFM.

To conclude, pEDFM shows a similar convergence behaviour, in terms of grid resolution, to the widely used DFM approach. This confirms that, in order to diminish the model representation error, moderate local grid refinements can be applied near fractures.

4.3.4. Sensitivity to the fracture-matrix conductivity contrast

This last sensitivity study is aimed at determining the response of pEDFM while changing the conductivity contrast between the +-shaped fracture network ($k^f = 10^{-8}, \dots, 10^{8}$) and the matrix ($k^m = 1$). To this end, a coarse grid resolution of $N_x = N_y = 3^5$ was used and the resulting pressure was compared to that from the reference case, where $N_x = N_y = 3^7$, using Eq. 4.20.



(a) Error $k^f = 10^{-5}$

(b) Error $k^f = 1$

(c) Error $k^f = 10^{5}$

Figure 4.10: Sensitivity of pEDFM to the fracture-matrix conductivity contrast on the +-shaped fracture network case with a grid resolution of $3^5 \times 3^5$. The sequence of plots on the top show pressure error maps for pEDFM, when $k^f = 10^{-5}$, 1 and 10^{5}, respectively.

The results are depicted in Fig. 4.10 and are in line with the conclusions from previous sections. Namely, for fracture log-permeabilities on the positive side of the spectrum, the results of EDFM and pEDFM are in agreement. As the permeability contrast passes 5 orders of magnitude, the pressure error plateaus, since, at beyond this stage, the fractures are the main drivers of the flow. However, for fracture permeabilities close to or below that of the matrix, the error of EDFM increases dramatically. pEDFM, on the other hand is able to cope with these scenarios, due to its formulation, its behaviour showing an approximately symmetric trend, when compared to that of the positive side of the axis.

The snapshots on the top of Fig. 4.10 , taken for lower, similar and higher fracture permeabilities w.r.t. the matrix, show the error in the pressure produced by pEDFM. It is clear that the model inaccuracy is concentrated around the tips of fractures which actively influence the flow. Also note that there is a small error even in the case when $k^f = k^m$, since the pEDFM discretization (Section 4.1) is slightly different than that of a homogeneous reservoir. Of course, this result is only presented here for academic purposes – in realistic scenarios, when the contrast is not high enough, such fractures can be homogenized into the matrix field.

This concludes the sensitivity studies conducted in this chapter. The test cases presented in the following subsections are meant to test the applicability of pEDFM to more complex 2D and 3D fractured media.

4.3.5. Time-lapse 2D multiphase results

This set of experiments is aimed at determining the performance of pEDFM in multiphase flow scenarios on 2D porous media with increasingly complex fracture geometries and heterogeneities.

Homogeneous matrix

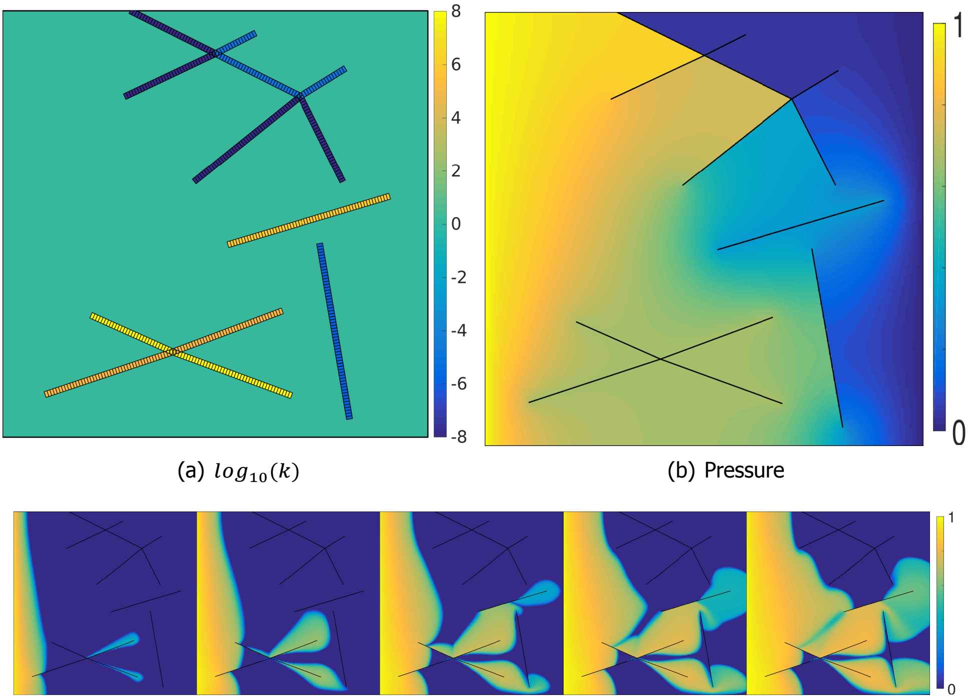
pEDFM is first applied in an incompressible 2-phase flow scenario through a 2D homogeneous domain which is crossed by a set of fractures with heterogeneous properties, as shown in Fig. 4.11. The boundary conditions are similar to those used for previous experiments, namely Dirichlet with non-dimensional values of $p = 1$ and $p = 0$ on the left and right edges, respectively, while the top and bottom sides are subject to no-flow conditions.

The low permeable fractures inhibit the flow, leaving only two available paths: through the middle of the domain and along the bottom boundary. As can be seen in the time-lapse saturation maps presented in Fig. 4.11, the front, indeed, respects these embedded obstacles. The injected fluid is mostly directed through the permeable X-shaped network and surpasses the vertical barrier, in the lower right part of the domain, on its way to the production boundary.

This result reinforces the conclusion that the conservative pressure field obtained using pEDFM leads to transport solutions which honour a wide range of matrix-fracture conductivity contrasts.

Heterogeneous matrix

The following experiment compares the behaviour of EDFM and pEDFM for simulating 2-phase incompressible flow through a 2D porous medium with het-



(a) $log_{10}(k)$

(b) Pressure

(c) 0.1 PVI

(d) 0.2 PVI

(e) 0.3 PVI

(f) 0.4 PVI

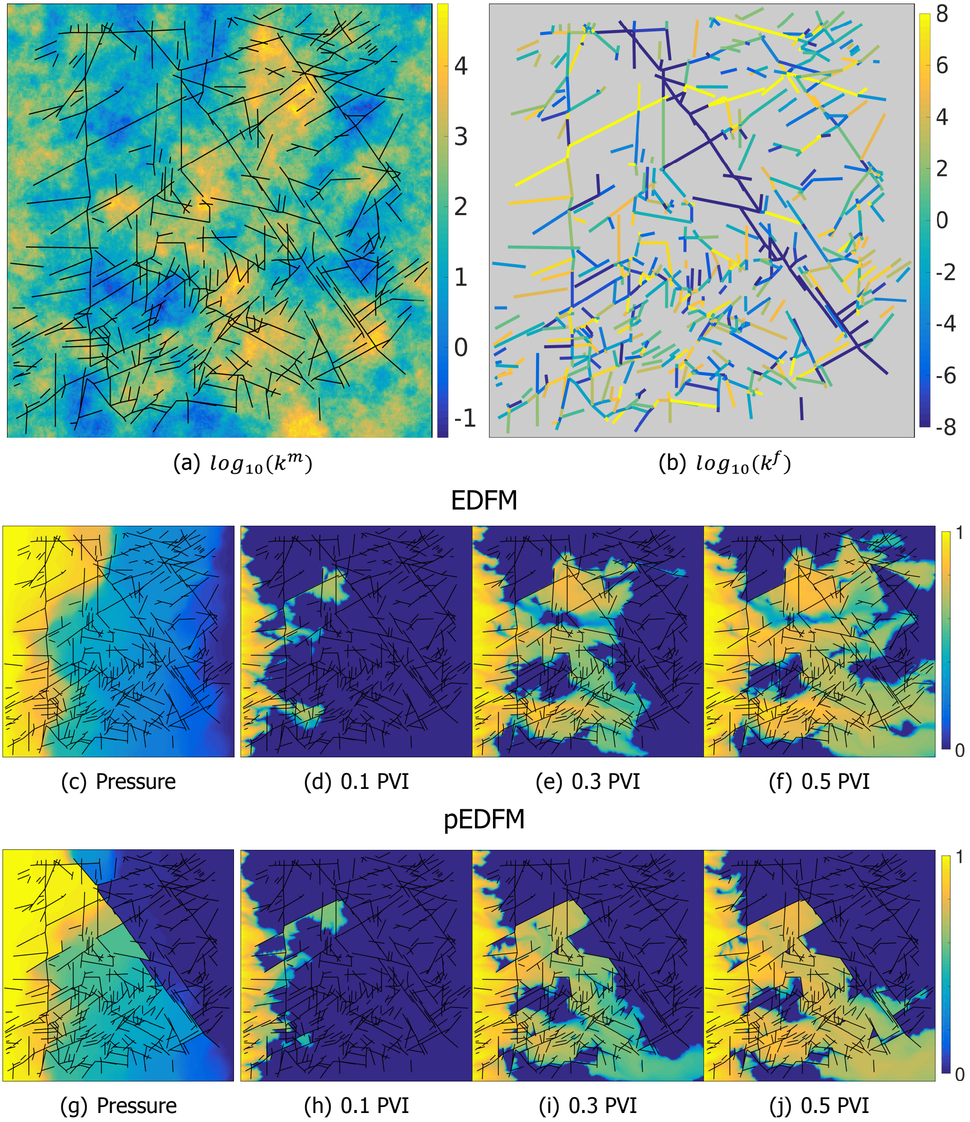
(g) 0.5 PVI

Figure 4.11: Fracture permeabilities (top-left), pressure field (top-right) and time-lapse saturation results (bottom) on a 2D test case with homogeneous matrix conductivity, under incompressible 2-phase flow conditions.

erogeneous (i.e. patchy) matrix permeability (Fig. 4.12). The interplay between the large- (matrix-matrix) and small-scale (fracture-matrix) conductivity contrasts raises additional numerical challenges [130] and is a stepping stone in assessing the model's applicability to realistic cases.

The embedded fracture map used for this test case (top of Fig. 4.12) is based on the Brazil I outcrop from [131, 132]. The conductivities of the fractures forming the North-West to South-East diagonal streak, were set to 10^{-8}, thus creating an impermeable flow barrier across the domain (noticeable in dark blue on the top-right of Fig. 4.12). For the rest of the fractures, permeabilities were randomly drawn from a log-uniform distribution supported on the interval $[10^{-8}, 10^{8}]$. Finally, similar to previous experiments, fixed pressure boundary conditions $p = 1$ and $p = 0$ are set on the left and right edges, respectively, while the top and bottom sides are subject to no-flow conditions.

The middle row of plots from Fig. 4.12 show the pressure field and time-lapse saturation results obtained using EDFM. Note that the injected fluid is allowed to bypass the diagonal flow barrier, towards the production boundary. This, once again shows the limitation of EDFM, which is only able to capture the effect of fractures



(a) $log_{10}(k^m)$

(b) $log_{10}(k^f)$

(c) Pressure

(d) 0.1 PVI

(e) 0.3 PVI

(f) 0.5 PVI

(g) Pressure

(h) 0.1 PVI

(i) 0.3 PVI

(j) 0.5 PVI

Figure 4.12: Heterogeneous matrix and fracture permeability maps (top), pressure and time-lapse saturation results for EDFM (middle) and pEDFM (bottom) on a 2D densely fractured test case, under incompressible 2-phase flow conditions.

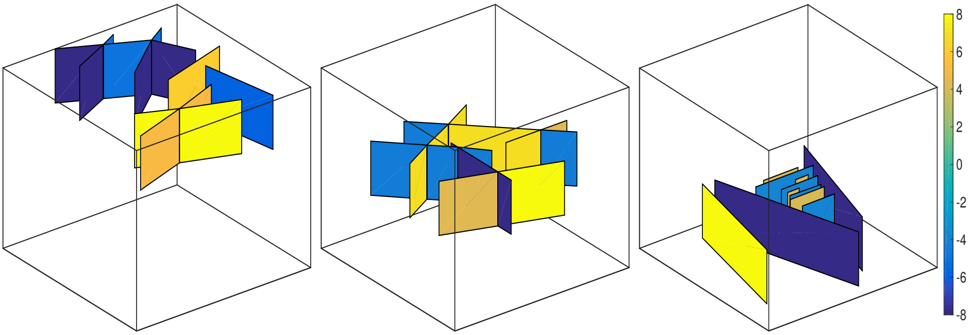
with permeabilities higher than the matrix. However, by disregarding flow barriers, EDFM delivers an overly optimistic and non-realistic production forecast.

In contrast to EDFM, the pressure field obtained using pEDFM shows sharp discontinuities (bottom-left of Fig. 4.12). The accompanying saturation plots confirm that the injected phase is confined by the diagonal barrier and forced to flow through the bottom of the domain, thus significantly delaying its breakthrough towards the production boundary.

These results confirm that pEDFM outperforms to EDFM, due to its applicability in cases with complex and dense fracture geometries and in the presence of matrix heterogeneities.

4.3.6. Comparison between pEDFM and unstructured DFM

Finally, a test case on a 3D domain containing 3 layers of fractures, stacked along the Z axis (Fig. 4.19) is conducted. The top layer is a vertically extruded version of the 2D fracture map from Fig. 4.11. The second layer consists of a single fracture network whose intersecting plates have highly heterogeneous properties. Finally, the third layer is represented by 3 large individual plates, with a cluster of small parallel fractures packed in between.



(a) $log_{10}(k^f)$ layer 1   (b) $log_{10}(k^f)$ layer 2   (c) $log_{10}(k^f)$ layer 3

Figure 4.13: 3D porous medium containing 3 layers of fractures with heterogeneous properties.

In this scenario, the incompressible single-phase flow is driven from the left boundary, where the pressure is set to the non-dimensional value of $p = 1$, towards the right, where $p = 0$, while all the other boundaries of the domain are subject to no-flow conditions. No other source terms are present and gravity and capillary effects are neglected. The results of pEDFM, on a matrix grid with $N_x = N_y = N_z = 100$ and a total of 23381 fracture cells, are compared to those obtained using DFM on an unstructured grid (Fig. 4.14), where the number of tetrahedra (matrix) and triangles (fractures) were chosen to approximately match the degrees of freedom on the structured grid.

The two pressure fields are plotted in Fig. 4.15 using iso-surfaces for equidistant

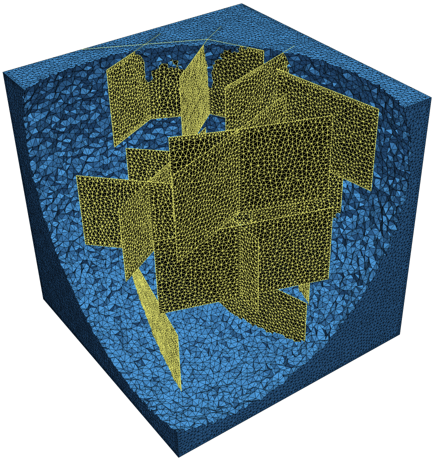
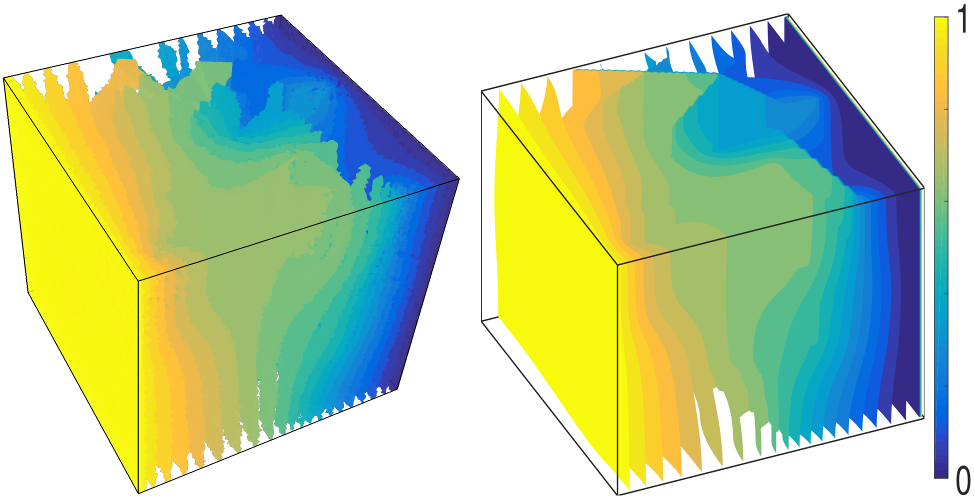


Figure 4.14: Fracture-conforming unstructured grid constructed by DFM for the 3D test case.



(a) DFM pressure

(b) pEDFM pressure

Figure 4.15: Comparison between the pressures obtained using pEDFM and unstructured DFM (using similar grid resolutions) for a 3D incompressible single-phase test case with 3 layers of heterogeneous fractures.

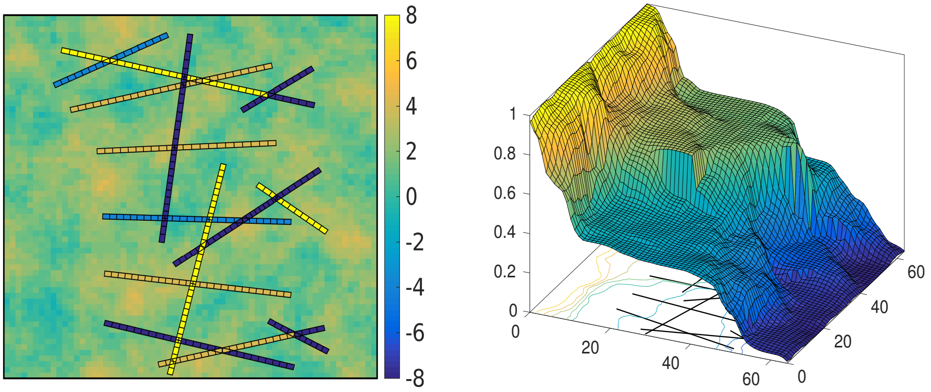
values, and are in good agreement, for decision-making purposes. This numerical experiment shows that pEDFM has good potential for field-scale application.

4.3.7. F-AMS-pEDFM convergence study

For the experiments presented in this section, the pEDFM pressure system is resolved using the iterative F-AMS algebraic multiscale method. As mentioned before, the basis functions were computed according to the Decoupled-AMS strategy [104], where the linear systems (for edges, faces and interiors) were constructed by imposing a $T_{thresh} = 10$ threshold on the transmissibility, as explained earlier. These systems, along with the resulting coarse-scale pressure system, are resolved using a direct solver (based on LU decomposition). Finally, F-AMS-pEDFM is iterated as preconditioner to GMRES, while using 5 iterations of ILU(0) as second-stage smoother [see 96, for a thorough sensitivity study on smoother iterations], until a residual 2-norm of 10^{-6} is reached.

Consider the 2D reservoir from Fig. 4.16 with heterogeneous (patchy) permeability containing a highly contrasting embedded fracture network (with conductivities up to 8 orders of magnitude above or below the matrix average). The resulting fine-scale pressure solution, by imposing a pressure value of 1 along the West and 0 along the East boundary, is shown in Fig. 4.16(b).

In order to employ F-AMS-pEDFM, a coarsening ratio of 8×8 was considered in the matrix, while the number of coarse-scale DOF in the fracture network is varied. The resulting pressure solution after 1 F-AMS-pEDFM iteration (without any smoothing) is shown in Fig. 4.17 for three different fracture coarsening scenarios. Note that, by increasing the number of coarse nodes (and thus, basis functions) used to represent the pressure in the highly heterogeneous fracture network, a better approximation is achieved (in comparison to the target pressure field from Fig. 4.16(b)).



(a) $log_{10}(k)$ (b) Pressure

Figure 4.16: Permeability (a) and pressure solution (b) in a 2D heterogeneous reservoir with an embedded fracture network. Each fracture plate was randomly assigned a permeability between $[-8, 8]$ on the log_{10}-scale. The pressure is fixed to values of 1 and 0 on the West and East boundaries, respectively.

This behaviour is further confirmed in Fig. 4.18, which shows that a smaller

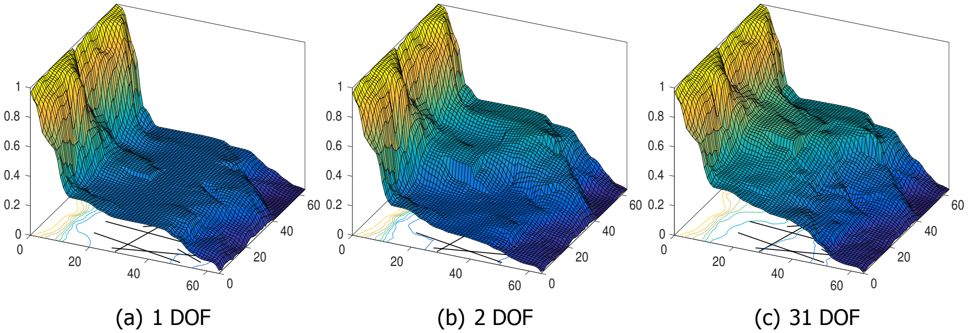


Figure 4.17: Pressure solution after a single F-AMS-pEDFM iteration, where different numbers of coarse-scale DOF have been considered in the fracture network. A matrix coarsening ratio of 8×8 was used in all cases.

number of linear solution iterations is necessary to achieve the convergence goal if a more refined fracture coarse grid is employed.

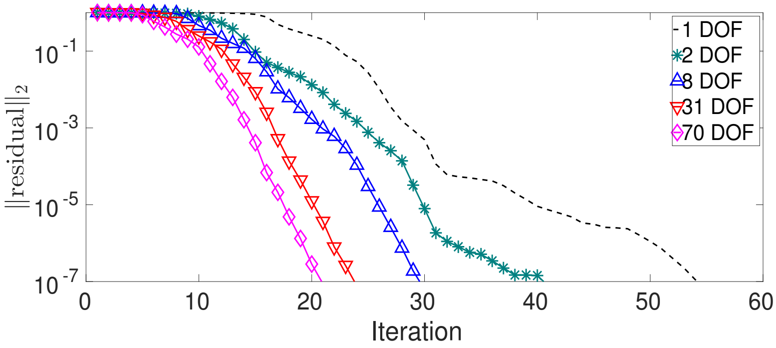
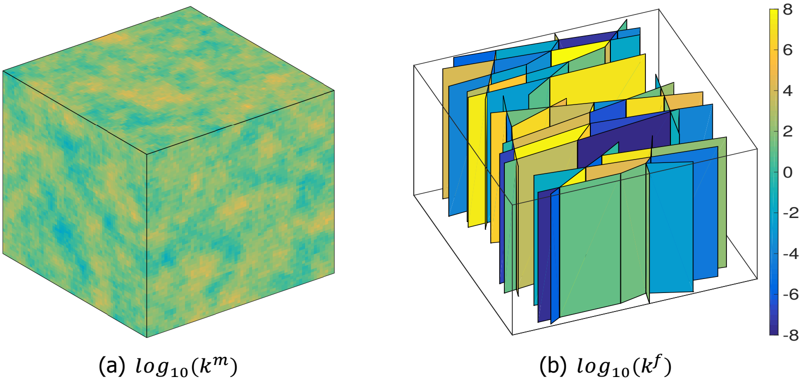


Figure 4.18: Convergence rate of F-AMS-pEDFM with different number of coarse-scale DOF in the fracture network.

To conclude, these experiments show that F-AMS-pEDFM is a convergent iterative solution strategy for highly heterogeneous fractured porous media. The coarsening strategy used in each media has a significant impact on the convergence rate. As such, devising an algorithm for generation of optimal primal- and dual-coarse grid for realistic reservoirs is an important subject for further research.

4.3.8. F-AMS-pEDFM scalability with increasing fracture density

This final section investigates the performance of F-AMS-pEDFM on a 3D heterogeneous reservoir containing 3 embedded fracture networks whose sizes are increased over three stages. The (patchy) matrix and (8-fold contrasting) fracture permeability distribution (in the final stage) are shown in Fig. 4.19. At the same time, the pressure solution and fracture network during each stage is given in Fig. 4.20.



(a) $log_{10}(k^m)$

(b) $log_{10}(k^f)$

Figure 4.19: Permeability of a 3D heterogeneous reservoir with 3 embedded fracture networks. Each fracture plate was randomly assigned a permeability between $[-8, 8]$ on the log_{10}-scale. The pressure is fixed to a value of 1 on the West boundary and 0 on the East boundary.

On this test case, F-AMS-pEDFM was run using a coarsening ratio of $4 \times 8 \times 6$ in the matrix and 6×8 in the fractures. The basis functions were computed using the Decoupled-AMS strategy using a $T_{thresh} = 10^2$ transmissibility truncation threshold. A detailed breakdown of the CPU time spent in each stage was recorded at every iteration. Note that the effect of other basis function coupling strategies on the F-AMS-pEDFM convergence behaviour and CPU times [104] is a subject for further research.

For comparison purposes, the pEDFM pressure system was also solved using the industrial-grade multigrid solver, SAMG [45]. More specifically, at each GMRES iteration, a single V-cycle of SAMG was performed, until the residual norm dropped below 10^{-6}. It is important to note that SAMG is a commercial black-box package. Thus, it is not possible to measure its CPU breakdown as accurately as for F-AMS-pEDFM. Instead, the time spent on the first SAMG iteration is considered as "Initialization", while subsequent iterations were counted as "Solution".

Note that for both F-AMS and SAMG, the setup stage was performed only once, before the first iteration. Finally, it is important to emphasize that this study is aimed solely to demonstrate the scalability of the F-AMS-pEDFM method.

The results depicted in Fig. 4.21 show that, by prescribing coarsening ratios, F-AMS-pEDFM is able to scale with an increased number of fractures in the domain. The slightly higher CPU time can be attributed to the computation of extra basis functions, as well as the inversion of a slightly larger coarse-scale pressure system, due to the added number of coarse nodes. At the same time, the additional number of iterations necessary to converge is not severe and can be attributed to the highly contrasting properties of the new fractures to the matrix (up to 8 orders of conductivity contrast).

In conclusion, F-AMS-pEDFM is a promising multiscale method for heterogeneous fractured reservoirs and an appealing candidate for implementation in next-generation flow simulators. In addition, accurate approximations for the pressure field can be obtained after only few iterations of the algorithm, followed by a flux-reconstruction stage, which allows the solution of the transport equations in time-dependent multiphase simulations.

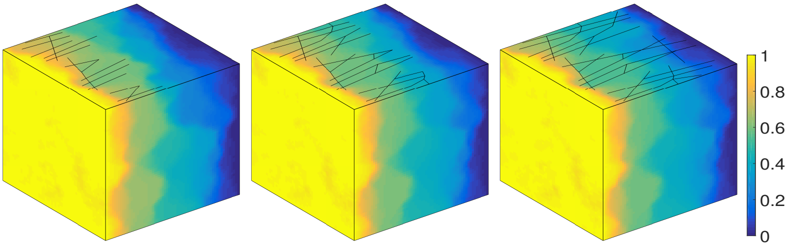


Figure 4.20: Pressure solution obtained for reservoirs with different fracture densities. Each reservoir contains 3 fracture networks which expand in size as follows: stage 1 (left) 36 plates (233×64 cells); stage 2 (middle) 61 plates (412×64 cells); stage 3 (right) 82 plates (551×64 cells). The F-AMS coarsening ratios used were $4 \times 8 \times 6$ in the matrix and 6×8 in the fractures.

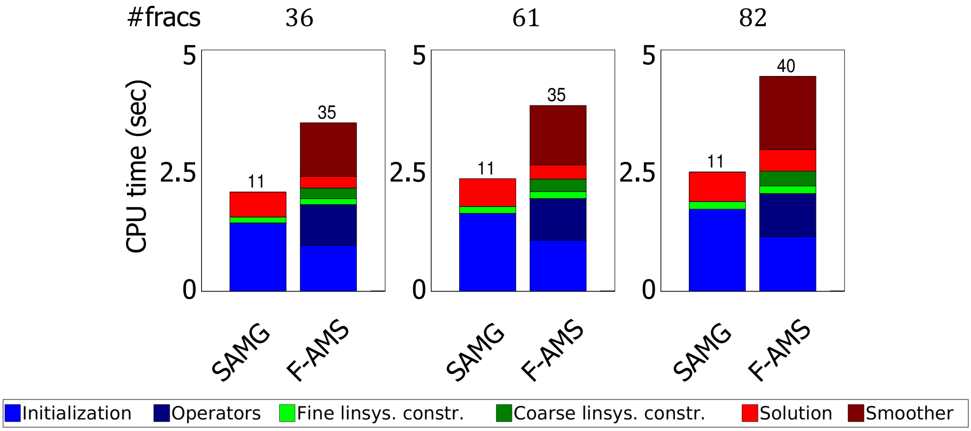


Figure 4.21: F-AMS-pEDFM performance for reservoirs with different fracture densities. The number of performed iterations to reach 10^{-6} residual 2-norm is given on top of each bar.

4.4. Conclusions

A novel Projection-based Embedded Discrete Fracture Model (pEDFM) was devised for flow simulation through fractured porous media. It inherits the grid flexibility of the classic EDFM approach. However, unlike its predecessor, the pEDFM formulation can capture the effect of fracture conductivities ranging from highly permeable networks to inhibiting flow barriers.

The new model was validated on 2D and 3D test cases, while studying its sensitivity towards fracture position within a matrix cell, grid resolution and the cross-media conductivity contrast. The results show that pEDFM is scalable and able to handle dense and complex fracture maps with heterogeneous properties in single-, as well as multiphase flow scenarios. Further, its results on structured grids were found comparable to those obtained using the DFM approach on unstructured, fracture-conforming meshes.

Finally, the possibility of solving the pEDFM pressure system using the F-AMS multiscale framework [104] was investigated. To this end, the computation of basis functions using the Decoupled-AMS strategy was modified to consider transmissibility thresholding. The numerical results show that the resulting F-AMS-pEDFM algorithm attains good convergence rates when only few coarse-scale DOF are prescribed in the fracture networks. The method was found to scale with fracture density in the domain and its performance was compared to that of a commercial-grade solver, SAMG [45].

In conclusion, pEDFM is a flexible model, its simple formulation recommending it for implementation in next-generation simulators for fluid flow through fractured porous media.

The author is happy to observe that the work presented in this chapter has sparked interest in other research groups [133]. Other directions for further improvement include the extension of pEDFM to unstructured grids. Furthermore, systematic benchmarking studies (including CPU time comparisons) between pEDFM and alternative mass-conservative DFM approaches on unstructured grids can provide valuable insights for its application and performance in real-field cases. In terms of the algebraic multiscale formulation, F-AMS-pEDFM, it is important to investigate the effect of different basis function coupling strategies, as well as the method's sensitivity to the geometry of the coarse grid. Finally, the possibility of including complex physics (gravity, capillarity, fracture geomechanics) is the subject of ongoing work.

5

Concluding remarks and future perspectives

In a world with perfectly accurate field measurements and infinite computing power, the work presented in this thesis would merely qualify as an eccentric pastime. But necessity drives innovation and, as long as reality can be modelled using a large-scale system of PDEs with heterogeneous coefficients of uncertain values, there will be a market for efficient approximate solution methods.

Algebraic MultiScale (AMS) methods were designed and have evolved specifically to tackle this challenge. The purpose of the present project is to extend their applicability to a wide range of scenarios involving fluid flow through fractured porous media. This is achieved in three stages:

5.1. C-AMS: rock and fluid compressibility

For a long time after their inception, AMS methods were applied exclusively to test-cases with simplified physics – i.e. incompressible rock and single-phase fluids. In order to relax these assumptions and extend the applicability of AMS to more realistic scenarios, the inclusion of pressure-driven rock and fluid compressibility is studied in this work. These effects introduce time-dependency and non-linearities into the mass balance equations, which require special treatment during the solution process.

A novel Algebraic MultiScale is devised to simulate Compressible flow in heterogeneous porous media (C-AMS, see Chapter 2). It operates by solving the system of equations over a coarse-scale domain, subsequently interpolating the results back to the original resolution. An important point of novelty in this approach are the four different formulations for the basis functions, used in the construction of the prolongation operator (map from coarse- to fine-scale). They capture the spatial heterogeneities in the domain and are computed algebraically, based on localized solutions of the (modified) mass balance equations.

The various components of C-AMS, i.e. the linear and non-linear tolerances, the formulation of the restriction and prolongation operators, the choice and number of iterations of the smoother and the coarsening ratio, make the subject of thorough sensitivity studies, based on CPU times. This leads to the identification of an optimal multiscale strategy for compressible flow scenarios, whose performance is ultimately benchmarked against the SAMG commercial Algebraic MultiGrid solver [45].

The proposed method shows good performance on domains of increasing size and aspect ratios. It excels especially in simulations with many time-steps. This is because C-AMS can save CPU time by updating its operators adaptively, i.e. only when the flow coefficients change dramatically at a given time-step. This feature makes it possible to offset the relatively high initialization time, spent mainly on the computation of the basis functions.

The geometry of the coarse grid is found to be crucial in the performance of C-AMS. Big coarsening factors lead to slow convergence rates, while small coarsening factors imply long setup times. This recommends the investigation of adaptive coarsening, i.e. building the coarse grid using, e.g., graph partitioning algorithms with the fine-scale transmissibility coefficients chosen as weights [134], to ensure a good trade-off between convergence rate and initialization time.

5.2. F-AMS: flow through fractured media

Highly conductive fracture networks, present in the matrix rock, have a significant impact on fluid flow patterns through the subsurface. Yet, their highly localized effect and complicated geometries demand high resolution grids during numerical simulation. This creates a niche for AMS methods, which can recover the fine-scale details of the solution, while solving the system of equations at coarse-scale.

The development of an AMS method suitable for flow through Fractured porous media (F-AMS) is the main focus of this thesis (see Chapter 3). To this end, the prolongation operator, previously defined exclusively for the matrix rock, is now extended by the addition of basis functions for the fractures and any perforating wells present in the domain. Four different strategies to compute the basis functions are investigated (see Appendix B), which differ by the degree of cross-media (i.e. fracture-matrix) coupling they capture. The fully-coupled approach has the potential to deliver the most accurate solutions, however, this comes at the expense of the sparsity pattern in the multiscale operators (restriction and prolongation). The fully decoupled or one-way coupled basis functions maintain this sparsity, while delivering acceptable approximate results.

Just as was done with C-AMS, the various components of the F-AMS algorithm are subjected to comprehensive sensitivity studies, eliciting their convergence and efficiency on a wide range of 3D heterogeneous fractured test cases. The coarsening factor is found, once again, to play an important role in the performance of the method. As expected, a greater number of coarse-scale DOF need to be allocated in the medium (fractures or matrix rock) which dominates the flow. As a rule of thumb using a coarsening factor equal to $\sqrt{N_{\text{cells}}}$ in both media is found to maintain a good trade-off between convergence and setup time. Further studies compare the performance of F-AMS to that of SAMG, while changing the cross-media conductivity contrast, the fracture density and domain scale. The two methods register similar CPU times in all cases.

5.3. pEDFM: fractures with arbitrary conductivities

The Embedded Discrete Fracture Model (EDFM, see [18, 19, 27]) and Discrete Fracture Model (DFM, see [24–26]) are the two main options for flow simulation when fractures need to be explicitly represented. The former can capture fractures with a wide range of length scales and conductivities by confining them at the interfaces between matrix grid cells. This often leads to the construction of complex unstructured grids. EDFM is free of this constraint, since it discretizes each medium (fractures and matrix rock) independently. Its formulation, however, was originally designed for highly conductive fractures and the numerical experiments conducted in this work show that it fails to capture the effects of embedded flow barriers.

This motivates a Projection-based extension of EDFM (pEDFM) to address this drawback, while preserving its advantages. The new model is validated on test cases with embedded fractures of both high and low conductivity, w.r.t to that of the matrix rock. The numerical experiments show that it is equivalent to DFM (and, thus, is most accurate) when fractures are explicitly placed at rock cell interfaces. However, only a small increase in error is observed for fractures located near the grid cell center. The model is scalable w.r.t to grid resolution, cross-media conductivity contrast and fracture density. It is shown to appropriately account for the fracture permeability on a wide range of 2D and 3D test cases, in both single- and multiphase flow scenarios.

In addition, a multiscale formulation for pEDFM is devised. The main challenge lies in the fact that the new model cancels some of the fine-scale matrix-matrix transmissibilities around impermeable fractures. This can lead to rank-deficient systems when solving for the local basis functions, if their formulation disregards matrix-fracture coupling (Decoupled-AMS from [104]). In order to mitigate this, the fine-scale transmissibilities are kept within predefined bounds when constructing these localized systems. The resulting F-AMS-pEDFM method is found to attain good convergence properties and reasonable CPU times, when compared to SAMG. Further improvements are expected with an adaptive coarse grid or by using F-AMS basis functions which consider cross-media coupling.

5.4. Future perspectives

The work presented in this thesis is a significant step forward in unlocking the potential of multiscale methods for real-field fractured reservoir simulation. Their algebraic formulation, first introduced in [61] and expanded here (see Appendix B), has sparked the interest from both the scientific community and the industry, as can be seen from the increasing number of papers published around AMS and its possible extensions.

Important physical effects such as gravity, capillarity and phase transitions (compositional flow) have been systematically neglected in the governing equations used throughout this thesis. They are crucial in real-field cases and should be considered as top priorities for future implementation. At the same time, the set of PDEs should be increased by adding both energy (geothermal, see [75, 76]) and momentum balance (geomechanics, see [77]), ultimately leading to a multiscale multi-physics framework. The main challenge in achieving this lies in the treatment of the coupling between the various types of unknowns (pressure, saturation, composition, displacement, temperature), both at the fine- (temporal and spatial discretization, non-linearities, etc.) and coarse-scale (basis function formulations). In this case, a fully-implicit approach may be suitable, and, as such, suitable multiscale preconditioners should be devised to accelerate convergence (e.g. CPR [80]).

The spatial grids used to represent the reservoirs considered here were structured and uniformly distributed, in order to facilitate the proof-of-concept implementation by avoiding geometric complexities. However, the geological features in real fields may require the use of unstructured grids. At the same time, (p)EDFM can be employed in parts of the domain, to simplify the meshing process. The formulation and performance of AMS methods on such hybrid discretizations are an interesting topic for future research, which can build on top of developments such as [66, 81, 82, 84, 115]. On a similar note, perspectives for applying the multiscale idea to the time-level discretization make for exciting food-for-thought.

In addition to using a suitable coarse-scale grid, further gains in convergence rate can be obtained by applying enrichment strategies to the multiscale prolongation operator [86–88, 102] or tweaking it to maintain basis function monotonicity [90]. These techniques can potentially provide significant efficiency gains in multiphysics scenarios.

The CPU benchmark studies conducted in this thesis are made possible by the in-house C++ object-oriented simulator, DARSim 1, developed during the project. Their findings offer important insight into the efficiency of AMS, when used as a fully converged solver for flow through porous media. However, one of the main advantages of this class of methods is the possibility to reconstruct a conservative flux field, at any stage during the solution process. This enables fast approximate solution results, useful for decision-making, by e.g. choosing a tolerance level in accordance with the uncertainties in the geological parameters. As such, this aspect is important to be considered in future benchmark studies.

Finally, any successful-commercial reservoir simulator must make full use of parallel-computing hardware, such as clusters, the cloud or GPUs. The ongoing work on this topic [62, 135] should be extended to tackle the challenges (i.e. scalability, memory bandwidth, inter-process and inter-thread communication, etc) associated with flow through fractured media.

A

Average distance between fracture and matrix cells

The computation of the average distance between a matrix control volume and a fracture surface, which appears in Eqs. (4.7) and (4.11), may involve numerical integration for arbitrarily shaped cells. For 2D structured grids, however, analytical formulas were given in [27] for a few specific fracture orientations. This subsection, an adaptation of Chapter 2.3.2 from [20], provides a general procedure to handle fracture lines with arbitrary orientation.

The interfaces of each cell intersected by a fracture are extended until they intersect the fracture line, resulting in four right triangles with surfaces A_1 to A_4, as shown in Fig. A.1. Then, given the average distance between each triangle and its hypotenuse, $\langle d \rangle_1$ to $\langle d \rangle_4$, as (see [27]),

$$\langle d \rangle_i = \frac{Lx_i \cdot Ly_i}{3\sqrt{{Lx_i}^2 + {Ly_i}^2}}, \tag{A.1}$$

where Lx_i and Ly_i are the lengths of the axis-aligned sides of triangle i, the average distance between grid cell i and fracture line f is obtained,

$$\langle d \rangle_{if} = \frac{A_1\langle d \rangle_1 + A_2\langle d \rangle_2 - A_3\langle d \rangle_3 - A_4\langle d \rangle_4}{A_1 + A_2 - A_3 - A_4}. \tag{A.2}$$

Note that no modification is required to the formula in the case when fractures lie outside the cell, i.e. for the non-neighbouring connections from Eq. 4.11. In addition, this procedure can be directly applied to 3D structured grids where fractures are extruded along the Z axis, while a generalization for fracture plates with any orientation is the subject of future research.

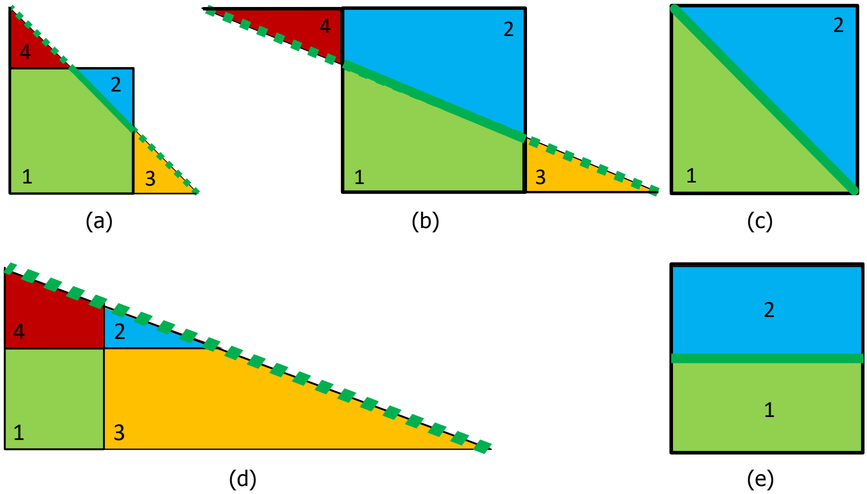


Figure A.1: Analytical calculation of the average distance between a fracture and a matrix cell on 2D structured grids. Four right triangles are constructed by intersecting the cell's edges with the fracture line. Note that triangles 3 and 4 may overlap with triangles 1 or 2 (a,b,d). When the fracture coincides with the cell diagonal, triangles 3 and 4 have zero area (c). If the fracture is aligned with one of the axes, two rectangles are formed instead (e). The same procedure is followed when the fracture lies outside the cell (d).

B

Algebraic computation of the F-AMS prolongation operators

Consider the fine-scale system for matrix and fractures, $\mathbf{A}p = q$, i.e.,

$$\underbrace{\begin{bmatrix} \mathbf{A}^{mm} & \mathbf{A}^{mf} \\ \mathbf{A}^{fm} & \mathbf{A}^{ff} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} p^m \\ p^f \end{bmatrix}}_{p} = \underbrace{\begin{bmatrix} q^m \\ q^f \end{bmatrix}}_{q}. \tag{B.1}$$

The permutation operator $\wp$ containing matrix and fracture permutation block operators $\wp^m$ and $\wp^f$, respectively,

$$\wp \equiv \begin{bmatrix} \wp^m & 0 \\ 0 & \wp^f \end{bmatrix}, \tag{B.2}$$

is defined such that it reorders the linear system (B.1) based on the wirebasket ordering [61, 63, 129] of Internal (I), Face (F), Edge (E) and Vertex (V) for both matrix (superscript ^{m}) and fracture (superscript ^{f}) unknowns, i.e.,

$$\underbrace{\left[\begin{array}{cccc|ccc}
\mathbf{A}^{I^m I^m} & \mathbf{A}^{I^m F^m} & 0 & 0 & \mathbf{A}^{I^m F^f} & \mathbf{A}^{I^m E^f} & \mathbf{A}^{I^m V^f} \\
\mathbf{A}^{F^m I^m} & \mathbf{A}^{F^m F^m} & \mathbf{A}^{F^m E^m} & 0 & \mathbf{A}^{F^m F^f} & \mathbf{A}^{F^m E^f} & \mathbf{A}^{F^m V^f} \\
0 & \mathbf{A}^{E^m F^m} & \mathbf{A}^{E^m E^m} & \mathbf{A}^{E^m V^m} & \mathbf{A}^{E^m F^f} & \mathbf{A}^{E^m E^f} & \mathbf{A}^{E^m V^f} \\
0 & 0 & \mathbf{A}^{V^m E^m} & \mathbf{A}^{V^m V^m} & \mathbf{A}^{V^m F^f} & \mathbf{A}^{V^m E^f} & \mathbf{A}^{V^m V^f} \\
\hline
\mathbf{A}^{F^f I^m} & \mathbf{A}^{F^f F^m} & \mathbf{A}^{F^f E^m} & \mathbf{A}^{F^f V^m} & \mathbf{A}^{F^f F^f} & \mathbf{A}^{F^f E^f} & 0 \\
\mathbf{A}^{E^f I^m} & \mathbf{A}^{E^f F^m} & \mathbf{A}^{E^f E^m} & \mathbf{A}^{E^f V^m} & \mathbf{A}^{E^f F^f} & \mathbf{A}^{E^f E^f} & \mathbf{A}^{E^f V^f} \\
\mathbf{A}^{V^f I^m} & \mathbf{A}^{V^f F^m} & \mathbf{A}^{V^f E^m} & \mathbf{A}^{V^f V^m} & 0 & \mathbf{A}^{V^f E^f} & \mathbf{A}^{V^f V^f}
\end{array}\right]}_{\wp \mathbf{A} \wp^{\mathbf{T}}}
\underbrace{\left[\begin{array}{c}
p^{I^m} \\ p^{F^m} \\ p^{E^m} \\ p^{V^m} \\ \hline p^{F^f} \\ p^{E^f} \\ p^{V^f}
\end{array}\right]}_{\wp p}
=
\underbrace{\left[\begin{array}{c}
q^{I^m} \\ q^{F^m} \\ q^{E^m} \\ q^{V^m} \\ \hline q^{F^f} \\ q^{E^f} \\ q^{V^f}
\end{array}\right]}_{\wp q}. \tag{B.3}$$

Note that fractures have only Face, Edge, and Vertex cells, since they are represented in a lower-dimensional space than the matrix. Also, according to the two-point flux approximation (TPFA) stencil for structured grids, $\mathbf{A}^{I^m E^m}$, $\mathbf{A}^{E^m I^m}$, $\mathbf{A}^{I^m V^m}$, $\mathbf{A}^{V^m I^m}$, $\mathbf{A}^{V^m F^m}$, $\mathbf{A}^{F^m V^m}$, $\mathbf{A}^{F^f V^f}$, $\mathbf{A}^{V^f F^f}$ are zero. More importantly, for media with embedded fractures, the coupling off-diagonal blocks $\mathbf{A}^{mf}$ and $\mathbf{A}^{fm}$ are full, i.e., each matrix cell may overlap with fracture cells of any type (F, E, or V). This is the main reason behind the consideration of the four types of basis functions, each with a different level of matrix-fracture coupling, as previously discussed in this paper.

The algebraic construction of the prolongation operator for each strategy is described next.

B.1. Decoupled-AMS

In the Decoupled-AMS prolongation operator, the matrix-fracture coupling terms are completely neglected. To this end, all off-diagonal block matrix entries (belonging to $\mathbf{A}^{mf}$ and $\mathbf{A}^{fm}$) are set to zero. In addition, similar to the AMS [61] and C-AMS [96] methods, the linear system is further simplified to account for the localization boundary condition within each medium (by neglecting connectivity between each cell and its lower-ranked neighbours in the wirebasket hierarchy). This leads to the following approximate linear system:

$$\underbrace{\left[\begin{array}{cccc|ccc}
\underline{\mathbf{A}}^{I^mI^m} & \mathbf{A}^{I^mF^m} & 0 & 0 & 0 & 0 & 0 \\
0 & \overline{\underline{\mathbf{A}}}^{F^mF^m} & \mathbf{A}^{F^mE^m} & 0 & 0 & 0 & 0 \\
0 & 0 & \overline{\underline{\mathbf{A}}}^{E^mE^m} & \mathbf{A}^{E^mV^m} & 0 & 0 & 0 \\
0 & 0 & 0 & \breve{\mathbf{A}}^{mm} & 0 & 0 & \breve{\mathbf{A}}^{mf} \\
\hline
0 & 0 & 0 & 0 & \underline{\mathbf{A}}^{F^fF^f} & \mathbf{A}^{F^fE^f} & 0 \\
0 & 0 & 0 & 0 & 0 & \overline{\underline{\mathbf{A}}}^{E^fE^f} & \mathbf{A}^{E^fV^f} \\
0 & 0 & 0 & \breve{\mathbf{A}}^{fm} & 0 & 0 & \breve{\mathbf{A}}^{ff}
\end{array}\right]}_{\wp \mathbf{A}' \wp^{\mathbf{T}}}
\underbrace{\left[\begin{array}{c}
p'^{I^m} \\ p'^{F^m} \\ p'^{E^m} \\ p'^{V^m} \\ \hline p'^{F^f} \\ p'^{E^f} \\ p'^{V^f}
\end{array}\right]}_{\wp p'}
=
\underbrace{\left[\begin{array}{c}
0 \\ 0 \\ 0 \\ \breve{q}^{m} \\ \hline 0 \\ 0 \\ \breve{q}^{f}
\end{array}\right]}_{\wp q'}. \tag{B.4}$$

Here, the diagonal blocks marked as $\overline{\mathbf{A}}$ indicate that the matrix-matrix and fracture-fracture transmissibilities, neglected due to this localization assumption ($\mathbf{A}^{F^mI^m}$, $\mathbf{A}^{E^mF^m}$, $\mathbf{A}^{V^mE^m}$ and $\mathbf{A}^{E^fF^f}$, $\mathbf{A}^{V^fE^f}$, respectively) have also been removed from the diagonal term. At the same time, the notation $\underline{\mathbf{A}}$ indicates diagonal blocks where the neglected matrix-fracture transmissibilities have been removed from the diagonal term. Finally,

$$\breve{\mathbf{A}} \equiv \begin{bmatrix} \breve{\mathbf{A}}^{mm} & \breve{\mathbf{A}}^{mf} \\ \breve{\mathbf{A}}^{fm} & \breve{\mathbf{A}}^{ff} \end{bmatrix} = (\mathcal{R}\mathbf{A}\mathcal{P}), \qquad \breve{p} \equiv \begin{bmatrix} p'^{V^m} \\ p'^{V^f} \end{bmatrix}, \qquad \breve{q} \equiv \begin{bmatrix} \breve{q}^{m} \\ \breve{q}^{f} \end{bmatrix} = \mathcal{R}q, \tag{B.5}$$

are the components of the coarse-scale system.

After solving for the coarse-scale pressures, $\breve{p} = \breve{\mathbf{A}}^{-1}\breve{q}$, the approximate system can be inverted algebraically, due to its upper-triangular structure. Consequently, the prolongation operator, which satisfies $p' = \mathcal{P}\breve{p}$, reads

$$\mathcal{P} = \wp^{T} \left[\begin{array}{c|c}
-\underline{\mathbf{A}}^{I^mI^m\,-1}\mathbf{A}^{I^mF^m}\mathcal{P}^{F^mV^m} & 0 \\
-\overline{\underline{\mathbf{A}}}^{F^mF^m\,-1}\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^m} & 0 \\
-\overline{\underline{\mathbf{A}}}^{E^mE^m\,-1}\mathbf{A}^{E^mV^m} & 0 \\
\mathbf{I}^{V^mV^m} & 0 \\
\hline
0 & -\underline{\mathbf{A}}^{F^fF^f\,-1}\mathbf{A}^{F^fE^f}\mathcal{P}^{E^fV^f} \\
0 & -\overline{\underline{\mathbf{A}}}^{E^fE^f\,-1}\mathbf{A}^{E^fV^f} \\
0 & \mathbf{I}^{V^fV^f}
\end{array}\right], \tag{B.6}$$

where $\mathbf{I}$ is the identity matrix and the transpose operator $\wp^{T}$ back-transforms the wirebasket ordering into the natural ordering. Also, $\mathcal{P}^{E^mV^m}$, $\mathcal{P}^{F^mV^m}$ and $\mathcal{P}^{E^fV^f}$ are sub-blocks of the prolongation with the corresponding rows and columns given in the superscripts. For example,

$$\mathcal{P}^{E^mV^m} = -\overline{\underline{\mathbf{A}}}^{E^mE^m\,-1}\mathbf{A}^{E^mV^m},$$
$$\mathcal{P}^{F^mV^m} = -\overline{\underline{\mathbf{A}}}^{F^mF^m\,-1}\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^m}.$$

Note that, once computed, the higher-rank sub-blocks of $\mathcal{P}$ become boundary conditions for the values of basis functions in lower-rank cells, in accordance to the localization assumption (e.g. the values obtained for matrix edges, $\mathcal{P}^{E_mV_m}$, are used to compute the prolongation in adjacent faces, $\mathcal{P}^{F^mV^m}$).

B.2. Frac-AMS

The Frac-AMS approach considers the effect of the $\mathbf{A}^{mf}$ transmissibilities when computing basis functions. This leads to the following approximate system operator

$$\wp\mathbf{A}'\wp^{\mathbf{T}} = \left[\begin{array}{cccc|ccc}
\mathbf{A}^{I^mI^m} & \mathbf{A}^{I^mF^m} & 0 & 0 & \mathbf{A}^{I^mF^f} & \mathbf{A}^{I^mE^f} & \mathbf{A}^{I^mV^f} \\
0 & \overline{\mathbf{A}}^{F^mF^m} & \mathbf{A}^{F^mE^m} & 0 & \mathbf{A}^{F^mF^f} & \mathbf{A}^{F^mE^f} & \mathbf{A}^{F^mV^f} \\
0 & 0 & \overline{\mathbf{A}}^{E^mE^m} & \mathbf{A}^{E^mV^m} & \mathbf{A}^{E^mF^f} & \mathbf{A}^{E^mE^f} & \mathbf{A}^{E^mV^f} \\
0 & 0 & 0 & \breve{\mathbf{A}}^{mm} & 0 & 0 & \breve{\mathbf{A}}^{mf} \\
\hline
0 & 0 & 0 & 0 & \underline{\mathbf{A}}^{F^fF^f} & \mathbf{A}^{F^fE^f} & 0 \\
0 & 0 & 0 & 0 & 0 & \overline{\underline{\mathbf{A}}}^{E^fE^f} & \mathbf{A}^{E^fV^f} \\
0 & 0 & 0 & \breve{\mathbf{A}}^{fm} & 0 & 0 & \breve{\mathbf{A}}^{ff}
\end{array}\right], \quad \text{(B.7)}$$

where $\mathbf{A}^{F^mI^m}$, $\mathbf{A}^{E^mF^m}$, $\mathbf{A}^{V^mE^m}$, $\mathbf{A}^{V^mF^f}$ and $\mathbf{A}^{V^mE^f}$ are set to zero due to localization boundary condition corresponding to Frac-AMS coupling for matrix, while, at the same time, the $\mathbf{A}^{E^fF^f}$, $\mathbf{A}^{V^fE^f}$ are zero in the fracture equations.

The Frac-AMS prolongation operator reads

$$\mathcal{P} = \wp^{T}\left[\begin{array}{c|c}
-\mathbf{A}^{I^mI^m\,-1}\mathbf{A}^{I^mF^m}\mathcal{P}^{F^mV^m} & -\mathbf{A}^{I^mI^m\,-1}(\mathbf{A}^{I^mF^m}\mathcal{P}^{F^mV^f} + \mathbf{A}^{I^mf}\mathcal{P}^{fV^f}) \\
-\overline{\mathbf{A}}^{F^mF^m\,-1}\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^m} & -\overline{\mathbf{A}}^{F^mF^m\,-1}(\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^f} + \mathbf{A}^{F^mf}\mathcal{P}^{fV^f}) \\
-\overline{\mathbf{A}}^{E^mE^m\,-1}\mathbf{A}^{E^mV^m} & -\overline{\mathbf{A}}^{E^mE^m\,-1}\mathbf{A}^{E^mf}\mathcal{P}^{fV^f} \\
\mathbf{I}^{V^mV^m} & 0 \\
\hline
0 & -\underline{\mathbf{A}}^{F^fF^f\,-1}\mathbf{A}^{F^fE^f}\mathcal{P}^{E^fV^f} \\
0 & -\overline{\underline{\mathbf{A}}}^{E^fE^f\,-1}\mathbf{A}^{E^fV^f} \\
0 & \mathbf{I}^{V^fV^f}
\end{array}\right], \quad \text{(B.8)}$$

where the superscript ^{f} (e.g. from $\mathcal{P}^{fV^f}$) corresponds to all the fracture cells, regardless of their containing dual block. Similar as in the previous case, the sub-blocks of prolongation operator $\mathcal{P}^{F^mV^m}$, $\mathcal{P}^{F^mV^f}$, $\mathcal{P}^{fV^f}$, $\mathcal{P}^{E^mV^m}$, $\mathcal{P}^{E^mV^f}$, $\mathcal{P}^{fV^f}$ and $\mathcal{P}^{E^fV^f}$ represent the corresponding rows and columns given in their superscripts. For example, $\mathcal{P}^{E^mV^f} = -\overline{\mathbf{A}}^{E^mE^m\,-1}\mathbf{A}^{E^mf}\mathcal{P}^{fV^f}$ and, specially,

$$\mathcal{P}^{fV^f} = \begin{bmatrix} -\underline{\mathbf{A}}^{F^fF^f\,-1}\mathbf{A}^{F^fE^f}\mathcal{P}^{E^fV^f} \\ -\overline{\underline{\mathbf{A}}}^{E^fE^f\,-1}\mathbf{A}^{E^fV^f} \\ \mathbf{I}^{V^fV^f} \end{bmatrix}. \tag{B.9}$$

B.3. Rock-AMS

For Rock-AMS, the $\mathbf{A}^{mf}$ transmissibilities are set to zero,

$$\wp\mathbf{A}'\wp^{\mathbf{T}} = \left[\begin{array}{cccc|ccc} \underline{\mathbf{A}}^{I^mI^m} & \mathbf{A}^{I^mF^m} & 0 & 0 & 0 & 0 & 0 \\ 0 & \overline{\underline{\mathbf{A}}}^{F^mF^m} & \mathbf{A}^{F^mE^m} & 0 & 0 & 0 & 0 \\ 0 & 0 & \overline{\underline{\mathbf{A}}}^{E^mE^m} & \mathbf{A}^{E^mV^m} & 0 & 0 & 0 \\ 0 & 0 & 0 & \check{\mathbf{A}}^{mm} & 0 & 0 & \check{\mathbf{A}}^{mf} \\ \hline \mathbf{A}^{F^fI^m} & \mathbf{A}^{F^fF^m} & \mathbf{A}^{F^fE^m} & \mathbf{A}^{F^fV^m} & \mathbf{A}^{F^fF^f} & \mathbf{A}^{F^fE^f} & 0 \\ \mathbf{A}^{E^fI^m} & \mathbf{A}^{E^fF^m} & \mathbf{A}^{E^fE^m} & \mathbf{A}^{E^fV^m} & 0 & \overline{\mathbf{A}}^{E^fE^f} & \mathbf{A}^{E^fV^f} \\ 0 & 0 & 0 & \check{\mathbf{A}}^{fm} & 0 & 0 & \check{\mathbf{A}}^{ff} \end{array}\right], \tag{B.10}$$

where, the localization boundary condition was also applied.

The Rock-AMS prolongation operator reads

$$\mathcal{P} = \wp^T \left[\begin{array}{c|c} -\underline{\mathbf{A}}^{I^mI^m\,-1}\mathbf{A}^{I^mF^m}\mathcal{P}^{F^mV^m} & 0 \\ -\overline{\underline{\mathbf{A}}}^{F^mF^m\,-1}\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^m} & 0 \\ -\overline{\underline{\mathbf{A}}}^{E^mE^m\,-1}\mathbf{A}^{E^mV^m} & 0 \\ \mathbf{I}^{V^mV^m} & 0 \\ \hline -\mathbf{A}^{F^fF^f\,-1}(\mathbf{A}^{F^fE^f}\mathcal{P}^{E^fV^m} + \mathbf{A}^{F^fm}\mathcal{P}^{mV^m}) & -\mathbf{A}^{F^fF^f\,-1}\mathbf{A}^{F^fE^f}\mathcal{P}^{E^fV^f} \\ -\overline{\mathbf{A}}^{E^fE^f\,-1}\mathbf{A}^{E^fm}\mathcal{P}^{mV^m} & -\overline{\mathbf{A}}^{E^fE^f\,-1}\mathbf{A}^{E^fV^f} \\ 0 & \mathbf{I}^{V^fV^f} \end{array}\right], \tag{B.11}$$

where the superscript ^{m} (e.g. from $\mathcal{P}^{mV^m}$) corresponds to all the matrix cells, regardless of their containing dual block. The sub-blocks of the prolongation operator, i.e.,$\mathcal{P}^{F^mV^m}$, $\mathcal{P}^{E^mV^m}$, $\mathcal{P}^{E^fV^m}$, $\mathcal{P}^{mV^m}$ and $\mathcal{P}^{E^fV^f}$, are defined similar to the previous cases in the sense that they represent the corresponding rows and columns given in their superscripts. For example, $\mathcal{P}^{E^mV^m} = -\bar{\underline{\mathbf{A}}}^{E^mE^m-1}\mathbf{A}^{E^mV^m}$ and, specially,

$$\mathcal{P}^{mV^m} = \begin{bmatrix} -\underline{\mathbf{A}}^{I^mI^m-1}\mathbf{A}^{I^mF^m}\mathcal{P}^{F^mV^m} \\ -\bar{\underline{\mathbf{A}}}^{F^mF^m-1}\mathbf{A}^{F^mE^m}\mathcal{P}^{E^mV^m} \\ -\bar{\underline{\mathbf{A}}}^{E^mE^m-1}\mathbf{A}^{E^mV^m} \\ \mathbf{I}^{V^mV^m} \end{bmatrix}. \tag{B.12}$$

B.4. Coupled-AMS

In the Coupled-AMS approach, all adjacent Face and Edge blocks are merged between the media (Fig. 3.6), i.e. $F = F^m \cup F^f$ and $E = E^m \cup E^f$. Also, let V denote the set of coarse nodes, irrespective of their location. In this new setting, the algorithm proceeds similar to the original (i.e. unfractured) AMS [61]. More specifically, the $\wp$-reordered approximate linear system is defined as

$$\wp\mathbf{A}'\wp^{\mathbf{T}} = \begin{bmatrix} \mathbf{A}^{II} & \mathbf{A}^{IF} & \mathbf{A}^{IE} & \mathbf{A}^{IV} \\ 0 & \bar{\mathbf{A}}^{FF} & \mathbf{A}^{FE} & \mathbf{A}^{FV} \\ 0 & 0 & \bar{\mathbf{A}}^{EE} & \mathbf{A}^{EV} \\ 0 & 0 & 0 & \check{\mathbf{A}} \end{bmatrix}, \tag{B.13}$$

where the localization boundary condition was appropriately employed.

Then, the Coupled-AMS prolongation operator reads

$$\mathcal{P} = \wp^{T}\begin{bmatrix} -\mathbf{A}^{II^{-1}}(A^{IF}\mathcal{P}^{FV} + \mathbf{A}^{IE}\mathcal{P}^{EV} + \mathbf{A}^{IV}) \\ -\bar{\mathbf{A}}^{FF^{-1}}(\mathbf{A}^{FE}\mathcal{P}^{EV} + \mathbf{A}^{FV}) \\ -\bar{\mathbf{A}}^{EE^{-1}}\mathbf{A}^{EV} \\ \mathbf{I}^{VV} \end{bmatrix}. \tag{B.14}$$

C

Distance-based fracture coarsening

A fracture network can be represented as a graph, in which the fracture lines (plates) are the arcs, while the nodes are the locations at which fractures intersect. This leads to a quasi-unstructured grid, where the complexity mostly revolves around the representation of the intersections. In this work, each intersection is assigned a pressure value, which is explicitly represented in the fine-scale linear system (3.8) via an equation describing the conservation of flux coming from/going into the fracture control volumes it connects.

As previously described, F-AMS requires primal- and dual-coarse grids in the (quasi-unstructured) fracture domain. In order to hide this complexity from the user, this paper introduces a distance-based algorithm for the automatic coarsening of fracture networks, as described in Table C.1. The network's graph is traversed in a breadth-first order such that a distance of at least d_{min} cells is guaranteed between each pair of resulting coarse nodes. As such, d_{min} can be seen as a fracture coarsening factor. Note that choosing $d_{\text{min}} = \infty$ results in a single coarse node, as shown in Fig. 4.17(a). Moreover, Fig. 3.8(b) depicts the result of the coarsening algorithm for $d_{\text{min}} = 20$ cells on a fairly complex fracture network.

Table C.1: Distance-based fracture coarsening algorithm.

```
Repeat the following for each fracture network, $f_i$; let $N_{cells}$ be its fine cell count:
  1. Choose $d_{\mathrm{min}}$, the minimum distance between two fracture coarse nodes.
  2. Initialize three empty queues: $Q_{vertex}, Q_1, Q_2$.
  3. Initialize two vectors of length $N_{cells}$: $level$ (set to $\infty$) and $primal$ (set to $0$).
  4. $N_{primal} \leftarrow 0$
  5. Choose a starting cell from $f_i$ and add it to $Q_{vertex}$.
  Repeat until $Q_{vertex}$ is empty:
      6. $vertex \leftarrow$ extract top of $Q_{vertex}$
      7. $N_{primal} \leftarrow N_{primal} + 1$
      8. Create primal block number $N_{primal}$ with $vertex$ as its coarse node.
      9. $primal[vertex] \leftarrow N_{primal}$ and $level[vertex] \leftarrow 0$.
      10. Add $vertex$ to $Q_1$.
      For $dist$ from $1$ up to and including $d_{\mathrm{min}}$:
          Repeat until $Q_1$ is empty:
              11. $cell \leftarrow$ extract top of $Q_1$
              For each neighbour of $cell$, $neigh_j$, with $level[neigh_j] > dist$:
                  12. remove $neigh_j$ from $Q_{vertex}$, if it is present
                      (i.e. $level[neigh_j] = d_{\mathrm{min}}$)
                  13. $level[neigh_j] \leftarrow dist$
                  14. $primal[neigh_j] \leftarrow N_{primal}$
                  15. add $neigh_j$ to $Q_2$.
          16. swap $Q_1$ and $Q_2$.
      17. empty $Q_1$ into $Q_{vertex}$.

  Upon completion, $primal[i]$ gives the index of fine cell $i$'s primal block, while
  $level[i]$ is the distance from cell $i$ to the nearest vertex. The fine cells which
  were not marked as vertices will form edges on the dual-coarse grid.
```

D

Linear elasticity and poromechanics

The developments from Chapter 2 have introduced fluid and rock compressibility effects into the reservoir simulation process. More specifically, the latter was represented by changing the porosity, as a function of pressure given by the constitutive relationship (2.16). However, in cases when the deformation induced into the rock by fluid flow is mission critical (e.g. due to environmental or safety concerns), this represents an over-simplification of reality. From the physical point of view, the change in rock properties is, instead, dictated by *momentum balance*, i.e. the equilibrium between the *tensile* effect of pore pressure and any external (usually *compressive*) forces applied to the rock body.

The topics of *geomechanics* [136, 137] and *poromechanics* [138] have been widely covered in the literature, especially in dealing with single-phase slightly compressible flow through deformable rocks [139–146]. Since poromechanics lies at the boundary between two separate research topics, i.e. solid mechanics coupled with flow through porous media, there is significant variance in the derivations of the governing equations, often stemming from different choices of parameters or inconsistent notation. Moreover, extensions to multiphase flow [147, 148], fractured media [78, 122] and multiscale methods [77] have only recently come into focus.

For completeness and in order to mitigate further confusion, this chapter provides a detailed derivation and discretization of the PDEs governing multiphase *fully* compressible poromechanics. The coupling between the pressure, saturation and displacement unknowns is resolved in a sequentially-implicit manner, following the *fixed-stress split* [140, 146]. Validation is performed on the Terzaghi problem, by comparing to the analytical solution (see [144]).

D.1. Governing equations

Let $\Omega \subset \mathbb{R}^3$ be a region in space occupied by a deformable porous medium, which is fully saturated by a compressible fluid.

D.1.1. Momentum balance

The momentum balance equation on Ω reads,

$$\nabla \, . \, \boldsymbol{\sigma} = \overline{f} \tag{D.1}$$

where $\boldsymbol{\sigma}$ is the 2nd order symmetric total stress tensor and $\overline{f}$ is the vector of forcing terms. This can be expanded as

$$\begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \cdot \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{D.2}$$

which ultimately translates into the following linear system of equations,

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{yx}}{\partial y} + \frac{\partial \sigma_{zx}}{\partial z} = f_x \tag{D.3}$$

$$\frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{zy}}{\partial z} = f_y \tag{D.4}$$

$$\frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} = f_z \tag{D.5}$$

From the symmetricity of $\boldsymbol{\sigma}$, it follows that $\sigma_{xy} = \sigma_{yx}$, $\sigma_{xz} = \sigma_{zx}$ and $\sigma_{yz} = \sigma_{zy}$.

In order to close the system of equations, the stress constitutive equation (based on Hooke's law) is introduced,

$$\boldsymbol{\sigma} = \mathbb{E} : \boldsymbol{\epsilon} - bp\mathbf{I} \tag{D.6}$$

where $\mathbb{E}$ is the 4th order elasticity tensor, with dimension $3 \times 3 \times 3 \times 3$. Each of its components, $\mathbb{E}_{ijkl}$ relates total stress σ_{ij} to strain ϵ_{kl} in the solid. Also, $b = 1 - \frac{K_m}{K_s}$ is Biot's coefficient, K_m and K_s are the porous medium and solid bulk moduli, respectively, p is the pore pressure, $\mathbf{I}$ is the 2nd order identity tensor,

$$\boldsymbol{\epsilon} = \nabla^s \overline{u} \tag{D.7}$$

is the 2nd order symmetric strain tensor, $\nabla^s = \frac{1}{2}\left(\nabla + \nabla^T\right)$ is the symmetric gradient operator,

$$\overline{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \tag{D.8}$$

is the dispacement vector, i.e.

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} \\ \epsilon_{yx} & \epsilon_{yy} & \epsilon_{yz} \\ \epsilon_{zx} & \epsilon_{zy} & \epsilon_{zz} \end{bmatrix} = \begin{bmatrix} \frac{\partial u_x}{\partial x} & \frac{1}{2}\left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u_y}{\partial x} + \frac{\partial u_x}{\partial y}\right) & \frac{\partial u_y}{\partial y} & \frac{1}{2}\left(\frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y}\right) \\ \frac{1}{2}\left(\frac{\partial u_z}{\partial x} + \frac{\partial u_x}{\partial z}\right) & \frac{1}{2}\left(\frac{\partial u_z}{\partial y} + \frac{\partial u_y}{\partial z}\right) & \frac{\partial u_z}{\partial z} \end{bmatrix} \tag{D.9}$$

Voigt notation

To further expand Equation (D.6), the strain and total stress vectors, along with the accompanying divergence operator, are introduced by exploiting the symmetric structure of $\boldsymbol{\sigma}$ and $\boldsymbol{\epsilon}$,

$$\overline{\sigma} = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{xz} \\ \sigma_{xy} \end{bmatrix}, \qquad \overline{\epsilon} = \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \\ 2\epsilon_{xy} \end{bmatrix}, \qquad \mathbf{div} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{bmatrix}. \tag{D.10}$$

Consequently, Equations (D.1) and (D.7) translate directly into matrix (or Voigt) notation as,

$$\mathbf{div}\ \overline{\sigma} = \overline{f} \tag{D.11}$$

$$\overline{\epsilon} = \mathbf{div}^T \overline{u} \tag{D.12}$$

respectively. More importantly, the 4th order elasticity tensor $\mathbb{E}$ can now be written as a 6×6 elasticity matrix, $\mathbf{E}$, and, assuming an isotropic medium, Equation (D.6) translates into

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{xz} \\ \sigma_{xy} \end{bmatrix} = \underbrace{\begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & & & \\ \lambda & \lambda + 2\mu & \lambda & & & \\ \lambda & \lambda & \lambda + 2\mu & & & \\ & & & \mu & & \\ & & & & \mu & \\ & & & & & \mu \end{bmatrix}}_{\mathbf{E}} \begin{bmatrix} \frac{\partial u_x}{\partial x} \\ \frac{\partial u_y}{\partial y} \\ \frac{\partial u_z}{\partial z} \\ \frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \\ \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \\ \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \end{bmatrix} - \begin{bmatrix} bp \\ bp \\ bp \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{D.13}$$

where λ is the Lamè coefficient and μ is the shear modulus.

Finally, by replacing Equation (D.13), the system of PDEs governing momentum balance for isotropic porous media reads,

$$\mathbf{div}\left(\mathbf{E}\ \mathbf{div}^T \overline{u} - bp\overline{I}\right) = \overline{f} \tag{D.14}$$

where $\bar{I} = [1\ 1\ 1\ 0\ 0\ 0]^T$ is the Voigt notation for the Kroenecker delta. This expands to

$$\frac{\partial}{\partial x}\left[(\lambda+2\mu)\frac{\partial u_x}{\partial x}+\lambda\frac{\partial u_y}{\partial y}+\lambda\frac{\partial u_z}{\partial z}-bp\right]+\frac{\partial}{\partial y}\left(\mu\frac{\partial u_x}{\partial y}+\mu\frac{\partial u_y}{\partial x}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u_x}{\partial z}+\mu\frac{\partial u_z}{\partial x}\right)=f_x \tag{D.15}$$

$$\frac{\partial}{\partial x}\left(\mu\frac{\partial u_x}{\partial y}+\mu\frac{\partial u_y}{\partial x}\right)+\frac{\partial}{\partial y}\left[\lambda\frac{\partial u_x}{\partial x}+(\lambda+2\mu)\frac{\partial u_y}{\partial y}+\lambda\frac{\partial u_z}{\partial z}-bp\right]+\frac{\partial}{\partial z}\left(\mu\frac{\partial u_y}{\partial z}+\mu\frac{\partial u_z}{\partial y}\right)=f_y \tag{D.16}$$

$$\frac{\partial}{\partial x}\left(\mu\frac{\partial u_x}{\partial z}+\mu\frac{\partial u_z}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u_y}{\partial z}+\mu\frac{\partial u_z}{\partial y}\right)+\frac{\partial}{\partial z}\left[\lambda\frac{\partial u_x}{\partial x}+\lambda\frac{\partial u_y}{\partial y}+(\lambda+2\mu)\frac{\partial u_z}{\partial z}-bp\right]=f_z \tag{D.17}$$

D

Finite element discretization

The domain Ω is discretized according to a mesh composed of N_{elems} elements and N_{nodes} nodes. The displacement at any point within element e, $\bar{u}$, is approximated by interpolating its nodal displacement values, $\tilde{u}_{e,i} = \left[u_{x,e,i}\ u_{y,e,i}\ u_{z,e,i}\right]^T$, using predefined local shape functions, $\mathbf{N}_i$,

$$\bar{u} \approx \sum_{i=1}^{N_{\text{nodes},e}} \mathbf{N}_{i,e}\ \tilde{u}_{e,i} \tag{D.18}$$

where,

$$\mathbf{N}_{i,e} = \begin{bmatrix} N_{i,e,x} & & \\ & N_{i,e,y} & \\ & & N_{i,e,z} \end{bmatrix} \tag{D.19}$$

are chosen s.t. $N_{i,e,x} = N_{i,e,y} = N_{i,e,z} = N_{i,e}$, where $N_{i,e}$ is a trilinear function, for simplicity. Further, let

$$\mathbf{N}_e = \left[\mathbf{N}_{1,e} \cdots \mathbf{N}_{N_{\text{nodes},e},e}\right] \tag{D.20}$$

be a $3 \times 3N_{\text{nodes},e}$ matrix obtained by placing the shape function matrices side-by-side. Note that the shape functions form a partition of unity, i.e. $\mathbf{N}_e\bar{1} = 1$, where $\bar{1}$ is a vector of $3N_{\text{nodes},e}$ components equal to 1. Finally, let

$$\tilde{u}_e = \begin{bmatrix} \tilde{u}_{e,1} \\ \vdots \\ \tilde{u}_{e,N_{\text{nodes},e}} \end{bmatrix} \tag{D.21}$$

be a vector of length $3N_{\text{nodes},e}$ obtained by stacking the nodal displacement unknowns.

Then, (D.18) can be written more compactly as

$$\bar{u} \approx \mathbf{N}_e\tilde{u}_e, \tag{D.22}$$

and by substitution into (D.14), premultiplication by $\mathbf{N}_e{}^T$ and integration over the volume of the element, Ω_e, the following *weak form* is obtained

$$\int_{\Omega_e} \mathbf{N}_e{}^T \mathbf{div}\left(\mathbf{E}\ \mathbf{div}^T \mathbf{N}_e \tilde{u}_e - bp_e \bar{I}\right) dV = \int_{\Omega_e} \mathbf{N}_e{}^T \bar{f} dV, \tag{D.23}$$

where p_e is the pore pressure inside the element. By using the notation $\mathbf{D}_e = \mathbf{div}^T \mathbf{N}_e$, which is a $6 \times 3N_{\text{nodes,}e}$ matrix, the previous equation reads,

$$\int_{\Omega_e} \mathbf{D}_e{}^T \mathbf{E} \mathbf{D}_e\ \tilde{u}_e\ dV = \int_{\Omega_e} bp_e\ \mathbf{D}_e{}^T \bar{I}\ dV + \int_{\Omega_e} \mathbf{N}_e{}^T \bar{f} dV. \tag{D.24}$$

Integration

Consider an arbitrary function $f(x)$. Using the Gaussian quadrature rule,

$$\int_{-1}^{1} f(x) dx \approx \sum_{i=1}^{N_{\text{pts}}} w_i\ f(\mathcal{G}_i), \tag{D.25}$$

where $\mathcal{G}_i$ are the Gauss integration points, with w_i their associated weights, s.t. $\sum_{i=1}^{N_{\text{pts}}} w_i = 2$ (i.e. the length of the integration interval). Note that an integral over an arbitrary interval $[a, b]$, has to be first transformed to an equivalent integral over $[-1, 1]$ before the rule can be applied. An N_{pts} rule is known to be exact when integrating polynomial functions up to rank $2N_{\text{pts}} - 1$.

Each of the integrals in (D.24) is evaluated numerically using Gaussian quadrature rules with $N_{\text{pts}} = 2$ over each spatial dimension, which, by definition, operate using the following parameters,

$$\mathcal{G}_1 = \sqrt{\frac{1}{3}}, \qquad \mathcal{G}_2 = -\sqrt{\frac{1}{3}}, \qquad w_1 = w_2 = 1. \tag{D.26}$$

Let $\bar{x}_{i,e}$ denote the vector of Cartesian coordinates of node i in hexahedral element e (i.e. 6 faces, 8 nodes). It can be easily verified that

$$\bar{x} = \sum_{i=1}^{N_{\text{nodes,}e}} \mathbf{N}_{i,e} \bar{x}_{i,e} \qquad \forall \bar{x} \in \Omega_e, \tag{D.27}$$

with

$$N_{i,e}(\xi_1, \xi_2, \xi_3) = \frac{1}{8}(1 \pm \xi_1)(1 \pm \xi_2)(1 \pm \xi_3), \tag{D.28}$$

where $\bar{\xi} = [\xi_1, \xi_2, \xi_3]^T \in [-1, 1]^3$ is the reference coordinate system in element e s.t. the reference coordinates of the nodes are $\bar{\xi}_{i,e} = [\pm 1, \pm 1, \pm 1]^T$. Note that there is correspondence between the sign of component $j \in \{1, 2, 3\}$ of $\bar{\xi}_{i,e}$ and the sign of ξ_j, in the expression of shape function $N_{i,e}$ for node i.

Eqs. (D.27) and (D.28) relate the Cartesian coordinates $\bar{x} = [x, y, z]^T \in \mathbb{R}^3$ of any point within finite element e to the reference coordinates $\bar{\xi} \in [-1, 1]^3$ through $\bar{x} = g(\bar{\xi})$, which implies the following change of variable when integrating,

$$\begin{aligned}\int_{\Omega_e} f(\bar{x}) d\bar{x} &= \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f(g(\bar{\xi}))\, |\det \mathbf{J}|\, d\xi_1 d\xi_2 d\xi_3 \\ &\approx \sum_{i=1}^{N_{\text{pts}}}\sum_{j=1}^{N_{\text{pts}}}\sum_{k=1}^{N_{\text{pts}}} w_i w_j w_k\ f(g(\mathcal{G}_{i,j,k}))\, \left|\det \mathbf{J}_{i,j,k}\right|,\end{aligned} \tag{D.29}$$

where $\mathcal{G}_{i,j,k} = \left[\pm\sqrt{\frac{1}{3}}, \pm\sqrt{\frac{1}{3}}, \pm\sqrt{\frac{1}{3}}\right]^T$ and $w_i = w_j = w_k = 1$, while $\mathbf{J}$ is the 3×3 *jacobian* matrix of the transformation, defined as

$$\mathbf{J} = \frac{\partial(x, y, z)}{\partial(\xi_1, \xi_2, \xi_3)}. \tag{D.30}$$

For a convenient computation of $\mathbf{J}$, the chain rule is employed,

$$\mathbf{J} = \frac{\partial(x, y, z)}{\partial N_{i,e}} \frac{\partial N_{i,e}}{\partial(\xi_1, \xi_2, \xi_3)} \equiv \mathbf{X}_e \mathbf{\Phi}, \tag{D.31}$$

where $\mathbf{X_e}$ is a $3 \times N_{\text{nodes},e}$ matrix containing the Cartesian coordinates of the nodes and $\mathbf{\Phi}$ is a $N_{\text{nodes},e} \times 3$ matrix of basis function derivatives w.r.t. the reference coordinates.

Furthermore, the elements of matrix $\mathbf{D}_e = \mathbf{div}^T \mathbf{N}_e$ can be directly obtained from those of

$$\frac{\partial N_{i,e}}{\partial(x, y, z)} = \frac{\partial N_{i,e}}{\partial(\xi_1, \xi_2, \xi_3)} \frac{\partial(\xi_1, \xi_2, \xi_3)}{\partial(x, y, z)} \equiv \mathbf{\Phi}\ \mathbf{J}^{-1}. \tag{D.32}$$

Algebraic form

After all substitutions are performed, the displacement equation can be written in matrix form for each element,

$$\mathbf{A}_e \tilde{u}_e - p_e \tilde{b}_e = \tilde{f}_e, \tag{D.33}$$

where $\mathbf{A}_e = \int_{\Omega_e} \mathbf{D}_e{}^T \mathbf{E} \mathbf{D}_e\ dV$ is the local stiffness matrix, $\tilde{b}_e = \int_{\Omega_e} b \mathbf{D}_e{}^T \bar{I}\ dV$ is the local pore pressure contribution and $\tilde{f}_e = \int_{\Omega_e} \mathbf{N}_e{}^T \bar{f} dV$ is the local forcing term. These local systems are assembled to form the global linear system,

$$\mathbf{A}_{\text{mech}}\ \tilde{u} + \mathbf{B}_{\text{mech}}\ \bar{p} = \tilde{f} \tag{D.34}$$

where $\mathbf{A}_{\text{mech}}$ is the $3N_{\text{nodes}} \times 3N_{\text{nodes}}$ global stiffness matrix, $\tilde{u}$ is the $3N_{\text{nodes}}$ global vector of nodal displacements, $\mathbf{B}_{\text{mech}}$ is the $3N_{\text{nodes}} \times N_{\text{elems}}$ global mechanics-flow coupling matrix, $\bar{p}$ the N_{elems} vector of element pore pressures, while $\tilde{f}$ is the $3N_{\text{nodes}}$ vector of forcing terms.

D.1.2. Mass balance

Considering a multiphase fluid flowing through the isotropic porous medium Ω, the mass balance equations read

$$\frac{\partial m_\alpha}{\partial t} + \nabla \,.\, w_\alpha = \rho_\alpha q_\alpha, \qquad \alpha = 1, \ldots, N_{\text{phases}} \tag{D.35}$$

where m_α denotes mass, w_α is the mass flux, ρ_α is density and q_α is the volumetric source term for phase α.

By definition,

$$m_\alpha = \phi \rho_\alpha s_\alpha, \tag{D.36}$$

where ϕ is the porosity and s_α is the phase saturation, while, from Darcy's law,

$$w_\alpha = -\rho_\alpha \mathbf{k} \lambda_\alpha \cdot \nabla p, \tag{D.37}$$

where $\mathbf{k}$ is the 2nd order positive-definite permeability tensor, here, considered diagonal and isotropic, $\lambda_\alpha = \frac{k_{r,\alpha}}{\mu_\alpha}$, $k_{r,\alpha}$ and μ_α are the phase mobility, relative permeability (not to be confused with the Lame parameter λ) and viscosity (not to be confused with the shear modulus μ), respectively.

By replacing (D.36) and (D.37) into (D.35) and expanding the time derivative according to the chain rule,

$$\frac{\partial \phi}{\partial t} \rho_\alpha s_\alpha + \phi \frac{\partial \rho_\alpha}{\partial t} s_\alpha + \phi \rho_\alpha \frac{\partial s_\alpha}{\partial t} - \nabla \,.\, (\mathbf{k} \lambda_\alpha \cdot \nabla p) = \rho_\alpha q_\alpha, \tag{D.38}$$

which can be further transformed by division with ρ_α followed by summation over all phases, giving

$$\frac{\partial \phi}{\partial t} + \phi \sum_{\alpha=1}^{N_{\text{phases}}} \left(\frac{s_\alpha}{\rho_\alpha} \frac{\partial \rho_\alpha}{\partial t} \right) - \sum_{\alpha=1}^{N_{\text{phases}}} \frac{1}{\rho_\alpha} \nabla \,.\, (\mathbf{k} \lambda_\alpha \cdot \nabla p) = q_t, \tag{D.39}$$

where $q_t = \sum_{\alpha=1}^{N_{\text{phases}}} q_\alpha$ denotes the total volumetric source term.

An exponential model is assumed for the phase density,

$$\rho_\alpha = \rho_{\alpha,0} \exp\left(c_\alpha (p - p_0)\right), \tag{D.40}$$

where $\rho_{\alpha,0}$ and p_0 are the reference phase density and pore pressure, while $c_\alpha = \frac{1}{K_\alpha}$ is the phase compressibility, with K_α, the phase bulk modulus. Furthermore, from Biot's linear elasticity theory,

$$\phi = \phi_0 + b\left(\epsilon_v - \epsilon_{v,0}\right) + (b - \phi_0)\, c_s\, (p - p_0), \tag{D.41}$$

where ϕ_0 is the reference porosity, $\epsilon_v = tr(\epsilon) \equiv \epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz} \equiv \nabla \,.\, \overline{u}$ is the volumetric strain and $c_s = \frac{1}{K_s}$ is the compressibility of the solid grains, with K_s, their bulk modulus.

Consequently, the time derivatives present in (D.39) become,

$$\frac{\partial \rho_\alpha}{\partial t} = \frac{\partial \rho_\alpha}{\partial p}\frac{\partial p}{\partial t} = c_\alpha \rho_\alpha \frac{\partial p}{\partial t} \tag{D.42}$$

$$\frac{\partial \phi}{\partial t} = \frac{\partial \phi}{\partial \epsilon_v}\frac{\partial \epsilon_v}{\partial t} + \frac{\partial \phi}{\partial p}\frac{\partial p}{\partial t} = b\frac{\partial \epsilon_v}{\partial t} + (b-\phi_0)c_s\frac{\partial p}{\partial t} \tag{D.43}$$

leading to the following PDE for pressure

$$b\frac{\partial \epsilon_v}{\partial t} + \frac{1}{M}\frac{\partial p}{\partial t} - \sum_{\alpha=1}^{N_\text{phases}} \frac{1}{\rho_\alpha}\nabla\,.\,(\rho_\alpha \mathbf{k}\lambda_\alpha \cdot \nabla p) = q_t, \tag{D.44}$$

where $\frac{1}{M} = \phi c_f + (b-\phi_0)c_s$ is the inverse Biot's modulus and $c_f = \sum_{\alpha=1}^{N_\text{phases}} s_\alpha c_\alpha$ is defined as the overall fluid compressibility. Note that (D.44) is an extension of the pressure equation from [140] to fully compressible multiphase flow.

Finite volume discretization

Eq. (D.44) is solved for the pore pressure using the finite-volume method, to ensure mass conservation. To this end, the volume integral over each element is applied,

$$\int_{\Omega_e}\left[b\frac{\partial \epsilon_v}{\partial t} + \frac{1}{M}\frac{\partial p}{\partial t}\right] dV - \sum_{\alpha=1}^{N_\text{phases}} \int_{\Omega_e} \frac{1}{\rho_\alpha}\nabla\,.\,(\rho_\alpha \mathbf{k}\lambda_\alpha \cdot \nabla p)\ dV = \int_{\Omega_e} q_t\ dV. \tag{D.45}$$

Assuming changes in $\frac{1}{\rho_\alpha}$ are negligible over the element, the Gauss theorem can be applied,

$$\int_{\Omega_e}\left[b\frac{\partial \epsilon_v}{\partial t} + \frac{1}{M}\frac{\partial p}{\partial t}\right] dV - \sum_{\alpha=1}^{N_\text{phases}} \frac{1}{\rho_\alpha}\int_{\partial\Omega_e} (\rho_\alpha \mathbf{k}\lambda_\alpha \cdot \nabla p)\cdot \overline{n}\ dS = \int_{\Omega_e} q_t\ dV, \tag{D.46}$$

where $\partial\Omega_e$ is the surface of element, with $\overline{n}$ its normal.

Integration

In order to perform the integration, the properties are assumed constant over the volume of the element, ΔV_e, which has $N_{\text{interf},e}$ interfaces. Then,

$$\Delta V_e\left(b\frac{\partial \epsilon_v}{\partial t} + \frac{1}{M}\frac{\partial p}{\partial t}\right) - \sum_{\alpha=1}^{N_\text{phases}} \frac{1}{\rho_\alpha} \sum_{i=1}^{N_{\text{interfs},e}} \Delta A_i \left(\rho_{\alpha,i}\mathbf{k}_i\lambda_{\alpha,i}\cdot\nabla p\right)\cdot \overline{n}_i = Q_t, \tag{D.47}$$

where ΔA_i is the interfacial area, while Q_t denotes the total fluid volume exchanged with source terms. Note that the two sums in the advection terms can be interchanged, as convenient.

The permeability at the interface, $\mathbf{k}_i$, is computed as the harmonic average between the values of the neighbouring cells, while the phase density, $\rho_{\alpha,i}$, and mobility, $\lambda_{\alpha,i}$, are evaluated based on upwinded pressure and saturation values.

Algebraic form

The porosity and density dependence on pressure makes (D.44) a nonlinear PDE. Thus, after backward Euler time integration, the Newton-Raphson approach is employed, while lagging $\frac{1}{M}$ in the accumulation and $\frac{1}{\rho_\alpha}$ and $\rho_{\alpha,i}$ in the advection terms by one iteration ν. This finally leads to the following pressure system,

$$\mathbf{A}_{\text{flow}}^{(\nu)}\, \overline{p}^{(\nu+1)} + \mathbf{B}_{\text{flow}}\, \widetilde{u} = \overline{c}_{\text{flow}}^{(\nu)} + \overline{q}_t, \tag{D.48}$$

where $\mathbf{A}_{\text{flow}}$ is the $N_{\text{elems}} \times N_{\text{elems}}$ transmissibility matrix, $\mathbf{B}_{\text{flow}}$ is the $N_{\text{elems}} \times 3N_{\text{nodes}}$ flow-mechanics coupling matrix, $\overline{c}_{\text{flow}}$ is the N_{elems} right-hand-side vector, while $\overline{q}_t$ is the N_{elems} vector of total volumetric source terms.

D.1.3. Phase transport

The transport equation is obtained from (D.35) as,

$$\frac{\partial \phi \rho_\alpha s_\alpha}{\partial t} + \nabla\, .\, (f_\alpha w_t) = f_\alpha o_t, \tag{D.49}$$

where $w_t = \sum_{\beta=1}^{N_{\text{phases}}} \rho_\beta \mathbf{k} \lambda_\beta {\cdot} \nabla p$ denotes the total advection mass flux, $o_t = \sum_{\beta=1}^{N_{\text{phases}}} \rho_\beta q_\beta$ is the total source mass flux , while the mass fractional flow function, f_α, is defined as

$$f_\alpha = \frac{\rho_\alpha \lambda_\alpha}{\sum_{\beta=1}^{N_{\text{phases}}} \rho_\beta \lambda_\beta}, \tag{D.50}$$

for each phase, such that $w_\alpha = f_\alpha w_t$.

Finite volume discretization

By applying the integral over the volume of each element on either side of (D.49) and using the Gauss theorem,

$$\frac{\partial}{\partial t} \int_{\Omega_e} \phi \rho_\alpha s_\alpha \; dV + \int_{\partial \Omega_e} (f_\alpha w_t) \cdot \overline{n} \; dS = \int_{\Omega_e} f_\alpha o_t \; dV \tag{D.51}$$

Integration

Assuming that the quantities remain constant over the volume of the element, the following saturation equation is obtained,

$$\Delta V_e \frac{\partial \phi \rho_\alpha s_\alpha}{\partial t} + \sum_{i=1}^{N_{\text{interfs},e}} \Delta A_i \left(f_{\alpha,i} w_t \right) \cdot \overline{n}_i = f_\alpha O_t, \tag{D.52}$$

where o_t denotes the total fluid mass exchanged with source terms.

The interfacial mass fractional flow is evaluated based on upwinded values for pressure and saturation.

Algebraic form

The nonlinear transport PDE (D.49) is solved by backward Euler time integration in a Newton-Raphson loop, after linearization using a first-order Taylor expansion of f_α,

$$f_\alpha^{(\nu+1)} \approx f_\alpha^{(\nu)} + \frac{\partial f_\alpha}{\partial s_\alpha}^{(\nu)} \left(s_\alpha^{(\nu+1)} - s_\alpha^{(\nu)} \right). \tag{D.53}$$

leading to the following saturation system,

$$\mathbf{A}_{\text{trans},\alpha}^{(\nu)} \ \overline{s}_\alpha^{(\nu+1)} = \overline{c}_{\text{trans},\alpha}^{(\nu)} + \overline{o}_\alpha, \tag{D.54}$$

where $\mathbf{A}_{\text{trans},\alpha}$ is the $N_{\text{elems}} \times N_{\text{elems}}$ transmissibility matrix, $\overline{c}_{\text{trans},\alpha}$ is the N_{elems} right-hand-side vector, while $\overline{o}_\alpha$ is the N_{elems} vector of total mass source terms.

Transport is solved for the first $N_{\text{phases}} - 1$ phases, while the saturation of the last phase is obtained as

$$s_{N_{\text{phases}}} = 1 - \sum_{\beta=1}^{N_{\text{phases}}-1} s_\beta. \tag{D.55}$$

D.1.4. Coupling strategy

The multiphase poromechanics problem described above is solved using a three-stage sequentially implicit strategy (Fig. D.1). The first stage is the flow solver, where the pore pressure, $\overline{p}$ is obtained, followed by mechanics, which gives the displacement, $\tilde{u}$, and, finally, transport which solves for the phase saturations, $\overline{s}_\alpha, \forall \alpha = 1, \ldots, N_{\text{phases}}$.

The following paragraphs detail the operator splitting which enables this sequential procedure, while ensuring its convergence.

Flow and mechanics stages

The coupling between the (linear) mechanics and (non-linear) flow equations consists of the appearance of pore pressure, p, in Eq. (D.14) and the volumetric strain, $\epsilon_v = \nabla \,.\, \tilde{u}$, in Eq. (D.44). In order to resolve these strong two-way dependencies, the joint pressure-displacement system is written as,

$$\begin{bmatrix} \mathbf{A}_{\text{mech}} & \mathbf{B}_{\text{mech}} \\ \mathbf{B}_{\text{flow}} & \mathbf{A}_{\text{flow}}^{(\nu)} \end{bmatrix} \begin{bmatrix} \tilde{u} \\ \overline{p}^{(\nu+1)} \end{bmatrix} = \begin{bmatrix} \tilde{f} \\ \overline{g}^{(\nu)} \end{bmatrix}, \tag{D.56}$$

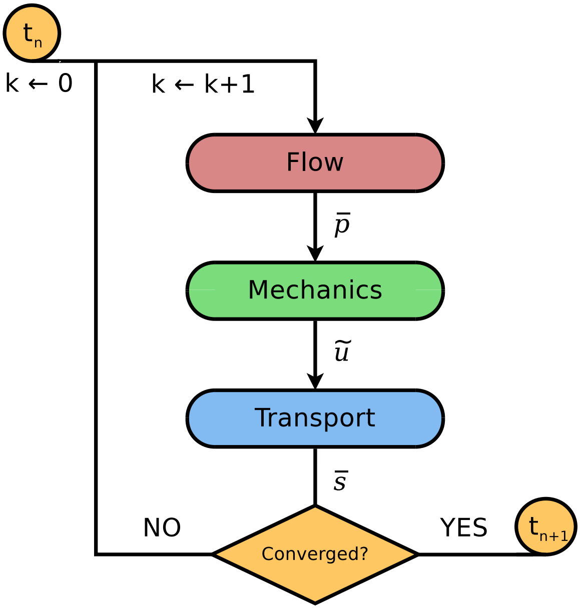


Figure D.1: Three-stage sequentially implicit solution strategy for the multiphase poromechanics problem.

where ν denotes the index in the Newton-Raphson (linearization) iterative loop. By comparing Eqs. (D.24) and (D.46), it follows that

$$\mathbf{B}_{\text{flow}} = \mathbf{B}^T_{\text{mech}}. \tag{D.57}$$

$\mathbf{A}^{(\nu)}_{\text{flow}}$ can be further split into its accumulation and advection (i.e. transmissibility) parts,

$$\mathbf{A}^{(\nu)}_{\text{flow}} = \mathbf{C}^{(\nu)} + \mathbf{T}^{(\nu)} \tag{D.58}$$

and the detailed expression for the right-hand-side corresponding to the flow problem (in accordance with Eq. (D.48)) is

$$\overline{g}^{(\nu)} = \mathbf{B}_{\text{flow}}\widetilde{u}^{(n)} + \underbrace{\mathbf{C}^{(\nu)}\overline{p}^{(n)}}_{\overline{c}^{(\nu)}_{\text{flow}}} + \overline{q}_t, \tag{D.59}$$

where n denotes the previous timestep (Fig D.1).

The system (D.58) is traditionally solved in monolithic (i.e. fully-implicit) fashion [139, 143, 149]. More recently, however, sequentially-implicit methods for poromechanics have emerged [140–142], among which, the *fixed-stress split* [140] has seen widespread attention [142–144, 146, 148]. These methods are attractive because they allow the use of distinct solution strategies for each type of unknown, as appropriate, given the nature of their respective PDEs (elliptic momentum versus parabolic mass balance).

According to [146], the decoupling required for the formulation of these methods can be achieved in a compact way by preconditioning the jacobian operator from (D.58). The fixed-stress split, in particular, relies on the LDU-decomposition,

$$\begin{bmatrix} \mathbf{A}_{\text{mech}} & \mathbf{B}_{\text{mech}} \\ \mathbf{B}_{\text{flow}} & \mathbf{A}^{(\nu)}_{\text{flow}} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I} & \\ \mathbf{B}_{\text{flow}}\,\mathbf{A}^{-1}_{\text{mech}} & \mathbf{I} \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \mathbf{A}_{\text{mech}} & \\ & \mathbf{S}^{(\nu)} \end{bmatrix}}_{\mathbf{D}} \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{A}^{-1}_{\text{mech}}\,\mathbf{B}_{\text{mech}} \\ & \mathbf{I} \end{bmatrix}}_{\mathbf{U}}, \tag{D.60}$$

where $\mathbf{S}^{(\nu)}$ is the Schur-complement of $\mathbf{A}^{(\nu)}_{\text{flow}}$,

$$\mathbf{S}^{(\nu)} = \mathbf{A}^{(\nu)}_{\text{flow}} - \mathbf{B}_{\text{flow}}\,\mathbf{A}^{-1}_{\text{mech}}\,\mathbf{B}_{\text{mech}}. \tag{D.61}$$

By choosing the upper-triangular matrix $(\mathbf{DU})^{-1}$ as left-preconditioner, the following Richardson's iterative scheme [143] is obtained,

$$\begin{bmatrix} \mathbf{A}_{\text{mech}} & \mathbf{B}_{\text{mech}} \\ & \mathbf{S}'^{(\nu)} \end{bmatrix} \begin{bmatrix} \widetilde{u}^{(n+1)}_{(k+1)} \\ \overline{p}^{(\nu+1)}_{(k+1)} \end{bmatrix} = \begin{bmatrix} & \\ -\mathbf{B}_{\text{flow}} & \mathbf{M} \end{bmatrix} \begin{bmatrix} \widetilde{u}^{(n+1)}_{(k)} \\ \overline{p}^{(n+1)}_{(k)} \end{bmatrix} + \begin{bmatrix} \widetilde{f} \\ \overline{g}^{(\nu)} \end{bmatrix}, \tag{D.62}$$

where k gives the (sequential) iteration step. Note that a sparse approximation of $\mathbf{S}^{(\nu)}$ was used [146],

$$\mathbf{S}'^{(\nu)} = \mathbf{A}^{(\nu)}_{\text{flow}} + \mathbf{M} \tag{D.63}$$

$$\mathbf{M} = \frac{b^2}{K_m}\mathbf{R}, \tag{D.64}$$

where $\mathbf{R}$ denotes the element-wise volume integrals of the pressure shape functions, which are constant 1, in accordance with the finite-volume discretization of the mass balance equations. This makes $\mathbf{R}$ a diagonal matrix, having $\Delta V_e, \forall e = 1, \ldots, N_{\text{elems}}$ as its entries.

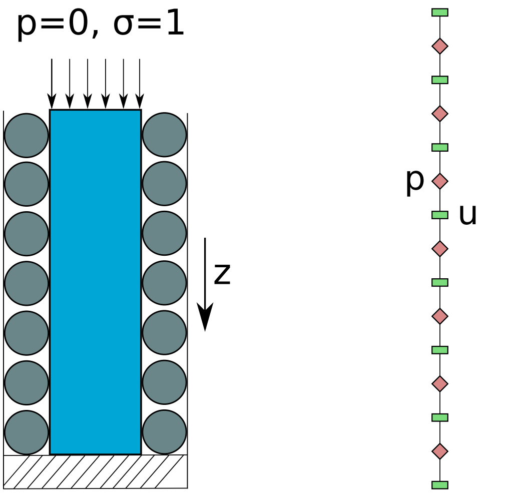
The dependence on the solution from the previous sequential iteration k in (D.62) acts as a regularization term, which ensures unconditional stability [140].

Transport stage

Given the pressure- and strain-dependent porosity update (D.41), the (nonlinear) transport equations are decoupled from pressure using the fractional flow formulation (see the dedicated paragraph from Section D.1.2), in a similar fashion to a black-oil-type simulator.

D.2. Validation

The validation of the sequentially-implicit poromechanics scheme presented above, is performed on the widely-used Terzaghi problem, depicted in Fig. D.2(a). In this



(a) Terzaghi's problem

(b) Simulation grid

Figure D.2: Setup and simulation grid for Terzaghi's problem. The saturated porous rock column is subjected to one-dimensional compaction, while fluid is allowed to flow through the top. In the discretization, the displacement unknowns are defined at cell corners, while pore pressure is solved at the cell centers.

setup, a saturated column of porous rock is subjected to one-dimensional compaction ($\sigma = 1$ Pa on the top, rollers on the sides and fixed to the bottom boundary), while fluid is allowed to flow through the top ($p = 0$ Pa on the top, no flow across the rest of the boundaries). The simulation parameters are given in Table D.1.

Parameter	**Value**
Young's modulus	10^4 Pa
Poisson's ratio	0.2
Biot's coefficient	1.0
Inverse Biot's modulus	0.0
Fluid mobility	10^{-4} m^2/cP
Initial pressure	1.0 Pa
Initial displacement	0.0 m

Table D.1: Simulation parameters for the Terzaghi problem.

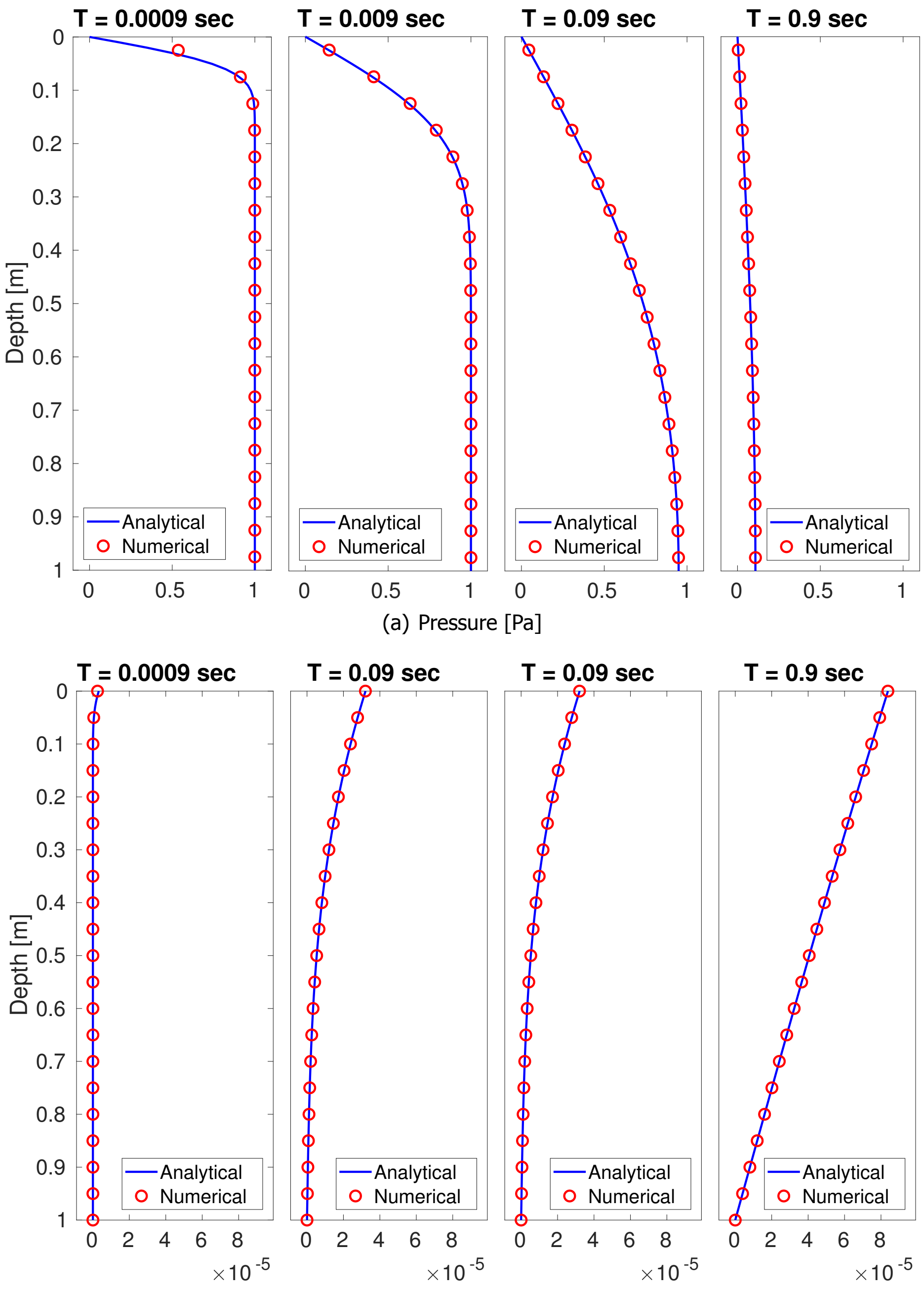
The first two lines give the pair of elasticity moduli that determine the elasticity of the rock. The conversion to any other pair among the 6 possible moduli (bulk Modulus K_m, Young's modulus E, Lame's first parameter λ, shear modulus μ, Poisson's ratio ν or P-wave modulus M) is easily done [137], e.g.,

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \tag{D.65}$$

$$\mu = \frac{E}{2(1+\nu)}. \tag{D.66}$$

The numerical solution is computed on the 1D discretized domain shown in Fig. D.2(b), where, as explained before, the pressure unknowns are located at cell centers, while the displacement is computed at the cell corners.

Simulation results, over several time-steps, are shown in Fig. D.3 alongside the analytical solution [144]. The two are in good agreement, which confirms that the sequentially-implicit scheme is able to capture the effect of poromechanics.



(a) Pressure [Pa]

(b) Displacement [m]

Figure D.3: Time-lapse pore pressure (top) and displacement solutions (bottom), computed numerically and analytically, for the Terzaghi problem.

References

[1] BP Global, *Statistical review of world energy,* (2017).

[2] A. H. Muggeridge, A. Cockin, K. Webb, H. Frampton, I. Collins, T. Moulds and P. Salino, *Recovery rates, enhanced oil recovery and technological limits,* Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **372**, 20120320 (2013).

[3] L. W. Lake, R. T. Johns, W. R. Rossen and G. A. Pope, *Fundamentals of Enhanced Oil Recovery* (Society of Petroleum Engineers, 2014).

[4] Q. Wang, X. Chen, A. N. Jha and H. Rogers, *Natural gas from shale formation – the evolution, evidences and challenges of shale gas revolution in United States,* Renewable and Sustainable Energy Reviews **30**, 1 (2014).

[5] D. Y. Leung, G. Caramanna and M. M. Maroto-Valer, *An overview of current status of carbon dioxide capture and storage technologies,* Renewable and Sustainable Energy Reviews **39**, 426 (2014).

[6] R. M. Cuéllar-Franca and A. Azapagic, *Carbon capture, storage and utilisation technologies: A critical analysis and comparison of their life cycle environmental impacts,* Journal of CO2 Utilization **9**, 82 (2015).

[7] S. A. Rackley, *Carbon Capture and Storage, Second Edition* (Butterworth-Heinemann, 2017).

[8] W. T. Pfeiffer and S. Bauer, *Subsurface porous media hydrogen storage – scenario development and simulation,* Energy Procedia **76**, 565 (2015).

[9] M. Panfilov, *Underground and pipeline hydrogen storage,* in *Compendium of Hydrogen Energy* (Elsevier, 2016) pp. 91–115.

[10] W. T. Pfeiffer, C. Beyer and S. Bauer, *Hydrogen storage in a heterogeneous sandstone formation: dimensioning and induced hydraulic effects,* Petroleum Geoscience **23**, 315 (2017).

[11] R. Bertani, *Geothermal power generation in the world 2005–2010 update report,* Geothermics **41**, 1 (2012).

[12] J. W. Lund and T. L. Boyd, *Direct utilization of geothermal energy 2015 worldwide review,* Geothermics **60**, 66 (2016).

[13] R. Bertani, *Geothermal power generation in the world 2010–2014 update report,* Geothermics **60**, 31 (2016).

[14] M. P. Anderson, W. W. Woessner and R. J. Hunt, *Applied Groundwater Modeling, Second Edition: Simulation of Flow and Advective Transport* (Academic Press, 2015).

[15] C. Zheng and G. D. Bennett, *Applied Contaminant Transport Modeling* (Wiley-Interscience, 2002).

[16] G. Dagan, *Flow and Transport in Porous Formations* (Springer, 1989).

[17] R. Al-Raoush and A. Papadopoulos, *Representative elementary volume analysis of porous media using x-ray computed tomography,* Powder Technology **200**, 69 (2010).

[18] S. H. Lee, C. L. Jensen and M. F. Lough, *Efficient finite-difference model for flow in a reservoir with multiple length-scale fractures,* SPE Journal **5**, 268 (2000).

[19] L. Li and S. H. Lee, *Efficient field-scale simulation of black oil in a naturally fractured reservoir through discrete fracture networks and homogenized media,* SPE Reservoir Evaluation & Engineering **11**, 750 (2008).

[20] S. B. Pluimers, *Hierarchical fracture modeling*, Master's thesis, Delft University of Technology, the Netherlands (2015).

[21] G. Barenblatt, I. Zheltov and I. Kochina, *Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks [strata],* Journal of Applied Mathematics and Mechanics **24**, 1286 (1960).

[22] J. Warren and P. Root, *The behavior of naturally fractured reservoirs,* SPE Journal **3**, 245 (1963).

[23] T. D. van Golf-Racht, *Fundamentals of Fractured Reservoir Engineering, Volume 12 (Developments in Petroleum Science)* (Elsevier Science, 1982).

[24] M. Karimi-Fard, L. Durlofsky and K. Aziz, *An efficient discrete-fracture model applicable for general-purpose reservoir simulators,* SPE Journal **9**, 227 (2004).

[25] V. Reichenberger, H. Jakobs, P. Bastian and R. Helmig, *A mixed-dimensional finite volume method for two-phase flow in fractured porous media,* Advances in Water Resources **29**, 1020 (2006).

[26] R. Ahmed, M. G. Edwards, S. Lamine, B. A. Huisman and M. Pal, *Three-dimensional control-volume distributed multi-point flux approximation coupled with a lower-dimensional surface fracture model,* Journal of Computational Physics **303**, 470 (2015).

[27] H. Hajibeygi, D. Karvounis and P. Jenny, *A hierarchical fracture model for the iterative multiscale finite volume method,* Journal of Computational Physics **230**, 8729 (2011).

[28] D. H. Johnston, *Practical Applications of Time-Lapse Seismic Data* (Society of Exploration Geophysicists, 2013).

[29] D. S. Oliver, A. C. Reynolds and N. Liu, *Inverse Theory for Petroleum Reservoir Characterization and History Matching* (Cambridge University Press, 2008).

[30] M. Christie, *Upscaling for reservoir simulation,* Journal of Petroleum Technology **48**, 1004 (1996).

[31] Y. Chen and L. J. Durlofsky, *Adaptive local–global upscaling for general flow scenarios in heterogeneous formations,* Transport in Porous Media **62**, 157 (2006).

[32] N. Schwenck, B. Flemisch, R. Helmig and B. I. Wohlmuth, *Dimensionally reduced flow models in fractured porous media: crossings and boundaries,* Computational Geosciences **19**, 1219 (2015).

[33] J. Warren and H. Price, *Flow in heterogeneous porous media,* Society of Petroleum Engineers Journal **1**, 153 (1961).

[34] S. Begg and P. King, *Modelling the effects of shales on reservoir performance: Calculation of effective vertical permeability,* in *SPE Reservoir Simulation Symposium* (Society of Petroleum Engineers, 1985).

[35] A. J. Desbarats, *Numerical estimation of effective permeability in sand-shale formations,* Water Resources Research **23**, 273 (1987).

[36] E. Kasap and L. W. Lake, *Calculating the effective permeability tensor of a gridblock,* SPE Formation Evaluation **5**, 192 (1990).

[37] L. J. Durlofsky, *Numerical calculation of equivalent grid block permeability tensors for heterogeneous porous media,* Water Resources Research **27**, 699 (1991).

[38] Y. Chen, L. Durlofsky, M. Gerritsen and X. Wen, *A coupled local–global upscaling approach for simulating flow in highly heterogeneous formations,* Advances in Water Resources **26**, 1041 (2003).

[39] M. Sahimi, R. Darvishi, M. Haghighi and M. R. Rasaei, *Upscaled unstructured computational grids for efficient simulation of flow in fractured porous media,* Transport in Porous Media **83**, 195 (2009).

[40] A. Fumagalli, S. Zonca and L. Formaggia, *Advances in computation of local problems for a flow-based upscaling in fractured reservoirs,* Mathematics and Computers in Simulation **137**, 299 (2017).

[41] M. A. Christie and M. J. Blunt, *Tenth SPE comparative solution project: A comparison of upscaling techniques,* in *SPE Reservoir Simulation Symposium* (Society of Petroleum Engineers, 2001).

[42] P. Jenny, S. H. Lee and H. A. Tchelepi, *Multi-scale finite-volume method for elliptic problems in subsurface flow simulation,* Journal of Computational Physics **187**, 47 (2003).

[43] H. Hajibeygi and P. Jenny, *Multiscale finite-volume method for parabolic problems arising from compressible multiphase flow in porous media,* Journal of Computational Physics **228**, 5129 (2009).

[44] U. Trottenberg, C. W. Oosterlee and A. Schuller, *Multigrid* (Academic Press, 2000).

[45] K. Stüben, *SAMG User Manual* (Fraunhofer Institute SCAI, 2010).

[46] T. Y. Hou and X.-H. Wu, *A multiscale finite element method for elliptic problems in composite materials and porous media,* Journal of Computational Physics **134**, 169 (1997).

[47] T. Y. Hou, X.-H. Wu and Z. Cai, *Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients,* Mathematics of Computation **68**, 913 (1999).

[48] Y. Efendiev and T. Y. Hou, *Multiscale Finite Element Methods: Theory and Applications* (Springer, 2009).

[49] Y. R. Efendiev, T. Y. Hou and X.-H. Wu, *Convergence of a nonconforming multiscale finite element method,* SIAM Journal on Numerical Analysis **37**, 888 (2000).

[50] J. E. Aarnes and T. Y. Hou, *Multiscale domain decomposition methods for elliptic problems with high aspect ratios,* Acta Mathematicae Applicatae Sinica, English Series **18**, 63 (2002).

[51] J. E. Aarnes, V. Kippe and K.-A. Lie, *Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels,* Advances in Water Resources **28**, 257 (2005).

[52] H. Zhou and H. A. Tchelepi, *Operator-based multiscale method for compressible flow,* SPE Journal **13**, 267 (2008).

[53] H. Zhou and H. A. Tchelepi, *Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models,* SPE Journal **17**, 523 (2012).

[54] P. Jenny, S. H. Lee and H. A. Tchelepi, *Adaptive multiscale finite-volume method for multiphase flow and transport in porous media,* Multiscale Modeling & Simulation **3**, 50 (2005).

[55] P. Jenny, S. H. Lee and H. A. Tchelepi, *Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media,* Journal of Computational Physics **217**, 627 (2006).

[56] H. Hajibeygi, G. Bonfigli, M. A. Hesse and P. Jenny, *Iterative multiscale finite-volume method,* Journal of Computational Physics **227**, 8604 (2008).

[57] Y. Saad, *Iterative Methods for Sparse Linear Systems* (Society for Industrial and Applied Mathematics, 2003).

[58] I. Lunati, M. Tyagi and S. H. Lee, *An iterative multiscale finite volume algorithm converging to the exact solution,* Journal of Computational Physics **230**, 1849 (2011).

[59] H. Hajibeygi and P. Jenny, *Adaptive iterative multiscale finite volume method,* Journal of Computational Physics **230**, 628 (2011).

[60] A. Kozlova, Z. Li, J. R. Natvig, S. Watanabe, Y. Zhou, K. Bratvedt and S. H. Lee, *A real-field multiscale black-oil reservoir simulator,* SPE Journal **21**, 2049 (2016).

[61] Y. Wang, H. Hajibeygi and H. A. Tchelepi, *Algebraic multiscale solver for flow in heterogeneous porous media,* Journal of Computational Physics **259**, 284 (2014).

[62] A. M. Manea, J. Sewall and H. A. Tchelepi, *Parallel multiscale linear solver for highly detailed reservoir models,* SPE Journal **21**, 2062 (2016).

[63] J. R. Wallis and H. A. Tchelepi, *Apparatus, method and system for improved reservoir simulation using an algebraic cascading class linear solver,* Patent US7684967 (2010).

[64] T. Arbogast, *Numerical subgrid upscaling of two-phase flow in porous media,* in *Lecture Notes in Physics* (Springer Berlin Heidelberg, 2000) pp. 35–49.

[65] T. Arbogast, *Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase darcy flow,* Computational Geosciences **6**, 453 (2002).

[66] O. Møyner and K.-A. Lie, *A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids,* Journal of Computational Physics **304**, 46 (2016).

[67] I. Lunati and P. Jenny, *Multiscale finite-volume method for density-driven flow in porous media,* Computational Geosciences **12**, 337 (2008).

[68] P. Jenny and I. Lunati, *Modeling complex wells with the multi-scale finite-volume method,* Journal of Computational Physics **228**, 687 (2009).

[69] C. Wolfsteiner, S. H. Lee and H. A. Tchelepi, *Well modeling in the multiscale finite volume method for subsurface flow simulation,* Multiscale Modeling & Simulation **5**, 900 (2006).

[70] S. H. Lee, C. Wolfsteiner and H. A. Tchelepi, *Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity,* Computational Geosciences **12**, 351 (2008).

[71] H. Hajibeygi and H. A. Tchelepi, *Compositional multiscale finite-volume formulation,* SPE Journal **19**, 316 (2014).

[72] M. Cusini, B. Fryer, C. van Kruijsdijk and H. Hajibeygi, *Algebraic dynamic multilevel method for compositional flow in heterogeneous porous media,* Journal of Computational Physics **354**, 593 (2018).

[73] H. Zhou, S. H. Lee and H. A. Tchelepi, *Multiscale finite-volume formulation for saturation equations,* SPE Journal **17**, 198 (2012).

[74] S. Lee, H. Zhou and H. A. Tchelepi, *Adaptive multiscale finite-volume method for nonlinear multiphase transport in heterogeneous formations,* Journal of Computational Physics **228**, 9036 (2009).

[75] T. H. Sandve, *Multiscale simulation of flow and heat transport in fractured geothermal reservoirs*, Ph.D. thesis, University of Bergen, Norway (2012).

[76] T. Praditia, *Multiscale Finite Volume Method for Coupled Single-Phase Flow and Heat Equations in Fractured Porous Media: Application to Geothermal Systems*, Master's thesis, Delft University of Technology, the Netherlands (2017).

[77] N. Castelletto, H. Hajibeygi and H. A. Tchelepi, *Multiscale finite-element method for linear elastic geomechanics,* Journal of Computational Physics **331**, 337 (2017).

[78] R. Deb and P. Jenny, *Modeling of shear failure in fractured reservoirs with a porous matrix,* Computational Geosciences **21**, 1119 (2017).

[79] R. Deb and P. Jenny, *Finite volume-based modeling of flow-induced shear failure along fracture manifolds,* International Journal for Numerical and Analytical Methods in Geomechanics **41**, 1922 (2017).

[80] M. Cusini, A. A. Lukyanov, J. Natvig and H. Hajibeygi, *Constrained pressure residual multiscale (CPR-MS) method for fully implicit simulation of multiphase flow in porous media,* Journal of Computational Physics **299**, 472 (2015).

[81] O. Møyner and K.-A. Lie, *A multiscale two-point flux-approximation method,* Journal of Computational Physics **275**, 273 (2014).

[82] O. Møyner and K.-A. Lie, *The multiscale finite-volume method on stratigraphic grids,* SPE Journal **19**, 816 (2014).

[83] S. B. M. Bosma, H. Hajibeygi, M. Ţene and H. A. Tchelepi, *Multiscale finite volume method for discrete fracture modeling with unstructured grids,* in *SPE Reservoir Simulation Conference* (Society of Petroleum Engineers, 2017).

[84] S. Bosma, H. Hajibeygi, M. Ţene and H. A. Tchelepi, *Multiscale finite volume method for discrete fracture modeling on unstructured grids (MS-DFM),* Journal of Computational Physics **351**, 145 (2017).

[85] M. Cusini, C. van Kruijsdijk and H. Hajibeygi, *Algebraic dynamic multilevel (ADM) method for fully implicit simulations of multiphase flow in porous media,* Journal of Computational Physics **314**, 60 (2016).

[86] Y. Efendiev, J. Galvis and X.-H. Wu, *Multiscale finite element methods for high-contrast problems using local spectral basis functions,* Journal of Computational Physics **230**, 937 (2011).

[87] E. T. Chung, Y. Efendiev, G. Li and M. Vasilyeva, *Generalized multiscale finite element methods for problems in perforated heterogeneous domains,* Applicable Analysis **95**, 2254 (2015).

[88] D. Cortinovis and P. Jenny, *Iterative galerkin-enriched multiscale finite-volume method,* Journal of Computational Physics **277**, 248 (2014).

[89] M. A. Hesse, B. T. Mallison and H. A. Tchelepi, *Compact multiscale finite volume method for heterogeneous anisotropic elliptic equations,* Multiscale Modeling & Simulation **7**, 934 (2008).

[90] Y. Wang, H. Hajibeygi and H. A. Tchelepi, *Monotone multiscale finite volume method,* Computational Geosciences **20**, 509 (2015).

[91] R. J. de Moraes, J. R. P. Rodrigues, H. Hajibeygi and J. D. Jansen, *Multiscale gradient computation for flow in heterogeneous porous media,* Journal of Computational Physics **336**, 644 (2017).

[92] J.-D. Jansen, D. Brouwer, G. Naevdal and C. P. J. W. van Kruijsdijk, *Closed-loop reservoir management,* First Break **23**, 43 (2005).

[93] I. Lunati and P. Jenny, *Multiscale finite-volume method for compressible multiphase flow in porous media,* Journal of Computational Physics **216**, 616 (2006).

[94] J. R. Natvig, B. Skaflestad, F. Bratvedt, K. Bratvedt, K.-A. Lie, V. Laptev and S. Khataniar, *Multiscale mimetic solvers for efficient streamline simulation of fractured reservoirs,* SPE Journal **16**, 880 (2011).

[95] M. Ţene, H. Hajibeygi, Y. Wang and H. Tchelepi, *Adaptive algebraic multiscale solver for compressible flow in heterogeneous porous media,* in *ECMOR XIV - 14th European Conference on the Mathematics of Oil Recovery* (EAGE Publications BV, 2014).

[96] M. Ţene, Y. Wang and H. Hajibeygi, *Adaptive algebraic multiscale solver for compressible flow in heterogeneous porous media,* Journal of Computational Physics **300**, 679 (2015).

[97] K. Aziz and A. Settari, *Petroleum Reservoir Simulation* (Chapman & Hall, 1979).

[98] V. Kippe, J. E. Aarnes and K.-A. Lie, *A comparison of multiscale methods for elliptic problems in porous media flow,* Computational Geosciences **12**, 377 (2008).

[99] H. Hajibeygi, S. H. Lee and I. Lunati, *Accurate and efficient simulation of multiphase flow in a heterogeneous reservoir with error estimate and control in the multiscale finite-volume framework,* SPE Journal **17**, 1071 (2012).

[100] N. Remy, A. Boucher and J. Wu, *Applied Geostatistics with SGeMS* (Cambridge University Press, 2009).

[101] G. Bonfigli and P. Jenny, *An efficient multi-scale poisson solver for the incompressible navier–stokes equations with immersed boundaries,* Journal of Computational Physics **228**, 4568 (2009).

[102] V. Dolean, P. Jolivet, F. Nataf, N. Spillane and H. Xiang, *Two-level domain decomposition methods for highly heterogeneous darcy equations. connections with multiscale methods,* Oil & Gas Science and Technology – Revue d'IFP Energies nouvelles **69**, 731 (2014).

[103] M. Ţene, M. S. A. Kobaisi and H. Hajibeygi, *Algebraic multiscale solver for flow in heterogeneous fractured porous media,* in *SPE Reservoir Simulation Symposium* (Society of Petroleum Engineers, 2015).

[104] M. Ţene, M. S. A. Kobaisi and H. Hajibeygi, *Algebraic multiscale method for flow in heterogeneous porous media with embedded discrete fractures (F-AMS),* Journal of Computational Physics **321**, 819 (2016).

[105] B. Berkowitz, *Characterizing flow and transport in fractured geological media: A review,* Advances in Water Resources **25**, 861 (2002).

[106] A. Moinfar, W. Narr, M.-H. Hui, B. T. Mallison and S. H. Lee, *Comparison of discrete-fracture and dual-permeability models for multiphase flow in naturally fractured reservoirs,* in *SPE Reservoir Simulation Symposium* (Society of Petroleum Engineers, 2011).

[107] S. K. Matthäi, A. A. Mezentsev and M. Belayneh, *Finite element node-centered finite-volume two-phase-flow experiments with fractured rock represented by unstructured hybrid-element meshes,* SPE Reservoir Evaluation & Engineering **10**, 740 (2007).

[108] A. Moinfar, A. Varavei, K. Sepehrnoori and R. T. Johns, *Development of an efficient embedded discrete fracture model for 3d compositional reservoir simulation in fractured reservoirs,* SPE Journal **19**, 289 (2014).

[109] R. Ahmed, M. G. Edwards, S. Lamine, B. A. H. Huisman and M. Pal, *Control-volume distributed multi-point flux approximation coupled with a lower-dimensional fracture model,* Journal of Computational Physics **284**, 462 (2015).

[110] S. Geiger-Boschung, S. K. Matthäi, J. Niessner and R. Helmig, *Black-oil simulations for three-component, three-phase flow in fractured porous media,* SPE Journal **14**, 338 (2009).

[111] T. H. Sandve, E. Keilegavlen and J. M. Nordbotten, *Physics-based preconditioners for flow in fractured porous media,* Water Resources Research **50**, 1357 (2014).

[112] D. W. Peaceman, *Interpretation of well-block pressures in numerical reservoir simulation(includes associated paper 6988),* SPE Journal **18**, 183 (1978).

[113] D. C. Karvounis and P. Jenny, *Adaptive hierarchical fracture model for enhanced geothermal systems,* Multiscale Modeling & Simulation **14**, 207 (2016).

[114] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini and H. Zhang, *PETSc Users Manual*, Tech. Rep. ANL-95/11 - Revision 3.6 (Argonne National Laboratory, 2015).

[115] S. Y. Shah, O. Møyner, M. Ţene, K.-A. Lie and H. Hajibeygi, *The multiscale restriction smoothed basis method for fractured porous media (f-MsRSB),* Journal of Computational Physics **318**, 36 (2016).

[116] M. Ţene, M. S. al Kobaisi and H. Hajibeygi, *Multiscale projection-based embedded discrete fracture modeling approach (F-AMS-pEDFM),* in *ECMOR XV - 15th European Conference on the Mathematics of Oil Recovery* (EAGE Publications BV, 2016).

[117] M. Ţene, S. B. Bosma, M. S. A. Kobaisi and H. Hajibeygi, *Projection-based embedded discrete fracture model (pEDFM),* Advances in Water Resources **105**, 205 (2017).

[118] A. B. Tatomir, A. Szymkiewicz, H. Class and R. Helmig, *Modeling two phase flow in large scale fractured porous media with an extended multiple interacting continua method,* Computer Modeling in Engineering & Sciences **77**, 81 (2011).

[119] M. Karimi-Fard and L. J. Durlofsky, *A general gridding, discretization, and coarsening methodology for modeling flow in porous formations with discrete geological features,* Advances in Water Resources **96**, 354 (2016).

[120] J. B. Moortgat and A. Firoozabadi, *Three-phase compositional modeling with capillarity in heterogeneous and fractured media,* SPE Journal **18**, 1150 (2013).

[121] J. Moortgat, M. A. Amooie and M. R. Soltanian, *Implicit finite volume and discontinuous galerkin methods for multicomponent flow in unstructured 3d fractured porous media,* Advances in Water Resources **96**, 389 (2016).

[122] T. T. Garipov, M. Karimi-Fard and H. A. Tchelepi, *Discrete fracture model for coupled flow and geomechanics,* Computational Geosciences **20**, 149 (2016).

[123] B. Flemisch, A. Fumagalli and A. Scotti, *A review of the XFEM-based approximation of flow in fractured porous media,* in *SEMA SIMAI Springer Series* (Springer International Publishing, 2016) pp. 47–76.

[124] R. Deb and P. Jenny, *Numerical modeling of flow-mechanics coupling in a fractured reservoir with porous matrix,* in *Proceedings of the 41st Workshop on Geothermal Reservoir Engineering* (2016) pp. 1–9.

[125] B. Flemisch, I. Berre, W. Boon, A. Fumagalli, N. Schwenck, A. Scotti, I. Stefansson and A. Tatomir, *Benchmarks for single-phase flow in fractured porous media,* Advances in Water Resources **111**, 239 (2018).

[126] J. H. Norbeck, M. W. McClure, J. W. Lo and R. N. Horne, *An embedded fracture modeling framework for simulation of hydraulic fracturing and shear stimulation,* Computational Geosciences **20**, 1 (2015).

[127] A. Fumagalli, L. Pasquale, S. Zonca and S. Micheletti, *An upscaling procedure for fractured reservoirs with embedded grids,* Water Resources Research **52**, 6506 (2016).

[128] Z. Chen, *Reservoir simulation: mathematical techniques in oil recovery* (Society for Industrial and Applied Mathematics, 2007).

[129] J. M. Nordbotten and P. E. Bjørstad, *On the relationship between the multiscale finite-volume method and domain decomposition preconditioners,* Computational Geosciences **12**, 367 (2008).

[130] H. Hamzehpour, M. Asgari and M. Sahimi, *Acoustic wave propagation in heterogeneous two-dimensional fractured porous media,* Physical Review E **93**, 063305 (2016).

[131] G. Bertotti and K. Bisdom, *Fracture patterns in the Jandeira Fm. (NE Brazil),* (2017).

[132] K. Bisdom, G. Bertotti and H. M. Nick, *The impact of different aperture distribution models and critical stress criteria on equivalent permeability in fractured rocks,* Journal of Geophysical Research: Solid Earth **121**, 4045 (2016).

[133] J. Jiang and R. M. Younis, *An improved projection-based embedded discrete fracture model (pEDFM) for multiphase flow in fractured reservoirs,* Advances in Water Resources **109**, 267 (2017).

[134] G. Karypis and V. Kumar, *A fast and high quality multilevel scheme for partitioning irregular graphs,* SIAM Journal on Scientific Computing **20**, 359 (1998).

[135] M. Khait and D. V. Voskov, *GPU-offloaded general purpose simulator for multiphase flow in porous media,* in *SPE Reservoir Simulation Conference* (Society of Petroleum Engineers, 2017).

[136] A. Verruijt, *Computational Geomechanics (Theory and Applications of Transport in Porous Media)* (Springer, 1999).

[137] H. F. Wang, *Theory of Linear Poroelasticity with Applications to Geomechanics and Hydrogeology* (Princeton University Press, 2000).

[138] O. Coussy, *Poromechanics* (Wiley, 2004).

[139] B. Jha and R. Juanes, *A locally conservative finite element framework for the simulation of coupled flow and reservoir geomechanics,* Acta Geotechnica **2**, 139 (2007).

[140] J. Kim, H. Tchelepi and R. Juanes, *Stability and convergence of sequential methods for coupled flow and geomechanics: Fixed-stress and fixed-strain splits,* Computer Methods in Applied Mechanics and Engineering **200**, 1591 (2011).

[141] J. Kim, H. Tchelepi and R. Juanes, *Stability and convergence of sequential methods for coupled flow and geomechanics: Drained and undrained splits,* Computer Methods in Applied Mechanics and Engineering **200**, 2094 (2011).

[142] J. Kim, H. A. Tchelepi and R. Juanes, *Stability, accuracy, and efficiency of sequential methods for coupled flow and geomechanics,* SPE Journal **16**, 249 (2011).

[143] J. A. White and R. I. Borja, *Block-preconditioned newton–krylov solvers for fully coupled flow and geomechanics,* Computational Geosciences **15**, 647 (2011).

[144] N. Castelletto, J. A. White and H. A. Tchelepi, *Accuracy and convergence properties of the fixed-stress iterative solution of two-way coupled poromechanics,* International Journal for Numerical and Analytical Methods in Geomechanics **39**, 1593 (2015).

[145] N. Castelletto, J. A. White and M. Ferronato, *Scalable algorithms for three-field mixed finite element coupled poromechanics,* Journal of Computational Physics **327**, 894 (2016).

[146] J. A. White, N. Castelletto and H. A. Tchelepi, *Block-partitioned solvers for coupled poromechanics: A unified framework,* Computer Methods in Applied Mechanics and Engineering **303**, 55 (2016).

[147] B. Jha and R. Juanes, *Coupled multiphase flow and poromechanics: A computational model of pore pressure effects on fault slip and earthquake triggering,* Water Resources Research **50**, 3776 (2014).

[148] T. Almani, K. Kumar, A. Dogru, G. Singh and M. Wheeler, *Convergence analysis of multirate fixed-stress split iterative schemes for coupling flow with geomechanics,* Computer Methods in Applied Mechanics and Engineering **311**, 180 (2016).

[149] J. H. Prevost, *Two-way coupling in reservoir-geomechanical models: vertex-centered galerkin geomechanical model cell-centered and vertex-centered finite volume reservoir models,* International Journal for Numerical Methods in Engineering **98**, 612 (2014).

Curriculum Vitæ

Matei Ţene

16 July 1988 Born in Bucharest, Romania.

Education

2003 – 2007 “Tudor Vianu” National High School of Computer Sciences,
Bucharest, Romania

2007 – 2011 BSc. Computer Science and Engineering
Artificial Intelligence track
“Politehnica” University of Bucharest, Romania

Thesis:
Autonomous robot for the Eurobot competition
Promotor: Prof. dr. ing. R. Vârbănescu

2011 – 2013 MSc. Applied Mathematics
Risk and Environmental Modelling track
Delft University of Technology, the Netherlands

Thesis:
Ensemble-based History Matching for Channelized Petroleum Reservoirs
Promotor: Prof. dr. A.H. Heemink

2014 – 2018 PhD Petroleum Engineering
Delft University of Technology, the Netherlands

Professional experience

March – May 2013	Intern, TNO, Utrecht, the Netherlands Ensemble-based history matching methods
June – August 2017	Intern, Chevron ETC, Houston, TX, USA Sequentially-implicit compositional flow simulation
From August 2018	Software Engineer, Schlumberger, Abingdon, UK INTERSECT Program

Volunteering

2016 – 2017	Founding board member Delft Student Dance Association “*Blue Suede Shoes*”

List of Publications

No pressure, no diamonds.

Thomas Carlyle

The following is a listing of the journal publications and conference papers authored, along with the talks given during the PhD program, in reverse chronological order.

Journal papers

S.B.M. Bosma, H. Hajibeygi, **M. Ţene**, H.A. Tchelepi, *Multiscale finite volume method for discrete fracture modeling on unstructured grids (MS-DFM)*, Journal of Computational Physics **351**, 145 (2017).

M. Ţene, S.B.M. Bosma, M.S. Al Kobaisi, H. Hajibeygi, *Projection-based embedded discrete fracture model (pEDFM)*, Advances in Water Resources **105**, 205 (2017).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *Algebraic multiscale method for flow in heterogeneous porous media with embedded discrete fractures (F-AMS)*, Journal of Computational Physics **321**, 819 (2016).

S.Y. Shah, O. Møyner, **M. Ţene**, K.-A. Lie, H. Hajibeygi, *The multiscale restriction smoothed basis method for fractured porous media (F-MsRSB)*, Journal of Computational Physics **318**, 36 (2016).

M. Ţene, Y. Wang, H. Hajibeygi, *Adaptive algebraic multiscale solver for compressible flow in heterogeneous porous media*, Journal of Computational Physics **300**, 679 (2015).

Conference proceedings and talks

S.B.M. Bosma, H. Hajibeygi, **M. Ţene**, H.A. Tchelepi, *Multiscale finite volume method for discrete fracture modeling with unstructured grids*, in SPE Reservoir Simulation Conference, (Society of Petroleum Engineers, 2017).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *Projection-based Embedded Discrete Fracture Model*, 9th International Conference on Porous Media and Annual Meeting, (2017).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *Multiscale projection-based embedded discrete fracture modeling approach (F-AMS-pEDFM)*, in ECMOR XV - 15th European Conference on the Mathematics of Oil Recovery, (EAGE Publications BV, 2016).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *F-AMS: a flexible multiscale framework for multiphase flow through naturally fractured porous media*, 11th International Conference on Computational Methods in Water Resources, (2016).

M. Ţene, H. Hajibeygi, *Imposing Convergence Constraints on Multiscale Mehods for the Efficient Solution of Multiphase Flow Through Naturally Fractured Porous Media*, SIAM Conference on Mathematical and Computational Issues in the Geosciences, (2015).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *Algebraic multiscale solver for flow in heterogeneous fractured porous media*, in SPE Reservoir Simulation Symposium, (Society of Petroleum Engineers, 2015).

M. Ţene, M.S. Al Kobaisi, H. Hajibeygi, *Adaptive Algebraic Multiscale Solver for Compressible Flow in Heterogeneous Porous Media*, in ECMOR XIV - 14th European Conference on the Mathematics of Oil Recovery, (EAGE Publications BV, 2014).

Acknowledgements

Truly great friends are hard to find, difficult to leave
and impossible to forget

Here is my attempt to say thank you to all who have made the past four years truly rich and memorable.

First and foremost, I would like to thank my daily supervisor, Dr. **Hadi Hajibeygi**. Dear, Hadi, what a journey has this been! You were always the engine that pushed the project forward, bringing out the best in me – even if, sometimes, caught inbetween deadlines, I failed to see it. Yes, we have faced our differences at times, but, through it all, there was not a single moment when I felt I could not rely on your unconditional support. I will never forget how we both stayed up all night to finish my first SPE RSS paper! Looking from afar, one may say that stretching deadlines has been the norm of this PhD; then again, we have always put quality above deadlines. I am proud to have been your first PhD student and I hope that this is not the end of our working relationship – of course, this time, "depending on *my* budget" ... :) I am truly happy to hear the news that you are soon going to become a father and wish you happiness and success in your new role!

A big thank you to my promotor, Prof. **Jan-Dirk Jansen**, for recommending me for the PhD position. I truly valued your impartial advice and guidance throughout, especially in the more difficult moments. The EnKF workshop in Bergen remains one of my fondest memories – an opportunity I would have missed, were it not for your encouragement and support. Congratulations and the best of luck in your new position as dean of the CiTG faculty!

Denis, even though you are one of the profs, to me it always felt like you are part of our PhD office! I must admit that, when we first met, I was a bit taken aback by the tough question you asked me during my SIAM GS talk. Yes, it was about treating relperm endpoints with multscale :). But all that went away quickly and I am very happy to have you as a friend. Starting August, I will be just across the channel, so that shouldn't stop us from catching up in 't Klooster from time to time.

Of course, I am very grateful for the help received from our department support staff. **Marlijn**, **Lydia**, **Margot**, **Marja**, **Marijke**, thank you for always greeting me with a smile. I have never walked out of your office without answers to my questions. And thanks to **Ralf** for helping me move between the different desks around the building over the 4 years.

My first desk in CiTG was in one of the flex offices, where I met **Nico**. I still remember watching the Winter Olympics with you and our discussions about Romania. Good luck in your career!

Then, it was time to mingle with the geophysicists – thank you, **Max**, **Koen**, **Kevin**, **Remi**, **Siddarth**, **Iris**, **Helena**, **Carlos**, **Anna**, **Andreas**, **Boris**, **Pawan**, **Amarjeet**, **Tomo**, **Niels**, **Ranjani** for welcoming me into your 3rd floor office. I remember the Tuesday football matches, our evening drinks and the many times I had to duck under my desk whenever you would shoot those foam bullet guns. You were my first big family in the faculty!

Eventually, the opportunity presented itself to work among the petroleum engineers, on the ground floor– at first, sitting next to the lab technicians. Thanks, **Henk**, **Jolanda**, **Ellen**, **Arno**, **Michiel**, for the good company, in the back of the room. **Marc**, you always lightened the mood when you spontaneously started to sing pop songs – veel succes met airport dancing, jonge! **Karel**, once again thanks for helping me tilt my monitor to portrait mode and opening the cassette on my old bike's wheel. **Karl-Heinz**, we always had a good laugh when I would meet you around the coffee machine or by the *art* piece in the exhibition room. I wish you all good health and patience with the students!

I will never forget **Chris "The Outlaw" Boeije** – you were always so laid back, down-to-earth and easy to get along with. I remember you would actively engage even with the quietest person in the room and make sure that they are included in the group. It's impossible to forget your sarcasm and witty way of looking at life. And, of course, our electric guitar lessons, swimming at de Hofbad and the "gorilla training" gym sessions with **Amin**. Best wishes to both of you and I hope to see you soon!

Jiakun, I cherish the memories of the dance classes we attended together. That was one presentation night show to remember! All the hard training paid off and I hope you and your wife will go back to ballroom someday. People call you Mr. President, rightfully so, for not many can say they have led the society of Chinese students in the Netherlands. Everyone looks up to you, myself included, and I hope being parents will make both you and Yan very happy!

Jinyu, my brother-from-another-mother, I hope the day you get to establish your start-up, OMT Industries, is not far away. I cherish the memories of our nights out and hope we get to visit China together, someday, for learning Mandarin is still on my "bucket list". Good luck with your graduation and all the happiness and good wishes for your upcoming wedding!

Xiaocong, you are one of the last to join the office, yet your energy and free-spirit have already made you stand out! Thank you for being my friend and best of luck with your work!

I can not forget you, **Mojti**, for your incredible courage in asking the prettiest girls for a dance during our nights out, as well as our gym, jogging and swimming sessions. I miss our light-spirited conversations and hope you will always find success!

Matteo, you were always a source of inspiration for me on how to be efficient and enjoy life during the PhD. Keep at it, my friend, and good luck in Livermore, or wherever you decide to venture next!

Dear, **Rafa**, which hat should I put on you?! A true friend, an encyclopedia for coding, my mentor, my confidant, my upcoming paranymph. I am simply grateful to have met you and have always found comfort and good advice in your words! Good luck to you and your girls and don't forget to drop me a line from time to time, whatever continent we happen to be on!

Nik, you were always the norm in the office, either for being a perfect gentleman or your amazing salsa moves! Indeed, you would instantly lighten everybody's mood and bring smiles on people's faces, as I'm sure you still do, over there in Edinburgh. Looking forward to join you on the island soon!

Jako, thank you for being a close friend throughout my PhD. I will never forget our conversations, as well as the great trip we had in California, from Las Vegas, all the way to San Francisco. Those were the days! I'm really glad to see you so happy as a young mother and wish you and your husband good health and success in what's yet to come around the corner!

Mark, you are most definitely the only person in the world I have shared a good steak with, in the aftermath of a natural disaster! How can I forget those tense days when we were both interns in Houston during hurricane Harvey?! All is well that ends well. I cherish our friendship and already miss our conversations! Take care and good wishes for the rest of your PhD!

Many thanks to **Ahmed** for being my desk neighbour, a great chef and a patient listener whenever I wanted to practice my Dutch. I still laugh when I remember the infamous "cat story" or that time you and Nik asked me to pick the lock on his cabinet. Very happy to hear about your engagement – may it bring you happiness and fortune!

Still, I'm sure that no one in the office will deny that **Sian** is the greatest cook there is. I really miss those Monday afternoon treats and hope you keep practicing your ballroom dancing!

Swej, we share quite the history, together, be it in Hadi's group, in the PhD office or on the basketball pitch. **Martijn**, our karting sessions will always remain in my mind. Thank you both for your company and I look forward to seeing you around in Delft!

Dear **Bander**, you were always the most knowledgeable, experienced, yet very modest, petroleum engineer in the office. I hope we get to see each other soon and I hereby add my wishes of good luck on your bright career ahead!

Yang, **Kai** and **Longlong**, I have learned so much about Chinese culture from you. Xiexie! **Brandon**, I feel very fortunate to have met you and hope you will soon find the happiness and peace of mind you deserve! God speed, my friends!

Rodrigo and **Siamak**, it was my pleasure to share the office with both of you these past years. Good luck in your work!

To **Dudu**, I say thank you for pushing the group to go outside the office and take part in more sports and social activities. Together, we conquered the summits of Mt. Etna, while eating pizzettas – that incredible view is still vivid in my memory.

Rahul, our father-figure, **Siavash**, **Durgesh**, and, of course, our superstar, **Elisa**, thank you for giving the office so much personality. Your legacy still lingers within those glass walls.

My amazing Master students, **Sander**, **Timothy**, **Mitra**, **Sebastian**, **Gusti**, **Irina**, it was a pleasure to work with you, through the good times and hiccups. I am very proud of you and will be closely watching the news for your names!

Dr.'s **Seong Lee** and **Xian-Huan Wen** – gentlemen, thank you for your warm welcome to Chevron's RPP team during my internship last summer. It was a truly memorable experience. I miss the work and the atmosphere in the HOU150 office. It is my hope that we will stay connected, now that I will start working on INTERSECT, as part of Schlumberger, in Abingdon.

I would like to express my gratitude towards the **PhD committee** members for their time and patience in reading the present manuscript. I look forward to our heated discussions during the defense!

Thanks are also due to **Khalifa Universiy** and **ADNOC** for supporting my project. Also, to the **DARSim** group for the friendly, yet competitive work atmosphere – I wish everyone good fortune moving forward!

Dana, **Dorota** and **Ghada**, thank you for picking up the work on our joint paper, where I left off! It took quite a bit of our free time over the past 4 years, but I am proud that our little pet publication finally made it to print. Good luck in your future endeavours!

Thank you **Victor**, **Roxana**, **Răzvan**, **Cristina** and **Dragoş**, my Romanian gang, for making me feel at home away from home. And, of course, **Alin**, **Tudor**, **Remus**, **Anca** and **Dan** – I miss our nights out at Locus – when will I get to see you in England?!

To **SoSalsa!**, thank you for spicing up my weekends and evenings in Delft with your great events! Also, a special thanks to the members of **Blue Suede Shoes**, and Boards I, II and III! Keep on dancing and representing TU Delft in competitions! I will do my best to stop by whenever I am in town. To my dance teachers, **Kaspar** and **Polina**, for their dedication, patience and many fun moments during our Friday classes! **Marieke**, thank you for being my amazing dance partner these past years! I still keep our NTDS medal on my wall. Success in your study and I'm certain there are many more prizes to come your way soon!

My group of friends from all over the world, **Mousa**, **Yasmin**, **Fardin**, **Mohsen**, **Leonoor**, **Cantika** and **Tim**, thanks for always being there to cheer me up during our outings and boardgame nights – I miss you all!

My dear **Alina**, I am very lucky to have met you. Thank you for your unconditional love and support! Our plan for world exploration has only just begun! Good luck in your own PhD and I am very excited to see where life takes us next!

Last but not least, I would like to thank my friends and family back in Romania, along with my brother, who has embarked on his own Dutch journey at Twente University. **Mom**, **dad**, **Mihai**, thank you for giving me the strength to overcome everything in my path. Even though we are in different parts of Europe, it always feels like you are close by! I love you dearly!

About the author

Matei Țene was born in Bucharest, Romania. He completed his BSc. in Computer Science & Engineering at the Politehnica University of Bucharest, with a specialization in Artificial Intelligence. Feeling adventurous, he then boarded the next plane for the Netherlands to pursue an MSc. in Applied Mathematics at the TU Delft, for which he was awarded a full scholarship and finished with distinction. Owing a lot to Delft for opening his eyes to the world, he decided to stay for 4 more years as PhD candidate under the supervision of Dr. Hadi Hajibeygi. Alongside his work on multiscale methods, during the past 4 years he was an avid participant in student ballroom dancing competitions. This passion motivated him to become a founding board member of the D.S.D.A. Blue Suede Shoes association for dancing students at TU Delft. As of August 2018, he has started working as Software Engineer for Schlumberger, in the INTERSECT team located in Abingdon, UK.

ISBN 978-94-6186-956-2