A framework for advanced processing of dynamic X-ray micro-CT data

Marjolein Heyndrickx

Proefschrift ingediend tot het bekomen van de graad van Doctor in de wetenschappen: fysica
Academiejaar 2019-2020

Examencommissie: Prof. dr. Philippe Smet (voorzitter)
Dr. ir. Jan Aelterman (secretaris)
Prof. dr. Luc Van Hoorebeke (promotor)
Prof. dr. ir. Matthieu Boone (promotor)
Prof. dr. Joost Batenburg
Dr. Tom Bultreys
Prof. dr. ir. Christian Vanhove

Fonds Wetenschappelijk Onderzoek

Universiteit Gent
Faculteit Wetenschappen

Vakgroep Fysica en Sterrenkunde
Campus Proeftuin
Proeftuinstraat 86 B-9000 Gent, België

Proefschrift ingediend tot het bekomen van de
graad van Doctor in de wetenschappen: fysica
Academiejaar 2019-2020

Contents

Nederlandstalige samenvatting
-Dutch summary-

X-stralen tomografie of het uitvoeren van een CT-scan is een techniek waarmee het inwendige van objecten onderzocht kan worden, zonder deze voorwerpen daarbij mechanische schade toe te brengen. CT is het bekendst omwille van zijn medische toepassingen, maar wordt ook elders gebruikt, zoals in de voedselcontrole, geneesmiddelenontwikkeling en onderzoek rond bijvoorbeeld geologie of materiaalwetenschappen. In dit werk gaat het om micro-CT. Dit is X-stralentomografie met een resolutie van de orde enkele tientallen micrometer of zelfs tot onder de 1 micrometer, waarmee relatief kleine objecten gescand kunnen worden, voornamelijk voor onderzoeksdoeleinden.
CT is inherent een driedimensionale techniek, waarmee een morfologisch 3D model (binnen- en buitenkant) van het gescande object wordt gemaakt. Het is ook mogelijk om hier de dimensie tijd aan toe te voegen en zo 4D micro-CT of dynamische micro-CT te bekomen. Dit wil zeggen dat het object aan veranderingen onderhevig kan zijn gedurende de scan. Het is precies het dynamisch proces dat deze verandering veroorzaakt, dat interessant is. Vele wetenschappers wensen zulke dynamische processen te onderzoeken en CT kan hier een goede methode voor zijn, mits we met de moeilijkheden kunnen omgaan. Deze thesis draait rond het ontwikkelen en verbeteren van de algoritmes die gebruikt worden om de data van 4D micro-CT scans te analyseren en te interpreteren.
In hoofdstuk 2 wordt X-stralentomografie uitgebreid uitgelegd, inclusief 4D micro-CT en de moeilijkheden die optreden bij het overgaan van 3D naar 4D. Hoofdstuk 3 bespreekt enkele technieken uit de literatuur die reeds zijn ontwikkeld om dynamische micro-CT mogelijk te maken, dikwijls voor een specifieke klasse van toepassingen.

Deze thesis zoekt verbeteringen voor twee types algoritmes om om te gaan met 4D micro-CT scans. De eerste soort zijn reconstructiealgoritmes. Een reconstructie is het berekenen van een virtueel 3D (of 4D in dit geval) volume uit de scandata, een volume dat het onderzochte staal beschrijft. De tweede soort is de analyse. Dit is heel variabele methodologie waarmee de informatie

wordt bekomen waarvoor het experiment in kwestie werd uitgevoerd, meestal (maar niet altijd) vertrekkende van het reconstructieresultaat.

In hoofdstuk 4 ontwikkelen we een nieuwe reconstructiemethode: de gewogen terugprojectie of 'weighted back projection'. Dit algoritme is een aangepaste versie van een iteratieve reconstructie, een van de twee belangrijkste klassen van reconstructiemethoden in micro-CT. Het maakt gebruik van twee soorten a priori kennis over het staal, i.e. kennis over het onderzochte object die we al kennen onafhankelijk van de 4D micro-CT scan zelf.
Ten eerste is er een volume nodig dat het staal in kwestie vrij goed beschrijft. Dit volume noemen we het initieel volume. Dit kan bijvoorbeeld een 3D model zijn in het geval van een gefabriceerde component, of het resultaat van een hoge kwaliteit micro-CT scan van hetzelfde object voor het dynamische proces in gang werd gezet. In deze thesis wordt een dergelijke hoge kwaliteitsscan gebruikt.
De tweede soort kennis die we nodig hebben is informatie over welke regio's in het initieel volume een hogere waarschijnlijkheid hebben om te veranderen omwille van het dynamische proces. Dit betekent dat gewogen terugprojectie gebruikt kan worden voor dynamische processen in een statisch rooster, een belangrijke klasse van onderzoeksproblemen. Een goed voorbeeld is vloeistofstroom doorheen een geologisch materiaal: de vloeistof kan enkel doorheen de poriën stromen. Dit voorbeeld zal doorheen deze thesis gebruikt worden.
Met deze twee vormen van a priori kennis komt de gewogen terugprojectie tot stand. Deze reconstructie slaagt erin om uit veel beperktere datasets - dat wil zeggen, datasets die veel minder radiografieën bevatten dan gebruikelijk - de benodigde informatie te kunnen halen, wat zeer goed van pas kan komen. Wel dienen er maatregelen te worden genomen om de ruis onder controle te houden, die in de gewogen terugprojectie sterker tot uiting komt dan in een 'normale' reconstructie. Dit kan eenvoudig worden gedaan via het verkleinen van de relaxatiefactor, een van de parameters in een iteratieve reconstructie. Samenvattend is de gewogen terugprojectie een krachtige reconstructietechniek, vooral wanneer een klein aantal radiografieën per rotatie technisch realistisch is op de gebruikte CT-scanner.

In hoofdstuk 5 bekijken we een methode die tussen de reconstructie en de analyse in staat: het fitten van een stuksgewijze rechte of 'piecewise linear fitting'. De basis van deze techniek is heel eenvoudig: voor elke voxel (een 3D pixel) in het reconstructievolume hebben we na de 4D reconstructie een attenuatiecoëfficiënt voor verschillende tijdspunten. Aan deze attenuatie in functie van tijd wordt een stuksgewijze rechte gefit: dit is een opeenvolging van verschillende rechten. Dit is een zeer eenvoudige functie, waardoor ze makkelijk te interpreteren valt en de fit relatief snel kan gebeuren. Boven-

dien kan deze functie elke andere functie benaderen, waardoor we ze voor elk mogelijk dynamisch proces kunnen gebruiken.
Op deze manier wordt de ruis sterk onderdrukt in de reconstructie. Bovendien zijn de gefitte parameters een goede basis voor de verdere analyse van het dynamisch proces, zoals we aantonen in het laatste deel van deze thesis, hoofdstuk 6.
In hoofdstuk 6 analyseren we, door te starten van piecewise linear fitting, de vloeistofstroom doorheen een Bentheimer zandsteen. We combineren deze analyse met een bestaande 3D analyse, waarmee de poriën in het volume worden geïdentificeerd. We bekijken via het resultaat van de piecewise linear fitting de doortocht van de vloeistof doorheen de poriën en komen zo te weten wanneer de vloeistof specifieke poriën binnenstroomt.

De technieken ontwikkeld in deze thesis zijn heel waardevol in verscheidene 4D micro-CT toepassingen. Toch is er ook hierna nog veel werk te doen in het gebied om het mogelijk te maken om meer dynamische processen te onderzoeken en zo steeds meer te weten te komen over onze wereld. Ik heb er vertrouwen in dat de ontwikkelingen in de komende jaren ons hier steeds dichter naartoe zullen brengen.

English summary

Computed tomography or CT scanning is a technique that can be used to examine the interior of objects without causing mechanical damage to these objects. CT is best known for its medical applications, but it is also used elsewhere, such as in food control, drug development and research in geology or material sciences. This work is about micro-CT or μCT. This is computed tomography with a resolution of the order of some tens of micrometers, down to even below 1 micrometer, with which relatively small objects can be scanned, mainly for research purposes.
CT is inherently a three-dimensional technique to create a morphological 3D model (inside and outside) of the scanned object. It is also possible to add the time dimension and thus obtain 4D micro-CT or dynamic micro-CT. This means that the object can undergo changes during the scan. It is precisely the dynamic process that causes these changes that is interesting. Many scientists wish to investigate such dynamic processes and CT is an ideal method for this, provided we can deal with the difficulties. This thesis revolves around developing and improving algorithms to deal with the data from 4D micro-CT scans.
Chapter 2 explains X-ray tomography in great detail, including 4D micro-CT and the difficulties that arise when moving from 3D to 4D. Chapter 3 discusses some techniques from the literature that have already been developed to enable dynamic micro-CT, often for a specific class of applications.

This thesis is looking for improvements to two types of algorithms to deal with 4D micro-CT scans. The first type are reconstruction algorithms. A reconstruction is the calculation of a virtual 3D (or 4D in this case) volume from the scan data, a volume that describes the sample examined. The second type is the analysis. This is highly variable methodology that obtains the information for which the experiment was performed, usually (but not always) starting from the reconstruction result.

In chapter 4 we develop a novel reconstruction method: weighted back projection. This algorithm is a modified version of an iterative reconstruction, one of the two most important classes of reconstruction methods in μCT. It

uses two types of prior knowledge about the sample, i.e. knowledge obtained independent of the 4D μCT scan.
First, a volume is required that describes the sample in question relatively well. We call this volume the initial volume. For example, this can be a 3D model in the case of a fabricated component, or the result of a high-quality μCT scan of the same object before the dynamic process was initiated. Such a high quality scan is used in this thesis.
The second type of knowledge needed is information on which regions in the initial volume have a higher probability of changing due to the dynamic process. This means that weighted back projection can be used for dynamic processes in a static grid, an important class of research problems. A good example is fluid flow through a geological material: the fluid can only flow through the pores. This example will be used throughout this thesis.
These two forms of a priori knowledge make weighted back projection possible. Weighted back projection succeeds in extracting the required information from severely limited data sets, i.e. containing fewer radiographs than usual, which can be very useful. However, measures must be taken to keep the noise under control, which presents itself more strongly in weighted back projection compared to a conventional reconstruction. This can easily be realised by lowering the relaxation factor, one of the parameters in an iterative reconstruction. In summary, weighted back projection is a powerful reconstruction technique, especially when a low number of radiographs per rotation is technically realistic on the used CT scanner.

In chapter 5 we examine a method situated in between the reconstruction and the analysis: piecewise linear fitting. The basis of this technique is very simple: after the 4D reconstruction, we have an attenuation coefficient for different points in time for each voxel (a 3D pixel) in the reconstruction volume. A piecewise linear function (a number of lines with breakpoints between them) is fitted to this attenuation in function of time. This is a very simple function, making it easy to interpret and the fitting procedure is relatively fast. Moreover, this function can mimic any other function, so that we can use it for every possible dynamic process.
Piecewise linear fitting decreases a large part of the noise in the reconstruction. Moreover, the fitted parameters are a good basis for further analysis of the dynamic process, as we demonstrate in the last part of this thesis, chapter 6.
In chapter 6, starting from the result of piecewise linear fitting, we analyse the fluid flow through a Bentheimer sandstone. We combine this analysis with an existing 3D analysis, which identifies the pores in the volume. We use the result of the piecewise linear fitting to observe the passage of the liquid through the pores and to find out when the liquid enters specific pores.

The techniques developed in this thesis are very valuable in various 4D μCT applications. Nevertheless, there is still much research to be done to make it possible to investigate more dynamic processes and thus learn more and more about our world. I am confident that developments in the coming years will continue to bring us closer to this goal.

Dankwoord

Niemand legt de weg naar een doctoraatsthesis alleen af, ik ook niet. Er zijn veel mensen die ik wil bedanken omdat ze me hebben bijgestaan bij dit grote werk.

Ten eerste Luc Van Hoorebeke, mijn promotor, die me de kans heeft gegeven om te doctoreren en die altijd klaar stond met uitleg en op wie ik altijd kon rekenen om alles tot in de puntjes na te lezen. Luc, je grapjes en je niesbuien zijn een gekend gegeven onder de hele groep en ik ga beide missen (hoewel de niesbuien alleen als ik ver genoeg uit de buurt ben).
Matthieu, mijn co-promotor, bedankt dat je mijn begeleiding ook in je al overvolle agenda wist te krijgen. Je was een enorme hulp waar ik bij terecht kon met vragen, voor nieuwe ideeën en ook om mijn teksten na te lezen. Bovendien waren je broodpuddings en je radijsjes altijd een welkome toevoeging aan het middageten.
Amelie, mijn "meter" toen ik net begon, bedankt om mij altijd de praktische dingen uit te leggen die iedereen magischerwijs weet, zoals waar ik het doctoraatsreglement kon vinden of hoe de binnenpost werkt. Bedankt ook voor de levensechte uitbeeldingen van grappige kleine Floris.
Sander, bedankt om zo'n toffe bureaugenoot te zijn en om mijn computer te fixen toen hij willekeurig bleef uitvallen. Bedankt ook om de website over te willen nemen, ik geloof dat ze bij jou in goede handen is.
Frederic, tegenwoordig komt bijna al mijn informatie over politiek van u en Sander en om de een of de andere reden moet ik altijd lachen met hoe jullie die uitleggen. Bedankt voor de vele pannekoeken en voor het idee van de broodzak.
Wannes, bedankt om de gigantische taak van het opkuisen van ctrex op je te hebben genomen. Het ziet er veel beter uit en ik heb pythontrucjes geleerd uit jouw stukken code die ik voorheen niet kende.
Bo, you often say things are 'awesome' and I want you to know that applies to you too. In only a short time you've done a lot of research work in our group.
Ivan, bedankt voor het barbecuevlees en de altijd vrolijke aanwezigheid. Het was zeer opvallend hoeveel er plots gedaan raakte toen jij bij de groep kwam.

Abhishek and Niloofar, thanks for opening my view on the world with your explanations about your home cultures.
Thomas De Schryver, hoewel je nu al een tijdje niet meer bij het UGCT bent, was het vooral jouw begeleiding die me op weg heeft geholpen om de wereld van μCT te leren kennen. Ik heb ook steeds met plezier verder gewerkt met de code waar jij aan begonnen bent.
Ik wil ook alle andere mensen van het INW bedanken, voor de ijsjes en de gesprekken over de middag en geweldige sfeer. Bedankt!

Bedankt ook aan de leden van mijn jury, voor het grondig lezen van mijn thesis, voor de interessante discussies en voor de suggesties die het eindresultaat veel beter hebben gemaakt.

Bedankt, mama, om me te brengen waar ik nu ben, om me mijn hele leven te hebben bijgestaan, ook al was dat zeker geen gemakkelijke opgave. Je stond er vaak alleen voor en je deed het toch. Bedankt!
Kasper, bedankt voor de reis naar China en de verbreding van mijn wereldbeeld. Anton, bedankt voor de memes en de muziek. Ik ken zo veel goede liedjes door u. Beide bedankt om altijd klaar te staan om te helpen en voor alles wat ik door jullie heb leren kennen.
Bedankt, papa, voor de vele spelletjes en de vaak vreemde maaltijden. Marina, ik zie dat mijn vader gelukkiger is sinds hij jou kent. Bedankt ook voor de naaitips en de patronen.
Bedankt, meme en pepe, voor de lantaarntochten en de paaseieren en het steeds samenbrengen van de hele familie.
Bedankt ook aan de rest van de familie. Ik kan steeds op jullie rekenen voor hulp en we hebben samen veel leuke dingen gezien.

Ik wil ook mijn vrienden bedanken.
Tahira, we hebben samen al veel meegemaakt. Bedankt om ons met dungeons & dragons te laten kennismaken en bedankt om de groep samen te brengen.
Isabelle, Liesbet en Kevin, bedankt voor alle grappige momenten, de verhalen die we samen hebben gemaakt, de snacks en de weekends aan zee.
Kasper Meerts, bedankt om me te laten zien dat volharding in de wetenschap uiteindelijk resultaten oplevert, zelfs wanneer er grote blokkades in de weg liggen.
Yana, we gaan al terug tot de kleuterschool en in al die jaren zijn we steeds goede vrienden gebleven. Bedankt voor de vele mooie momenten. Bedankt voor de katten en de kangoeroes.
Ellen en Tina, bedankt om me te willen includeren in de groep. Bedankt voor de marshmallows en de chalet en het vele gelach.
Robin, Aurelie, Simon, Michelle, Ken, Jarmo, Tine, Marjorie en Jonatan,

bedankt voor de vriendschap tijdens de studie en daarna. Zonder onze extensieve vragenmails tijdens de examens zou ik het materiaal nooit zo goed onder de knie hebben gekregen, en de uitgebreide bespreking van verkiezingen gaf me ook daarin veel meer inzicht.

Bedankt, Yorick, voor alles. Bedankt om te bestaan, bedankt om in mijn leven te zijn, bedankt voor het lekkere eten, bedankt voor de uitstapjes, bedankt om aan mijn zij te staan. Ik kan blijven dingen opsommen waar ik je voor wil bedanken en dan zal ik uiteindelijk bijna elk aspect van ons leven hebben beschreven. Bedankt voor alles.

Hartelijk bedankt iedereen!

Terminology

CBCT	Cone beam CT
Computed tomography	A non-destructive technique for visualising the inside of an object in 3D
CT	Short for computed tomography
Dark field	A radiograph used for normalisation. It records the response of the detector when the X-ray beam is turned off. See section 2.3.5.
DF	Short for dark field image
DVC	Short for digital volume correlation. See section 3.3.
EMCT	A CT-scanner of the UGCT often used for dynamic scans
FBP	Short for filtered back projection. See section 2.4.1.
FDK	An analytic reconstruction technique for a cone beam set-up. See section 2.4.1.
GPU	Short for graphical processing unit. An electronic circuit usually responsible for the creation of images for a display.
HECTOR	High Energy CT Optimized for Research, a CT-scanner of the UGCT.
IO	Short for intensity offset, which is the flat field image
Micro computed tomography	Computed tomography with a resolution of μm or better
MVF	Short for motion vector field. See section 3.3.
Scanning	Performing computed tomography on an object.
UGCT	The UGent centre for X-ray tomography
UGent	Ghent University
μCT	Short for micro computed tomography

1

Introduction

What is this thesis about?

High resolution X-ray computed tomography (μCT) is an important tool in multiple research domains to non-destructively visualise and analyse samples in 3D, both internally and externally [1, 2]. Its non-destructive nature makes μCT suited to be extended to the scanning of samples undergoing dynamic processes during a scan. As such, this so-called dynamic μCT or 4D μCT has gained interest as a technique to visualise dynamic processes taking place inside samples. The dynamic process is undisturbed by the measuring technique (μCT). However, the processing and analysis of this 4D μCT data is still challenging, facing motion artefact complications and high noise levels. By optimising these algorithms, we hope to push 4D μCT to become a measurement technique for faster and more difficult dynamic processes.
The processing of 4D μCT data in this PhD research and the developed techniques were implemented in ctrex, a hybrid code of Python and C, using parallel computing on graphical processing units (GPU) and the open-source module PyCuda [3].

This research was performed under an SB-grant ('Strategisch Basisonderzoek') of the Research Foundation Flanders (FWO). The goal, as described in the original project application, was threefold:

1. Develop methods to use prior knowledge in an iterative reconstruction

to improve the reconstruction quality.

2. Extend the existing 3D analysis methods to 4D, with time being the extra dimension.

3. Combine both reconstruction and analysis in one framework.

This dissertation will describe the progress that was made towards all three of these goals.

1.1 Research group

The research was performed at the Ghent University Centre for X-ray Tomography (UGCT) [4]. This is a collaboration between three Ghent University research groups: the Radiation Physics group (Department Physics and Astronomy, Faculty of Sciences, with which I am associated), the PProGRess group (Department of Geology, Faculty of Sciences) and the Laboratory of Wood Technology (Department of Environment, Faculty of Bioscience Engineering). The UGCT has five operational, home-build CT scanners ([4, 5, 6, 7]) and performs research on computed tomography and its applications.

1.2 Outline

The background for this PhD research is described extensively in chapter 2. We will first describe μCT and its applications and then explain in more detail how exactly μCT works. Special attention will go to reconstruction algorithms, as adaptations to these are one of the core focusses of this PhD research. Section 2.5 describes in detail some of the artefacts, i.e. imaging errors, that can occur in μCT and that should be compensated for and kept in mind when interpreting μCT results. After this, the chapter provides an explanation of 4D μCT, the core research area in which this PhD research is situated. We also describe the most used CT scanner in this work, the EMCT, and the sample on which the techniques developed in this PhD research will be demonstrated, a Bentheimer sandstone.

A literature review in the context of 4D μCT is provided in chapter 3. We will provide an (incomplete) overview of some of the most used techniques in the context of dynamic CT, along with the advantages, disadvantages and application area of each. As we will see, most techniques are applicable to specific applications and are less ideal in others. Therefore, depending on the sample and the dynamic process under investigation, different techniques can be chosen.

We developed an adapted iterative reconstruction algorithm which uses prior knowledge and is described in chapter 4: weighted back projection. We explain the algorithm, which uses two types of prior knowledge: an initial volume and a weight volume. The initial volume is a volume resembling the sample, typically obtained from a high-quality μCT scan before the dynamic process was initiated, although it can be a 3D model of a fabricated component as well. The weight volume describes in which areas in the sample the dynamic process is most likely to occur. Both these volumes are incorporated into an adapted version of an iterative reconstruction. In this work, this iterative reconstruction algorithm is SART, but any other can be adapted similarly.
In chapter 4, we will compare the results obtained with four different weight volumes and see which effect this choice has on the reconstruction. Such a weight volume can be either continuous or discrete and it can have zero weights or not. Chapter 4 will show the importance of choosing the right weight volume. We will test the ability of weighted back projection to reconstruct a volume from a limited amount of projections using the Bentheimer sandstone sample and only 10% of the available projections. This simulates a scan that was performed faster than the available scan, while still having the 'slower' data available as a benchmark to compare the results with.
A disadvantage of weighted back projection is the high susceptibility to noise. This can be compensated by setting a low relaxation factor, one of the parameters in an iterative reconstruction. We found this to be the simplest way to decrease the noise when comparing different methods in section 4.6.

Chapter 5 describes a novel technique which connects reconstruction to analysis. It starts from a 4D reconstruction, which can be made with any reconstruction algorithm, and fits a piecewise linear function to the time evolution of the attenuation coefficient of each voxel (3D pixel). The algorithm and the reasoning behind its aspects are explained at the start of the chapter. After this, the algorithm is demonstrated on the Bentheimer sandstone sample.
Piecewise linear fitting decreases the noise in the reconstruction significantly. This noise filtering also occurs in the spatial directions, even though piecewise linear fitting only intervenes in the temporal dimension. Despite the noise reduction in the spatial dimensions, spatial resolution is not affected. As will be demonstrated in chapter 5, the noise filtering effect of piecewise linear fitting can be similar to 'real' noise filters. These exist in plenty of different forms and therefore noise reduction was not the primary reason to develop piecewise linear fitting, only an added bonus. Piecewise linear fitting is a step towards analysis, which is its most important aspect.

An analysis following from piecewise linear fitting is explained in chapter 6, where we also use conventional 3D analysis. We use, for each voxel, the parameters from the piecewise linear fit to determine at which time the dynamic process influenced this voxel - if it influenced this voxel at all. This is combined with a 3D pore segmentation to examine the time of fluid entry for each pore, with a spread signifying noise levels and uncertainty. Overall, piecewise linear fitting proves to be a good tool for an analysis of this sample.

Finally, chapter 7 outlines a very short overview and the general conclusion for this PhD research.

1.3 Main contributions

The main contributions of the author to the field of μCT are:

- Developed and characterised weighted back projection, an iterative reconstruction technique using two forms of prior knowledge: an initial volume and a weight volume.
- Developed piecewise linear fitting: a post-processing technique specifically for 4D μCT which acts as a stepping stone towards analysis.
- Demonstrated the possibility to perform data analysis starting from a piecewise linear fit.
- Heavily extended the reconstruction framework ctrex, a GPU tool using Cuda and Python.

1.4 Publications related to this work

Journal articles:

- "Piecewise linear fitting in dynamic micro-CT"
  Marjolein Heyndrickx, Matthieu Boone, Thomas De Schryver, Tom Bultreys, Wannes Goethals, Glenn Verstraete, Valérie Vanhoorne and Luc Van Hoorebeke. Materials Characterization. 139. p.259-268 (2018).
- "Improving image quality in fast, time-resolved micro-CT by weighted back projection"
  Marjolein Heyndrickx, Matthieu Boone, Tom Bultreys, Wannes Goethals and Luc Van Hoorebeke. (submitted at Optics Express)

Conference contributions:

- "Effect of an initial solution in iterative reconstruction of dynamically changing objects"
  Marjolein Heyndrickx, Thomas De Schryver, Manuel Dierick, Matthieu Boone, Tom Bultreys, Veerle Cnudde and Luc Van Hoorebeke. Tomography of Materials and Structures, 2nd International conference, Proceedings. p.172-176 (2015).

- "Improving the reconstruction of dynamic processes by including prior knowledge"
  Marjolein Heyndrickx, Thomas De Schryver, Manuel Dierick, Matthieu Boone, Tom Bultreys, Veerle Cnudde and Luc Van Hoorebeke. HD-Tomo-Days (2016).

- "Dynamic CT reconstruction using piecewise linear fitting"
  Marjolein Heyndrickx, Matthieu Boone, Tom Bultreys and Luc Van Hoorebeke. 3rd International conference on tomography of materials and structures (2017).

- "4D-μCT analysis through piecewise linear fitting"
  Marjolein Heyndrickx, Matthieu Boone, Thomas De Schryver, Tom Bultreys, Wannes Goethals, Glenn Verstraete, Valérie Vanhoorne and Luc Van Hoorebeke. Porous Media, 10th International conference (2018).

- "Weighted backprojection: a reconstruction technique for dynamic micro-CT"
  Marjolein Heyndrickx, Matthieu Boone, Wannes Goethals, Tom Bultreys and Luc Van Hoorebeke. Tomography of Materials and Structures, 4th International conference (2019).

References

[1] V. Cnudde and M. N. Boone. *High-resolution X-ray computed tomography in geosciences: A review of the current technology and applications.* Earth-Science Reviews, 123:1–17, 2013.

[2] E. Maire and P. J. Withers. *Quantitative X-ray tomography.* International Materials Reviews, 59(1):1–43, 2014.

[3] A. Klöckner, N. Pinto, Y. Lee, B. Catanzaro, P. Ivanov, and A. Fasih. *PyCUDA and PyOpenCL: A scripting-based approach to GPU run-time code generation.* Parallel Computing, 38(3):157–174, 2012.

[4] B. C. Masschaele, V. Cnudde, M. Dierick, P. Jacobs, L. Van Hoorebeke, and J. Vlassenbroeck. *UGCT: New X-ray radiography and tomography facility.* Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 580(1):266–269, 2007. Proceedings of the 10 th International Symposium on Radiation Physics.

[5] B. Masschaele, M. Dierick, D. Van Loo, M. N. Boone, L. Brabant, E. Pauwels, V. Cnudde, and L. Van Hoorebeke. *HECTOR: A 240kV micro-CT setup optimized for research.* Journal of Physics: Conference Series, 463(1):012012, 2013.

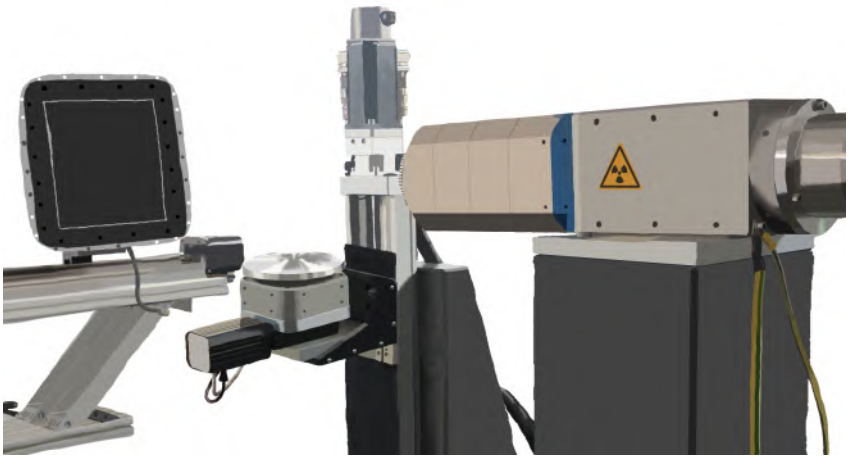
[6] M. Dierick, D. Van Loo, B. Masschaele, J. Van den Bulcke, J. Van Acker, V. Cnudde, and L. Van Hoorebeke. *Recent micro-CT scanner developments at UGCT.* Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 324:35 – 40, 2014. 1st International Conference on Tomography of Materials and Structures.

[7] B. Laforce, B. Masschaele, M. N. Boone, D. Schaubroeck, M. Dierick, B. Vekemans, C. Walgraeve, C. R. Janssen, V. Cnudde, L. Van Hoorebeke, and L. Vincze. *Integrated three dimensional micro-analysis combining X-ray microtomography and X-ray fluorescence methodologies.* Analytical Chemistry, 2017.

2

Dynamic computed tomography

Background information



This thesis is situated in the domain of high resolution X-ray micro computed tomography (μCT). Specifically, the goal is to improve the techniques for dealing with dynamic μCT, also known as 4D μCT. The current chapter will clarify the basic questions about this technique, such as: What exactly is computed tomography? How does it work? What do people use it for? What are the difficulties?

2.1 What is X-ray computed tomography?

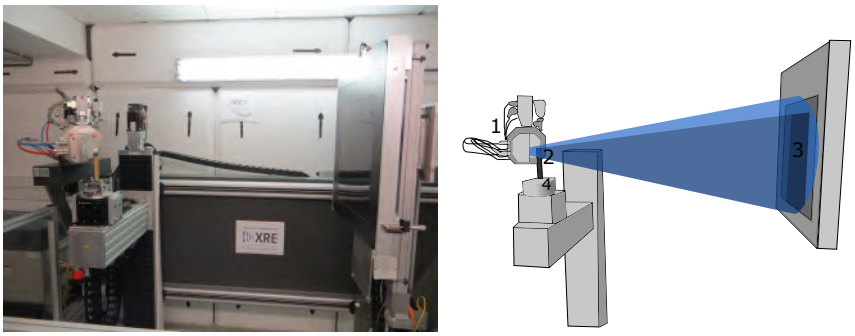


Figure 2.1: The high energy CT scanner optimised for research (HECTOR [1]), one of the CT scanners at the UGCT ([2], section 1.1), with some components indicated. 1. The X-ray tube. 2. The object under investigation, i.e. the sample. 3. The detector. 4. The rotation stage. The blue cone shows the X-ray beam. HECTOR has the geometry of the bottom left set-up of figure 2.7.

Computed tomography or CT-scanning is a technique for visualising the exterior and interior of an object in 3D (three spatial dimensions). The most important aspect of CT is that it is (mostly) non-destructive [3, 4]. The sample which is scanned will be intact after the scan, since it has not been cut open or otherwise mechanically damaged. Only in-vivo biological samples and X-ray sensitive materials might experience harmful effects from the radiation associated with the measurement [5]. This is especially important in medical research areas making use of CT and radiation sensitive materials and plants in μCT [6]. However, since this does not apply to the samples used in this research, the CT-scans are regarded as non-destructive in the scope of this work.
Computed tomography is performed with a system called a CT-scanner or a CT scanning system. There is a large variety of scanners, depending on their purpose. In this work, we focus on high-resolution CT systems (μCT), as

opposed to the more well known medical CT systems. Micro-CT or μCT is defined as computed tomography with a resolution of the order of 100 µm to 1 µm or better. In μCT, the scanned samples are typically a few mm to a few cm big. However, much larger samples can also be scanned, albeit generally at a worse resolution.
Figure 2.1 portrays an example of a lab-based scanner, with the key parts specifically indicated (more on this in section 2.3).
CT scanners can differ from each other not only in brand and specifications of the components such as source or detector, but also in the mechanical set-up (e.g. which part rotates, see for example section 2.7) and the X-ray beam geometry. Section 2.3.2 explains this further.

2.2 Applications of computed tomography

A highly advantageous property of CT is its ability to investigate the interior of objects in a non-destructive way. The most well-known application is medical imaging, for example the search for tumors in various parts of the body [7, 8]. Airport security mostly uses radiography, but sometimes checks luggage with CT as well [9].
Whenever it is useful to investigate the inside of an object in 3D without cutting it, computed tomography is a suitable technique. Researchers use computed tomography for their work in different fields, such as palaeontology [10], biology of small animals [11], woven textile [12], geology [13, 14], archaeology [15], wood research [16], materials science [17] and evaluation of medical devices [18].
In most of these examples, the object does not change significantly during imaging. However, there is a growing interest in imaging dynamic processes which occur inside the sample. This is possible thanks to the non-destructive nature of CT.
4D μCT or CT sampling of dynamic processes is a sub-domain of CT research and has applications in various scientific disciplines. Indeed, a lot of research requires the ability to observe dynamic processes in situ, i.e. inside samples as they take place. The process could be fluid filling up pores [19], crack formation under stress [20], the weathering of building stones [21] or limestones [22], self-healing materials upon fracturing [23], deformation of foams [24], evolution of the bubble sizes after mixing of dough [25], oil removal from sponges [26], structural changes due to temperature variations [27] and many others.

Figure 2.2: Photographic picture (left) and X-ray image (right) of a baboon mummy from the National Museum of Antiquities in Leiden. [28, 29]

2.3 How does computed tomography work?

Computed tomography [30] is the 3D extension of X-ray radiography (i.e. 2D X-ray imaging). When X-rays, originating from an X-ray source, encounter an object, they pass through it like visible light through glass [31] and only a fraction of the beam will be absorbed or removed from the beam in another way. The amount of attenuation (i.e. absorption or scattering, more on that in section 2.3.1) depends on the internal structure of the object. Therefore, the X-rays leaving the object will contain a 'shadow image' (as in the right of figure 2.2) of the interior of the object. For example, if the sample contains both a dense material and a light material, the X-rays passing through the dense material will be attenuated more. Therefore the transmitted X-ray beam that went through the dense part of the object will have a lower intensity than its counterpart that went through the light part. These transmitted X-ray beams and the information they contain can be measured with an X-ray detector.
The combination of the X-ray source, sample and X-ray detector therefore produces an X-ray image as in figure 2.2, called a radiograph or projection. The most dense materials (bones) that attenuate X-rays the most show on a radiograph as a low intensity, i.e. black, while for example air leaves the intensity of the X-ray beam mostly intact, showing white. Note that in medical imaging typically the negative image is used. The best-known applications of such images are medical and security (other applications can be found in section 2.2). The skeleton is clearly visible on these images, because it attenuates X-rays more than the surrounding tissue [32].
An X-ray image is transmission based and therefore inherently 2 dimensional.

To realise a 3D technique, the CT scanner takes X-ray images from different angles. Usually this is realised by having components that rotate: the scanner will collect multiple X-ray images during this rotation, resulting in X-ray images taken from different angles, each time showing the sample projected from a different direction. All of these images can be combined to calculate the 3D representation of the sample.
Section 2.3.2 explains how the scanner components are positioned and move with respect to each other. Sections 2.3.3 and 2.3.4 explain more about the components of a computed tomography scanner. Section 2.3.1 describes how X-rays interact with the sample to produce a 'shadow' or X-ray image. Section 2.4 discusses the software algorithms that produce a 3D image from the collection of 2D X-ray images.
In summary, a CT scan is performed as follows:

1. The X-ray source, pointing to the sample and the detector, is turned on.
2. A radiograph is recorded: the 'shadow image' from the sample on the detector.
3. The scanner rotates a small step (either the sample or the detector-source combination) .
4. A new radiograph is taken in this new position .
5. Repeat point 3 and 4 until a complete rotation has been performed.
6. A computer algorithm called a reconstruction transforms the resulting set of radiographs into a 3D image.

2.3.1 Interaction of X-rays with matter

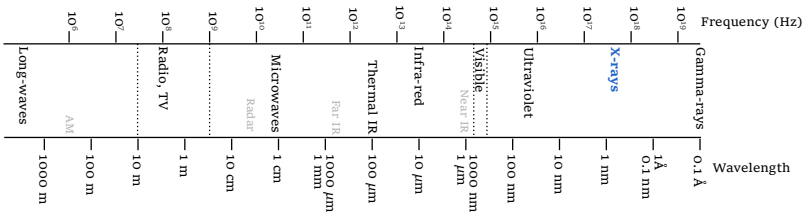


Figure 2.3: The electromagnetic spectrum with X-rays in bold. Adapted from [33]

X-rays are commonly defined as electromagnetic waves with a wavelength between 1 pm and 10 nm [34]. Note that this definition includes γ-rays, while

the more precise physical definition of X-rays includes their origin: charged particles (electrons outside the nucleus) emit X-rays, while γ-rays are emitted by nuclei or originate in annihilation reactions between matter and antimatter [35]. We will use the more common definition in the following and thus include γ-rays as if they are X-rays, since a γ-ray and an X-ray of identical wavelength have identical properties.
X-rays were discovered by W. Röntgen in 1895 [31]. As Röntgen has already noticed in his original article, X-rays pass through most materials, but the change in intensity after passing depends on the density of the material and "something else" (which turned out to be material composition). When putting more than one material next to each other in the path of the X-ray beam, this results in a shadow image (point 14 in [31]), where you can see a 2D projection of these different materials, for example the bones in a hand. The intensity of an X-ray beam passing through material A might decrease more than one passing through material B, causing these shadows. This section is about the processes that cause this intensity loss for X-ray beams passing through matter.
The intensity of a beam is proportional to the number of photons in it. An example: a monochromatic X-ray beam originates at the source at intensity $I_0 = \mathrm{E}{\cdot}10^4\,\mathrm{photons/s/cm^2}$ (with E the energy per photon), passes through a perfect mono-material slab of 1 cm thick and arrives at the other side with $I = \mathrm{E}{\cdot}7 \times 10^3\,\mathrm{photons/s/cm^2}$. This means the beam was attenuated by 0.3 or 30 %.
The equation governing the attenuation of X-rays is called the Lambert-Beer law [36, 37, 38]. In its simplest form, it reads:

$$I = I_0 e^{-\mu d} \tag{2.1}$$

with I the transmitted intensity of the X-ray beam, I_0 the original intensity, μ the attenuation coefficient of the material and d the thickness of the material through which the beam propagated. This is the equation for a monochromatic beam (i.e. only one photon energy) and a material of uniform composition and density. The slab material in the previous example therefore has an *attenuation coefficient* $\mu \approx 0.36\,\mathrm{cm^{-1}}$, resulting in $e^{-\mu d} = 0.7$.
Equation 2.3 presents the more general case of a polychromatic X-ray beam passing through an object.

$$I = \int_{\mathcal{E}} I_0(E) \cdot \exp\left(-\int_{\mathcal{L}} \mu(s,E)ds\right) dE \tag{2.2}$$

$$= \int_{\mathcal{E}} I_0(E) \cdot \exp\left(-\int_{\mathcal{L}} \frac{\mu}{\rho}(s,E)\rho(s)ds\right) dE \tag{2.3}$$

In this equation, $\mathcal{E}$ is the energy spectrum of the incident X-ray beam, E is the energy, $\mathcal{L}$ is the path followed by the X-ray, s is a position coordinate within

the material, ds is an infinitesimal path length and $\rho(s)$ is the local density of the material. $\rho(s)$ is present in this equation because μ depends on the density, the Z-value of the material and the energy of the X-rays. Since the density-dependence is directly proportional, this dependence can be removed through division and $\frac{\mu}{\rho}$ can now be tabulated for specific Z's and energies as in the database of Berger *et al.* [39]. Figure 2.4 shows an example. In both these figures, absorption edges are visible (the sharp increase or decrease in attenuation). An absorption edge appears at the binding energy of electrons in a particular shell. Once the incoming photon has this energy or higher, an additional photo-electric interaction (more on that later) becomes available for the incoming photon, thus increasing the attenuation.

The linear attenuation coefficient is μ (unit m^{-1}) and the mass attenuation

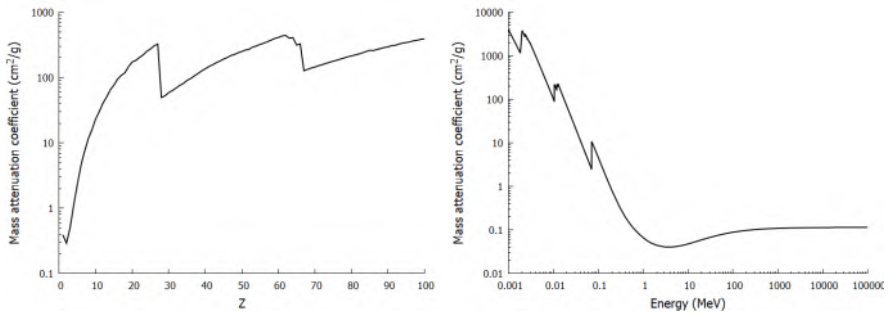


Figure 2.4: Graph of the mass attenuation coefficient $\frac{\mu}{\rho}$ in function of Z at a fixed energy of $8 \times 10^{-3}\,\mathrm{MeV}$ (left) and as a function of the energy at a fixed $Z = 75$, i.e. the element Rhenium (right). The data in these graphs was taken from the database of Berger *et al.* [39].

coefficient is μ/ρ (unit m^2/kg since ρ has unit kg/m^3). This attenuation is due to several physical interactions between the X-rays and the material, which can give rise to both absorbing and scattering the X-rays, the latter removing the X-rays from the primary beam. For the energies most used in computed tomography the important interactions are [40, 4, 41]:

- Photo-electric absorption
- Compton scattering
- Rayleigh scattering

Pair production and nuclear reactions can also appear, but these only occur at higher energies and are as such not relevant in this work. Figure 2.5 gives a schematic representation of these interactions and figure 2.6 shows their

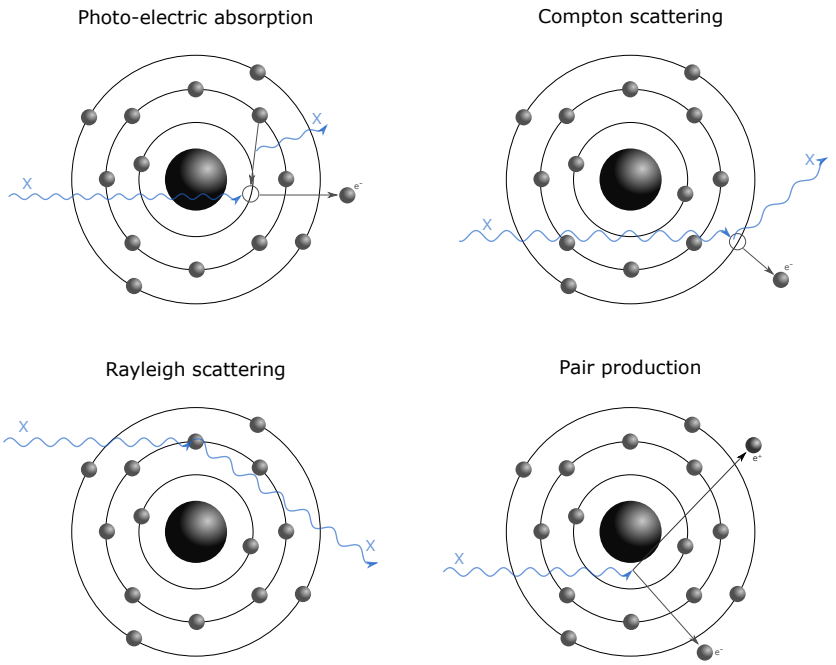


Figure 2.5: Different interactions between an X-ray and an atom. The incident X-photon is depicted by the wavy blue line.

relative magnitude at a particular energy for an example material, iron. In the following, these interactions will be described.
An incident X-ray photon arriving at an atom may bump into an electron, transfer all its energy (i.e. be absorbed) and in this way knock the electron out of its shell. This can only happen when the X-ray energy is higher than the binding energy of said electron. This is **photo-electric absorption**.
Photo-electric absorption leaves a hole in a shell of the atom, where the electron was. This can be filled fast by an electron at a higher shell, where the energy difference between its original position and the new, lower-energy shell will be emitted as a fluorescent photon, with an energy specific for the atom from which it was emitted.
The probability q_{pe} of photo-electric absorption happening typically increases with the Z-value of the material and decreases with the incident photon energy according to equation 2.4 (see figure 2.4).

$$q_{pe} \sim \frac{Z^3}{E^3} \tag{2.4}$$

This formula is only valid away from the absorption edges, i.e. the threshold values where a photon has sufficient energy for a new shell of electrons to interact through photo-electric absorption. Examples of such edges are visible in figure 2.6 at 7.112×10^{-3} MeV and in figure 2.4.
An X-ray photon can also bump into an electron and only transfer part of its energy. This is **Compton scattering** or incoherent scattering. The photon will be scattered: it will continue in a different direction, with a lower energy (higher wavelength) than before. Like in photo-electric absorption, this can only happen when the absorbed energy is higher than the binding energy, unless it is a free electron (not bound to an atom).
The probability q_C of Compton scattering increases with the Z-value of the material. It first rises with energy and then drops, according to a function f, which can be calculated with the Klein-Nishina approximation [42, 43].

$$q_C \sim Z f(E) \tag{2.5}$$

Rayleigh scattering or coherent scattering is scattering not on a single electron, but on the cloud of electrons as a whole. No electron gets kicked out in this interaction and there is no energy transfer. The X-ray photon continues its path in a different direction.
The probability for Rayleigh scattering q_R decreases with increasing energy and increases with increasing Z.

$$q_R \sim \frac{Z^n}{E^n} \tag{2.6}$$

with n a number between 2 and 2.5 [44].
Pair production occurs when a photon, under the influence of the electromagnetic field of the nucleus or the electrons, gets transformed into a pair of a particle and an antiparticle, usually an electron and a positron. This can only happen when the energy of the photon is higher than $2 \times 0.511\,\text{MeV} = 1.022\,\text{MeV}$, the rest energy of an electron and a positron combined. Any remaining photon energy above this threshold will be the kinetic energy of the resulting pair. Since this is much higher than the used energies in μCT, pair production does not contribute to the attenuation coefficients.
All the above formulas are approximations which are only valid in μCT X-ray energy ranges.

2.3.2 X-ray set-ups

The basic orientation of the components of a CT scanner is as follows:

- The X-ray source has the window where X-rays emerge pointed to the detector.

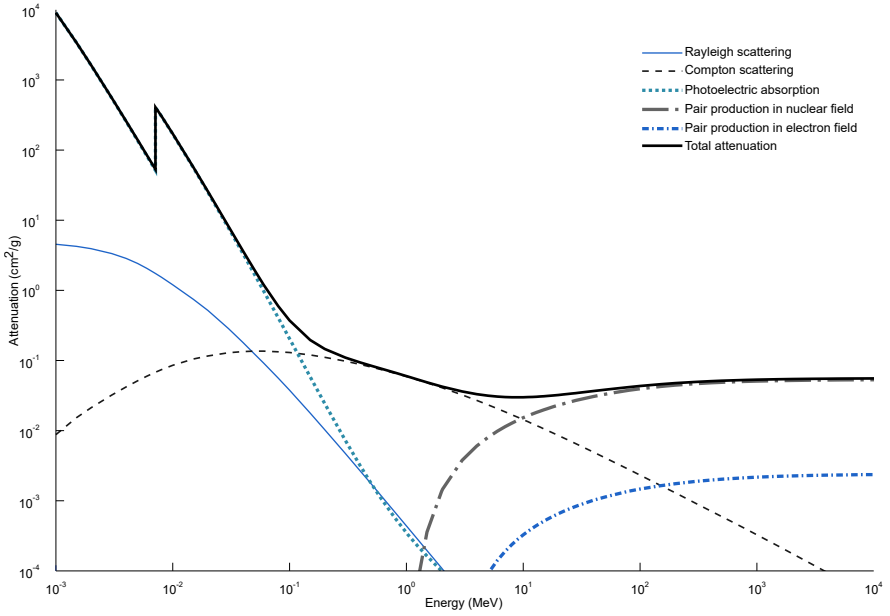


Figure 2.6: Different contributions to the mass attenuation coefficient μ/ρ of iron. Adapted from [45].

- The X-ray detector has the side sensitive to X-rays pointed to the source.
- The sample is in between source and detector, in the path of the X-rays.

There are two options for the rotation: either the sample is placed on a rotation stage and rotates around its own axis (as in the first three sub-figures of figure 2.7) or the X-ray source and detector rotate simultaneously around the sample (as in the last sub-figure of figure 2.7). In μCT, it is most often the sample that rotates. In medical CT the patient remains stationary while source and detector rotate. For the scans in this thesis, which are dynamic scans, it will be the gantry (i.e. source and detector) that rotates, in contrast to most μCT systems. This is because dynamic processes often require tubes or cables to supply for example water or electricity to the sample or sample holder. These may not be tangled up as a result of the rotation, especially since there will preferably be multiple, continuous rotations (or, equivalent, one continuous rotation spanning a large rotation angle, multiple times 360°) instead of a single full rotation.
The X-ray beam may have different shapes, depending on the source. Figure 2.7 depicts some of these. A fan beam geometry has a point source and

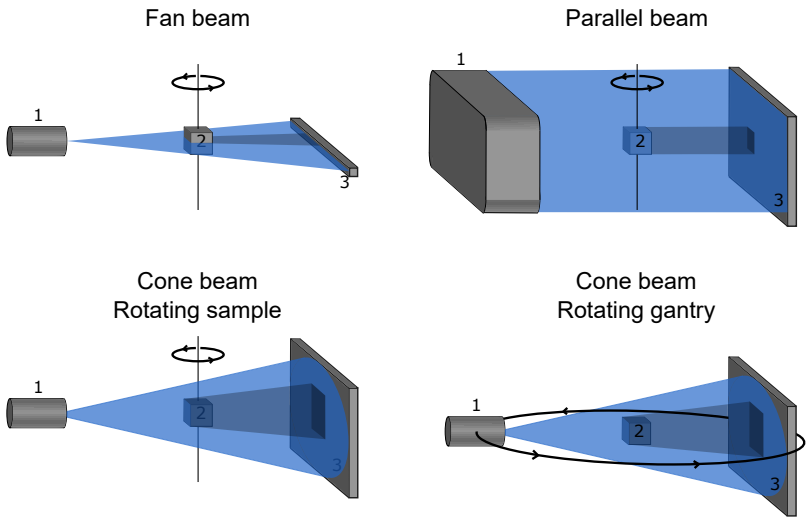


Figure 2.7: Four possible geometries for the X-ray CT set-up. Note that other geometries are also possible, for example parallel beam with a line detector (as in the shown fan beam geometry) or a rotating gantry in other geometries than cone beam. The indicated components are 1. the X-ray tube. 2. a sample. 3. the detector.

a line detector. A fan beam will need an additional movement (besides the rotation) to scan the entire sample, since the beam does not pass through the whole sample at once. In figure 2.7, this would be a vertical translation. A geometry with a 2D detector records the entire radiograph at once and therefore does not require this additional translation. This is the case for the depicted parallel and cone beam in figure 2.7. Most current lab-based CT scanners have a cone beam set-up, while synchrotron beamlines often use a 2D parallel beam set-up. Both geometries allow for the scans to be performed much faster, since a translation is no longer necessary.
Note that while the cone-beam in figure 2.7 fits completely on the detector, this is not a real occurring situation. Usually the detector is completely encompassed in the X-ray beam, so that all its pixels measure something, even the ones in the corners. The shadow image of the sample, on the other hand, should fit completely on the detector.
In this work, I will mostly use the cone-beam stationary-sample set-up (bottom right in figure 2.7). This is the set-up for EMCT [46] (see section 2.7), the UGCT scanner used for dynamic samples.

2.3.3 X-ray source

X-rays may originate from different sources. Natural X-ray sources are for example the sun, planets, comets and black hole binaries [47, 48]. Radioactive substances may, after decay, remain in an excited state and relaxation can then cause γ-rays to be emitted. This is, however, seldom used as an X-ray source in μCT for practical reasons [49]. In this work, an X-ray source will always refer to a man-made device which produces X-rays with the goal of using these in computed tomography. X-ray sources can be classified in two broad categories:

Synchrotron sources [50, 51, 52, 53] are circular particle accelerators that produce X-rays at a few different points. First, at the bending points, where the particle trajectories are curved by bending magnets to create a closed circular path. Secondly, at wigglers, an insertion device that is a line-up of a series of bending magnets. Thirdly, at undulators, another type of insertion device which produces the most intense beam of these three. The X-ray beam at one such bending point or insertion device is enclosed in a beam line and typical synchrotrons have multiple beam lines where X-ray experiments can be performed. An X-ray beam from a synchrotron is typically coherent and nearly monochromatic after adding a monochromator, both very desirable properties. The monochromator is possible while maintaining sufficient intensity of the beam because the beam intensity is incredibly high. This high intensity allows for much faster imaging with lower noise. Synchrotrons are therefore ideal X-ray sources for computed tomography. However, there is only limited access to beam lines, since building a synchrotron is costly and requires space. Requesting beam time (time at a beam line to perform your experiments) generally takes months and does not have guaranteed success.

Lab-based X-ray sources or X-ray tubes have less intensity, are not focussed and are typically polychromatic. However, they are cheap (affordable for a research lab or company), they fit into an ordinary room and are readily available.

There exist many types of X-ray tubes, but they are all based on the same

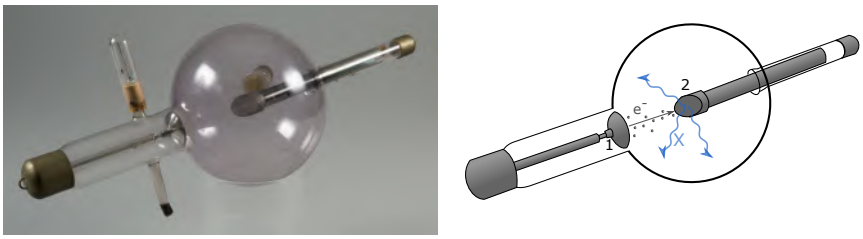


Figure 2.8: Photograph (left, [54]) and schematic drawing (right) of a Coolidge tube. On the drawing, 1 is the cathode and 2 is the anode.

principle. In figure 2.8 this principle is illustrated for a Coolidge tube, which is an old type of X-ray tube without additional housing besides the glass exterior, making it ideal to see the components inside. Electrons are emitted from a filament, the cathode. An electric field accelerates them towards a target material, the anode. In the anode, they bump into the atoms and decelerate. During this, they emit bremsstrahlung. The electrons may also excite electrons in the target material. These atoms will then emit relaxation radiation. These two effects together are the source of the X-rays and form the X-ray spectrum of that particular X-ray source. Figure 2.9 shows an example of an X-ray tube spectrum. The continuous curve is the bremsstrahlung, the two peaks are the characteristic K lines for rhodium (the relaxation radiation). The bremsstrahlung is cupped at low energies due to filtering in the air and in the tube itself.

The fact that this spectrum is not just an ultra-small peak, i.e. that it

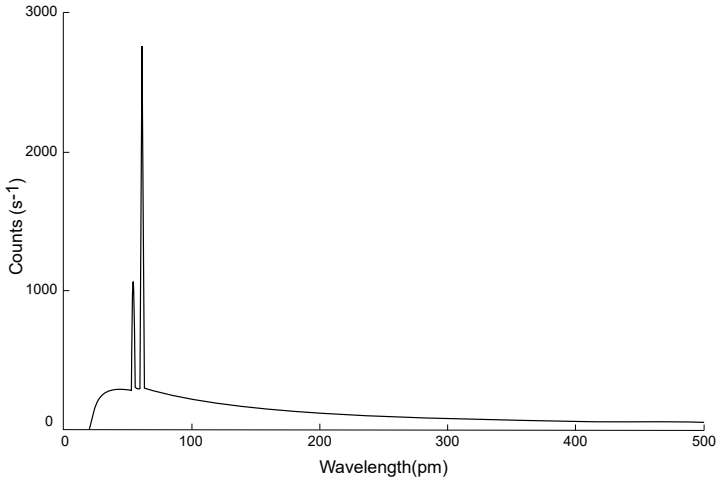


Figure 2.9: Spectrum of an X-ray tube with a rhodium target, operated at 60 kV. Adapted from [55].

is polychromatic, poses some problems in laboratory based X-ray CT. The major consequence is called beam hardening [56], which is discussed in more detail in section 2.5.2.

For the interested reader, both Boone [40] and Behling [57] provide a wide discussion on X-ray tubes.

2.3.4 X-ray detector

An X-ray detector detects the intensity of incoming X-rays on a pixelated matrix and digitizes this. The signals are sent to a computer and further processed there.
Two kinds of detectors exist: direct and indirect ones [58, 40]. In a direct detector, X-rays interact with a layer of the detector and create electron-hole pairs. These represent a charge, which gets affected by the electric field. The pair moves to the anode and cathode and this causes an electric signal. Indirect detectors have the X-rays interact with a scintillation layer. This layer emits visible light in response to these X-rays. A second layer transfers this light into an electric signal.
Direct detectors have the advantage that their resolution is better and they can count individual incoming photons, but in practice most detectors in μCT are indirect ones. This is because these detectors are easier to make, i.e. more commercially available (the light-to-electric layer has a bigger development industry behind it, since this can also be used in photography), are cheaper, have a higher flexibility (the scintillator can be exchanged) and they can be optimized for the best suited energy range. Direct detectors, on the other hand, depend on a particular semiconductor, mostly silicon (Si), which has a low efficiency at higher X-ray energies. Other possiblities are cadmium-telluride (CdTe) or cadmium-zink-telluride (CdZnTe) [59], of which there are only a few suppliers worldwide and chromium compensated galium arsenide (GaAs) [60, 59], of which supply is very limited.
A detector typically has different sensitivities to different energies of X-rays. Combined with the X-ray spectrum of the source, this determines how much each X-ray energy contributes to the radiography.
Boone [40] and Lowe and Sareen [61] discuss the different aspects of an X-ray detector.

2.3.5 Normalisation

A radiograph is a 2D collection of pixels, each measuring the attenuation along the path from the source to that particular detector pixel. This happens according to the Lambert-Beer equation [36, 37, 38], shown in equation 2.3 and repeated here:

$$I_{theory} = \int_{\mathcal{E}} I_0(E) \cdot \exp\left(- \int_{\mathcal{L}} \frac{\mu}{\rho}(s, E)\rho(s)ds \right) dE \tag{2.7}$$

I_{theory} is the intensity as it should be observed on the detector pixel, E is the energy of a photon, $I_0(E)$ is the intensity of the X-ray beam when it comes out of the source for a particular energy E, $\mathcal{E}$ is the energy spectrum of the ray, $\mathcal{L}$ is the path the ray followed from source to detector, s is a fraction

of this path, i.e. a location coordinate in the sample, $\rho(s)$ is the density of the sample at location s and $\frac{\mu}{\rho}(s, E)$ is the mass attenuation coefficient (described in section 2.3.1) of the sample at location s and energy E.
However, the measured $I_{measured}$ is not equal to I_{theory}. There are unwanted effects that can not easily be resolved (noise and artefacts), but some effects *can* be resolved. For this, CT scans use normalisation. At least two additional radiographs are taken, besides the 'real' ones described before: the dark field (DF) and the flat field/intensity offset (IO). The dark field is a radiograph taken when the source is turned off. This measures the signal the detector measures even when there is nothing to be measured - the detector background. The intensity offset measures the signal when the sample is not positioned on the scanner - this should measure $I_0 = \int_{\mathcal{E}} I_0(E)dE$ directly (hence the name IO), including the attenuation through air which we consider to be 0 in the reconstruction algorithms. By including the air attenuation in I_{IO} we have already incorporated it and can indeed ignore it in the reconstruction. Note that the attenuation through air is not exactly the same in I_{IO} and $I_{measured}$, since the IO also measures transmission through a part air that will have the sample in it in the actual measurement. This effect is small and can be neglected.
The normalised intensity is [62, 40, 58]:

$$I_{norm} = \frac{I_{measured} - I_{DF}}{I_{IO} - I_{DF}} \tag{2.8}$$

And this results in:

$$I_{norm} \approx \frac{I_{theory}}{I_0} \tag{2.9}$$

Often, more than one dark field and flat field are taken, typically using frame averaging. This can provide improvement, i.e. eliminate more Poisson noise [40].
If we now define $I/I_0 = I_{norm}$ the Lambert-Beer equation becomes:

$$I_{norm} = \frac{I}{I_0} = \exp\left(-\int_{\mathcal{E}}\int_{\mathcal{L}} \frac{\mu}{\rho}(s, E)\rho(s)dsdE\right) \tag{2.10}$$

Usually, this is simplified by assuming a monochromatic beam:

$$I_{norm} = \frac{I}{I_0} = \exp\left(-\int_{\mathcal{L}} \frac{\mu}{\rho}(s)\rho(s)ds\right) \tag{2.11}$$

A monochromatic beam means that each position in the sample only has one attenuation coefficient $\frac{\mu}{\rho}$, a number, instead of a $\frac{\mu}{\rho}$ that is a function of energy. Therefore the sample volume will be easier to reconstruct. This faulty assumption introduces beam hardening artefacts (section 2.5.2), but these are outside of the scope of this work. Therefore, we will assume this equation is adequate.

2.4 Reconstruction: how it becomes 3D

After performing a CT scan as described in section 2.3, the result is a series of radiographs or *projections* that will be used to obtain the 3D image representing the internal and external morphology of the scanned object.
The starting point are detector pixels i, each measuring an intensity I_i as in equation 2.11. The wanted end result is the distribution of attenuation coefficients through the sample μ. Because a continuous μ would actually mean infinite data points, which a computer can not store, and the starting data is discrete anyway, the end result will be a 3D grid of voxels. A voxel is the 3D equivalent of a pixel. In general in μCT and more particular in this work, a voxel is isotropic, i.e. a cube. This means we will have to discretise the Lambert-Beer equation, equation 2.11.

To change the continuous equation into a discrete one, the integral will

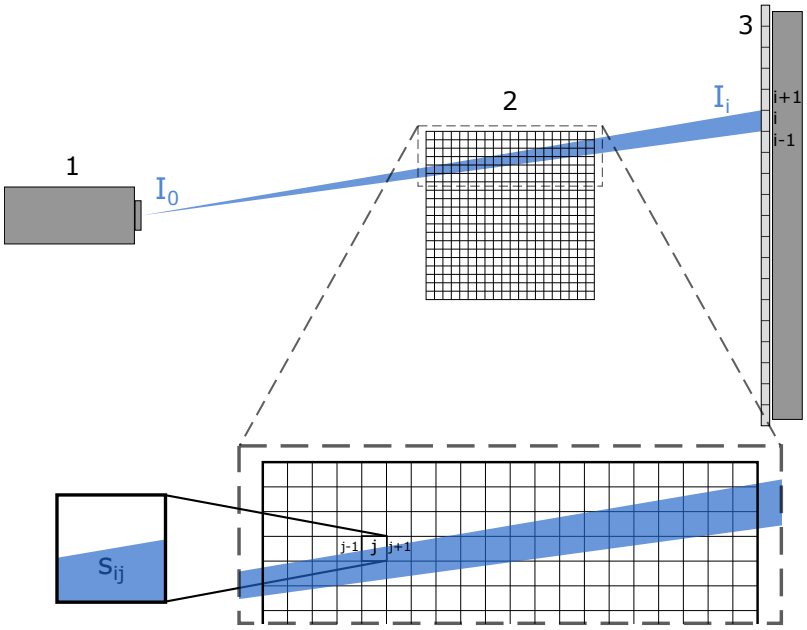


Figure 2.10: A representation of an X-ray (blue) coming from a source (1), passing through a volume of voxels (2) and arriving at a pixel of the detector (3). This is a 2D slice through the scanning situation.

become a sum.

$$p_i = -\ln\left(\frac{I_i}{I_0}\right) = \int_{\mathcal{L}_i} \mu(s)ds \tag{2.12}$$

$$\Rightarrow p_i = -\ln\left(\frac{I_i}{I_0}\right) = \sum_j \mu_j s_{ij} \tag{2.13}$$

i is the index of a detector pixel, while j is the index of a voxel in the volume. p is a projection value, which is the logarithm of a normalised intensity. s_{ij} represents the volume of voxel j that was passed through by ray i, normalised by dividing by the area of detector pixel i. This means s_{ij} has the unit of a distance and can also be considered the traversed distance through a voxel for an infinitely small X-ray beam. For most of the voxels (all the ones not on the i'th X-ray path), $s_{ij} = 0$. Figure 2.10 shows the path of one X-ray (defined as all the photons arriving at the same detector pixel) through the sample (volume grid of voxels), as well as the relevant symbols from the equations. This figure shows the 2D situation for ease of display. The 3D situation is analogous.
The reconstruction starts from all the p_i and wants to find all the μ_j by using equation 2.13. Two main categories of reconstruction algorithms exist: analytical and algebraic/iterative. Analytical reconstruction, section 2.4.1, is more mathematical and is less computationally complex. Algebraic reconstruction, section 2.4.2, is more intuitive and can be adapted more easily, but typically requires more computer time to complete due to its complexity.

2.4.1 Analytical reconstruction

Filtered back projection (parallel beam)

The most used analytical reconstruction technique is filtered back projection [63, 62]. It is most intuitive for a parallel beam geometry (figure 2.7), but can be adapted for other geometries.

In a parallel beam geometry, each X-ray is parallel to the others within the same projection (radiography). Consider a single horizontal line in a projection, corresponding with a single slice through the volume. The volume is built up of such slices, parallel to the xy-plane. Each slice can then be reconstructed separately. The algorithm will be explained for one slice, making this 2D instead of 3D. The extension to 3D is simply to stack the 2D reconstructions together. This simplicity to move to three-dimensionality is one of the big computational advantages of a parallel beam geometry, i.e. of synchrotron radiation μCT, compared to cone beam, where there is mix-up (see figure 2.10).
The basis of filtered back projection uses the Fourier slice theorem [63, 62], which we will first cite and then derive:

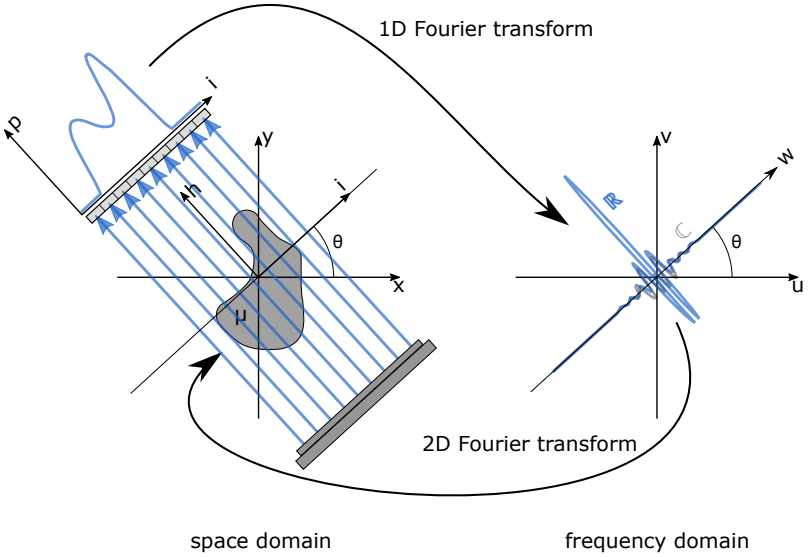


Figure 2.11: A representation of a parallel beam (blue) passing through a 2D sample (or one slice of a 3D sample) and arriving at the detector. The sample and the detector response can be expressed as functions, respectively μ and p. To the right is their Fourier transform (which has a real $\mathbb{R}$ and an imaginary $\mathbb{C}$ component). Remade and extended from figure in Pan *et al.* [64].

> The 1D Fourier transform of a parallel projection p_i of a 2D object function $\mu(x, y)$ at an angle θ with respect to the X-axis, gives a slice of the 2D Fourier transform $S(u, v)$ of the function $\mu(x, y)$ at an angle θ with respect to the u-axis.

or as mathematical formulas:

$$S(\theta, w) = \int_{-\infty}^{\infty} p(\theta, i) e^{-\imath 2\pi w i} di \tag{2.14}$$

$$\mu(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(u, v) e^{\imath 2\pi (ux+vy)} du dv \tag{2.15}$$

$$\mu(x, y) = \int_{0}^{\pi} \left(\int_{-\infty}^{\infty} S(\theta, w) |w| e^{\imath 2\pi w i} dw \right) d\theta \tag{2.16}$$

In the previous equations, $\imath = \sqrt{-1}$ is the complex number, usually indicated by $\imath$ or i (in scientific work) or j (in engineering work). In this work, i is the coordinate along the detector. Mind the difference between i and $\imath$.

Figure 2.11 shows the used symbols.
For the derivation of these equations, we go back to the definition of the projection p_i, equation 2.12, which we can rewrite as:

$$p(\theta, i) = \int_{-\infty}^{\infty} \mu(h, i) dh \tag{2.17}$$

In a parallel beam geometry, $\mathcal{L}_i$ (from equation 2.12) never runs along the z-direction, only in the xy-plane. If we define h the direction along $\mathcal{L}_i$ and i perpendicular to this, the right-handed coordinate system ihz is just the fixed coordinate system xyz rotated over an angle θ, i.e. it is the coordinate system moving along with the gantry or sample. The transformations between these coordinate systems are:

$$x = i\cos(\theta) - h\sin(\theta) \tag{2.18}$$
$$y = i\sin(\theta) + h\cos(\theta) \tag{2.19}$$
$$i = x\cos(\theta) + y\sin(\theta) \tag{2.20}$$
$$h = -x\sin(\theta) + y\cos(\theta) \tag{2.21}$$

Let us now take the 1D Fourier transform of $p(\theta, i)$:

$$S(\theta, w) = \int_{-\infty}^{\infty} p(\theta, i) e^{-\imath 2\pi w i} di \tag{2.22}$$

Substitution of equation 2.17 into equation 2.22 gives:

$$S(\theta, w) = \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} \mu(h, i) dh \right) e^{-\imath 2\pi w i} di \tag{2.23}$$

This is in the moving coordinate system ihz. Reverting to the fixed coordinate system xyz results in:

$$S(\theta, w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y) e^{-\imath 2\pi w (x\cos(\theta) + y\sin(\theta))} dx dy \tag{2.24}$$

A 2D Fourier transform of a function $f(x, y)$ is by definition:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-\imath 2\pi (ux + vy)} dx dy \tag{2.25}$$

We recognise equation 2.24 as a two dimensional fourier transform with $f(x, y) = \mu(x, y)$, $u = w\cos(\theta)$ and $v = w\sin(\theta)$, so $S(\theta, w) = F(w\cos\theta, w\sin\theta)$. Therefore, a reverse 2D Fourier transform will yield $\mu(x, y)$, i.e. the wanted

attenuation coefficients. A reverse 2D Fourier transform is:

$$f(x,y) = \mu(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{\imath 2\pi(ux+vy)}dudv \tag{2.26}$$

Transforming to the polar coordinate system (w,θ) in the frequency domain ((u,v) is the rectangular coordinate system in the frequency domain) happens through the following transformations:

$$u = w\cos(\theta) \tag{2.27}$$
$$v = w\sin(\theta) \tag{2.28}$$

While the differentials of the double integral will change according to:

$$dudv = wdwd\theta \tag{2.29}$$

The reverse 2D Fourier transform becomes:

$$\mu(x,y) = \int_0^{2\pi}\int_0^{\infty} S(\theta,w)e^{\imath 2\pi w(\cos(\theta)x+\sin(\theta)y)}wdwd\theta \tag{2.30}$$
$$= \int_0^{\pi}\int_0^{\infty} S(\theta,w)e^{\imath 2\pi w(\cos(\theta)x+\sin(\theta)y)}wdwd\theta \tag{2.31}$$
$$+ \int_0^{\pi}\int_0^{\infty} S(\theta+\pi,w)e^{\imath 2\pi w(\cos(\theta+\pi)x+\sin(\theta+\pi)y)}wdwd\theta \tag{2.32}$$

The integral in the last equation was split into a part θ from 0 to π and a part from π to 2π. Knowing that $S(w,\theta+\pi) = S(-w,\theta)$, $\cos(\theta+\pi) = -\cos(\theta)$ and $\sin(\theta+\pi) = -\sin(\theta)$, the last integral can be adapted to integrate over $w' = -w$:

$$\int_0^{\pi}\int_0^{\infty} S(\theta+\pi,w)e^{\imath 2\pi w(\cos(\theta+\pi)x+\sin(\theta+\pi)y)}wdwd\theta \tag{2.33}$$
$$= \int_0^{\pi}\int_0^{-\infty} S(\theta,w')e^{\imath 2\pi(-w')(-\cos(\theta)x-\sin(\theta)y)}(-w')(-dw')d\theta \tag{2.34}$$
$$= \int_0^{\pi}\int_{-\infty}^{0} S(\theta,w')e^{\imath 2\pi w'(\cos(\theta)x+\sin(\theta)y)}(-w')dw'd\theta \tag{2.35}$$

In the previous, notice that we swapped the integral bounds in the last step, causing an extra minus sign. We also know that $-w' = |w'|$ in this integral, since w' is always negative, going from $-\infty$ to 0. Now we can just rename w' to w and merge the two integrals that are summed in equation 2.32:

$$\mu(x,y) = \int_0^{\pi}\int_{-\infty}^{\infty} S(w,\theta)e^{\imath 2\pi w(\cos(\theta)x+\sin(\theta)y)}|w|dwd\theta \tag{2.36}$$

Since $\cos(\theta)x + \sin(\theta)y = i$, this becomes:

$$\mu(x,y) = \int_0^{\pi} \int_{-\infty}^{\infty} S(\theta,w) e^{\imath 2\pi w i} |w| dw d\theta \tag{2.37}$$

With this, we derived all three equations 2.14, 2.15 and 2.16.
To obtain the unknown $\mu(x,y)$ from the known $p(\theta,i)$ it is necessary to first perform a 1D Fourier transform (equation 2.14) to $S(\theta,w)$, then filter the result with $|w|$ and end with an inverse 2D Fourier transform (equation 2.16) to result in $\mu(x,y)$. Figure 2.11 illustrates the Fourier slice theorem for continuous functions p and μ.
The convolution theorem allows us to rewrite equation 2.37 or equation 2.16 in spatial coordinates. The theorem states that $\mathcal{F}(f(x)*g(x)) = \mathcal{F}(f(x))\mathcal{F}(g(x))$ with $\mathcal{F}$ the Fourier transform. With $f(x)$ representing $p(\theta,i)$, therefore $\mathcal{F}(f(x)) = S(\theta,w)$, and $\mathcal{F}(g(x))$ being $|w|$, we can rewrite equation 2.37 to:

$$\begin{aligned}
\mu(x,y) &= \int_0^{\pi} \int_{-\infty}^{\infty} \mathcal{F}(p(\theta,i))\mathcal{F}(g(x)) e^{\imath 2\pi w i} dw d\theta && (2.38)\\
&= \int_0^{\pi} \int_{-\infty}^{\infty} \mathcal{F}(p(\theta,i) * g(x)) e^{\imath 2\pi w i} dw d\theta && (2.39)\\
&= \int_0^{\pi} p(\theta,i) * g(x) d\theta && (2.40)
\end{aligned}$$

However, the inverse Fourier transform of $|w|$, i.e. $g(x)$, does not exist because this integral does not converge. This will be solved by going from the continuous to the discrete regime as explained below.
In practise, both the input data p and the output data μ will be discrete. The integrals in equations 2.14 and 2.16 will become sums. K is the number of radiographs taken, each at an angle θ_k. N is the number of detector pixels, each having a coordinate $i_n = n\Delta_i$. Δ_i is the distance between two detector pixels, also known as the detector pitch. Theoretically, the frequency domain of the Fourier transform extends to infinity in both directions: $-\infty \leftarrow w \rightarrow \infty$. However, for practical purposes it is safe to assume w to be limited to $-\frac{1}{2\Delta_i} < w < \frac{1}{2\Delta_i}$, since the Fourier components outside of this range will be negligible. These limits are the Nyquist frequency [65]. The Fourier transform will be sampled at frequencies $w_m = m\frac{1}{N\Delta_i}$, each being $\frac{1}{N\Delta_i}$ apart.

Equations 2.14, 2.15 and 2.16 become in discrete form:

$$S_{\theta k}(w_m) = \Delta h \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} p_{\theta k}(i_n) e^{-\imath 2\pi w_m i_n} \tag{2.41}$$

$$\mu(x,y) = \frac{\pi}{K} \sum_{k=1}^{K} \left(\frac{1}{N\Delta_i} \sum_{m=-\frac{N-1}{2}}^{\frac{N-1}{2}} S_{\theta_k}(w_m)|w_m| e^{\imath 2\pi w_m i_n} \right) \tag{2.42}$$

Equations 2.41 and 2.42 are an implementation of the filtered back projection algorithm for parallel beam. This consists of three steps:

1. Calculate the discrete Fourier transform of the projection data (equation 2.41)
2. Filter the result with $|w|$ and calculate the inverse Fourier transform (inner sum in equation 2.42)
3. Add the results of all the different angles θ_k (outer sum in equation 2.42)

It is also possible to use equation 2.40. When staying within the Nyquist limits, $g(x) = \mathcal{F}^{-1}(|w|)$ becomes an integral that converges:

$$g(x) = \int_{-\frac{1}{2\Delta_i}}^{\frac{1}{2\Delta_i}} |w| e^{\imath 1\pi w x} dw \tag{2.43}$$

The solution is:

$$g(x) = \frac{1}{4\Delta_i} \qquad \text{for } x = 0 \tag{2.44}$$

$$= 0 \qquad \text{for } x \text{ even and not } 0 \tag{2.45}$$

$$= -\frac{1}{\Pi^2 \Delta_i x^2} \qquad \text{for } x \text{ odd} \tag{2.46}$$

This theoretically correct filter is called the bandlimiting filter. Since the highest frequencies are amplified in this filter while they contain mostly noise, other filters often take its place. Two well known ones are the Shepp-Logan filter and the Cosine filter:

$$g(x)_{shepp-logan} = \frac{2}{\pi^2 \Delta_i (1 - 4x^2)} \tag{2.47}$$

$$g(x)_{cosine} = \frac{2}{\pi^2 \Delta_i} \left(sign(x) \frac{x}{2} - \frac{1 + 4x^2}{1 - 4x^2} \right) \frac{1}{1 - 4x^2} \tag{2.48}$$

Using these filters in equation 2.40 (and discretising the integral in that equation) yields an implementation of the filtered back projection.
Since the discrete positions (x, y) of the result usually do not correspond exactly to the sampled positions (i, θ_k), the algorithms described above are often accompanied by some sort of interpolation, usually linear.

FDK algorithm (cone beam)

The standard algorithm for the in lab-based μCT more used geometry of a cone beam (figure 2.7) is called the FDK-algorithm after its inventors Feldkamp, Davis and Kress [66, 63]. This algorithm is only approximate, but the errors remain small for small cone beam angles and the implementation is computationally efficient.
The FDK algorithm is a 3D extension to the earlier discussed FBP algorithm. In 3D, the z-direction, usually parallel to the rotation axis, complements the spatial dimensions x and y. The detector is now 2D instead of 1D, so the i coordinate gets supplemented with r. The coordinate r is vertical, just as z. In other paragraphs, both i and r are depicted by the index i.
The equations become for a cone beam [58]:

$$\mu(x, y, z) = \int_0^{2\pi} \frac{d_{SOD}^2}{U(x, y, \theta)^2} Q_\theta(i, r) d\theta \tag{2.49}$$

$$Q_\theta(i, r) = \int_{-\infty}^{\infty} S_\theta(w, r)|w| e^{\imath 2\pi w i} dw \tag{2.50}$$

$$S_\theta(w, r) = \int_{-\infty}^{\infty} \frac{d_{SOD}}{\sqrt{d_{SOD}^2 + i^2 + r^2}} p_\theta(i, r) e^{-\imath 2\pi w i} di \tag{2.51}$$

In this, d_{SOD} is the source-object distance, the distance between the X-ray source and the centre of the sample object. $U(x, y, \theta)$ is the distance from the source to the point (x,y) depicted on the central ray, i.e. the line connecting the X-ray source to the origin (the centre of the sample). It can be expressed as:

$$U(x, y, \theta) = R + x \sin(\theta) - y \cos(\theta) \tag{2.52}$$

A more complete explanation of these analytical reconstruction algorithms can be found in for example De Witte [58], Pan *et al.* [64], Feldkamp *et al.* [66] and Dierick [62].

2.4.2 Iterative reconstruction

Iterative reconstruction starts from an initial (usually empty) volume and improves this volume little by little, until it represents the object as well as

possible.
The goal of any reconstruction is finding μ_j, the local attenuation coefficient in every voxel j. This means solving the Lambert-Beer equation 2.13, repeated here:

$$p_i = -\ln\left(\frac{I_i}{I_0}\right) = \sum_j \mu_j s_{ij} \tag{2.53}$$

This is in theory possible because p_i are the known projection values and s_{ij} could be calculated. However, in practice this set of equations (one for each projection value I_i) is quite big: the number of pixels in the detector times the number of radiographs taken. In addition, this set could be over- or underdetermined and there is noise present, meaning there might not even be a unique solution μ_j. Iterative algorithms will therefore attempt to approach the solution as well as possible.
The archetype of iterative reconstruction techniques is the Kaczmarz method [67], which is a method to solve a linear set of equations. The Kaczmarz

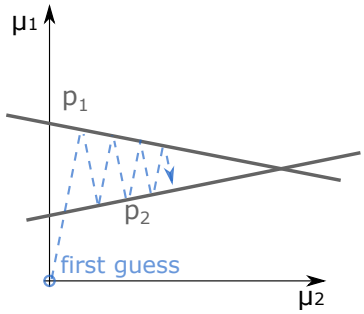


Figure 2.12: How the Kaczmarz method works in two dimensions, i.e. $M = 2$ and $N = 2$ (see text for the definition of M and N). The shown hyperplanes (lines in two dimensions) are $p_1 = s_{11}\mu_1 + s_{12}\mu_2$ and $p_2 = s_{21}\mu_1 + s_{22}\mu_2$. The first guess is in this image $(\mu_1, \mu_2) = (0, 0)$, but can be any point.

method provides a solution to the least squares problem. Figure 2.12 provides an illustration for two dimensions. The algorithm starts from an initial guess and projects this on the hyperplane defined by one of the equations. The result is the new current estimate and is projected onto the following hyperplane. This projection process, equation 2.57, is repeated until the current estimate converges to the solution of the set of equations.
Equation 2.13, the discrete Lambert-Beer equation as seen earlier, is an equation of a hyperplane in M-dimensional space (M being the amount of voxels in the volume, i.e. the amount of different j), determined by $(\mu_1, \mu_2, ..., \mu_M)$. There are N hyperplanes, the amount of different i. The solution is the point

where al these hyperplanes intersect. If this point does not exist due to noise, it is best to approximate it.
In vector notation the equations become for each i:

$$p_i = \vec{s}_i \cdot \vec{\mu} \tag{2.54}$$

$$\vec{s}_i = (s_{i1}, s_{i2}, ..., s_{iM}) \tag{2.55}$$

$$\vec{\mu} = (\mu_1, \mu_2, ..., \mu_M) \tag{2.56}$$

The second vector, $\vec{s}_i$ describes the intersection lengths of an X-ray i with any of the voxels j. Note that $s_{ij} = 0$ for most combinations of i and j. For example, in figure 2.10 a particular X-ray i is shown. Every voxel j that does not have part of this X-ray passing through its volume has $s_{ij} = 0$. The figure clearly shows this is the case for the majority of voxels j.
The starting point is a guess $(\mu_1^0, \mu_2^0, ..., \mu_M^0)$ which is projected perpendicular to the first hyperplane i. The i's can be put in any order. Mathematically this can be written as:

$$\vec{\mu}^k = \vec{\mu}^{k-1} + \frac{p_i - \vec{\mu}^{k-1} \cdot \vec{s}_i}{\vec{s}_i \cdot \vec{s}_i} \vec{s}_i \tag{2.57}$$

A derivation of this formula, the Kaczmarz update step, can for example be found in Kak and Slaney [63]. Different iterative reconstruction algorithms use slightly different versions of this update step.
We define *one iteration* as one sweep over every i, i.e. when equation 2.57 has been performed N times, the algorithm has done one iteration. This means k has two components: $k = k_{super} \cdot N + k_{sub}$. The iteration number k_{super} indicates at which iteration we are now, an iteration defined as just above. The other variable, k_{sub}, is a sub-iteration number that keeps count of how many i have been traversed in this k_{super}-iteration already and is a number between 0 and $N - 1$.
In the first step, $k = 1$, with $k_{super} = 0$ and $k_{sub} = 1$. The next step projects this new $\vec{\mu}$ to the next hyperplane i, while k_{sub} increases with 1. By continuing this way, $\vec{\mu}^k$ will converge to the wanted intersecting point. The convergence is faster when the sequential hyperplanes have a higher angular difference with respect to one another.
Note that $\vec{\mu}^{k-1} \cdot \vec{s}_i$ in the numerator of equation 2.57 can be considered the forward projection, as seen in figure 2.13.

Figure 2.13 shows a schematic overview of iterative reconstruction. The eventual μ_j^k should be a good representation of the scanned sample.
As mentioned earlier, an iterative reconstruction algorithm will start from a typically empty volume of $\mu_j = 0$. It will traverse every pixel of every radiograph, each i, and perform three main steps:

1. Project the current volume

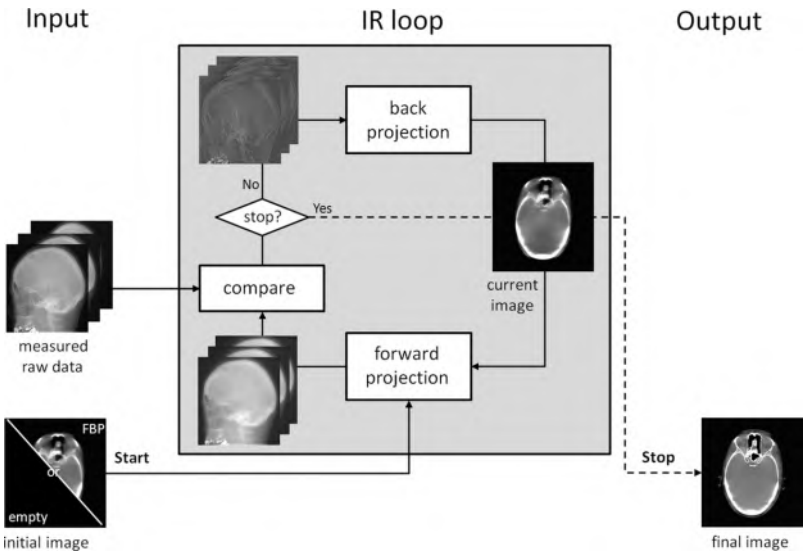


Figure 2.13: Schematic representation of an iterative reconstruction method. [68]

2. Calculate the difference between the projection and the measured radiograph

3. Back project this difference into the volume at the corresponding locations. This is the update step.

All radiographs can be traversed multiple times (i.e. multiple iterations). Step 1 simulates the projection of an X-ray to pixel i through the current estimation of μ_j^k, according to the Lambert-Beer equation. The simulated projection is q_i. This step is called the forward projection.

$$q_i = \sum_j \mu_j^k s_{ij} \tag{2.58}$$

Step 2 simply performs a subtraction between the just simulated q_i and the measured p_i. Finally, step 3 adapts the μ_j^k, considering the weight each μ_j^k contributed to this particular X-ray i:

$$\mu_j^k = \mu_j^{k-1} + \frac{p_i - q_i}{\sum_m s_{im}^2} s_{ij} \tag{2.59}$$

This equation is simply the application of equation 2.57 to X-ray CT. It is repeated for every i (in any chosen order), each time increasing k_{sub} and thus

k with 1.
The algorithm will keep iterating until it meets a certain criterium. This might be a chosen number of iterations, or it might be that the correction factor $|p_i - q_i|$ is smaller than some value for each i. It is wise not to use the latter criterium without also putting a restriction on the number of iterations, because noise can cause $|p_i - q_i|$ to never get smaller than the chosen value. In fact, the solution $\mu_j^{k_{\text{super}} \cdot N}$ might diverge again from the ideal solution after a few iterations if the noise is too high. This would cause an infinite loop.
The order in which the i are visited might influence the convergence rate. In the Kaczmarz method, the convergence is faster when sequential hyperplanes are (almost) orthogonal to each other. This means p_i^k and p_i^{k+1} should differ as much as possible, which is usually the case when looking at projections taken physically perpendicular to each other. There are algorithms, such as the weighted distance scheme [69], that help choose the best order. The worst order would be following the angular sequence of the scan, since each following projection will look similar to the previous one. When working with a lot of projections, an alternative is to just go through the projections in a random order, as this is faster than calculating the ideal and it is usually sufficiently good. The drawback is that the solution is not unique.
The s_{ij} determine how much voxel j contributes to X-ray i. Most voxels were not in the path of X-ray i, as seen in figure 2.10, and will have $s_{ij} = 0$. The other s_{ij} could have any value bigger than zero and can be calculated.
Since there are so many s_{ij} ($N \times M$), their calculation and storage is cumbersome. For example, when working with a tiny detector of 128×128 pixels, taking a really small amount of 128 projections, N is already 128^3. Reconstructing an equally small volume with this data, a cube with side 128 would make $M = 128^3$. This means there are 128^6 s_{ij} to be calculated and stored, which is about 4×10^{12} or 4 trillion, i.e. *a lot*, and this is for an unrealistically small μCT scan.
There are some approximations to ease the calculation of all these s_{ij}. The PhD thesis [58] lists a number of these. We will review here the two approximations used in this work, one for the forward projection (step 1) and one for the back projection (step 3).
For the forward projection, the approximation is ray-based or pixel-based. Figure 2.14 illustrates this for two instead of three dimensions. For calculating q_i, the algorithm takes equidistant steps of length Δl along the path $\mathcal{L}$ of X-ray i. At each visited point (x, y, z), it linearly interpolates the nearby voxels to get $\overline{\mu}(x, y, z)$. The result is multiplied with the length of one step Δl.

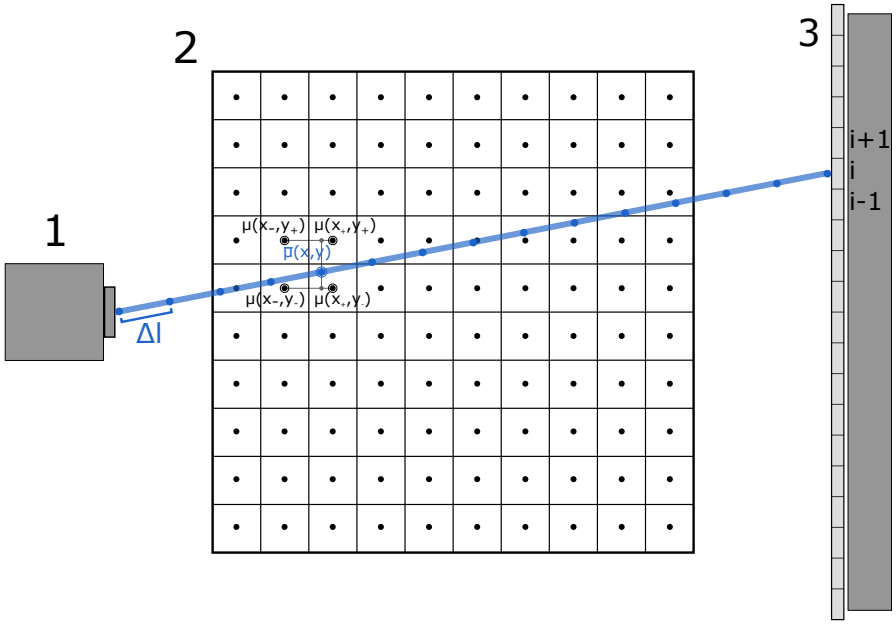


Figure 2.14: Schematic representation of the forward projection (equation 2.60) in the iterative reconstruction in two dimensions. 1 is the X-ray source. 2 is the volume grid. 3 is the detector.

$$q_i = \Delta l \sum_{\mathcal{L}_i} \overline{\mu}(x, y, z) \tag{2.60}$$

$$\overline{\mu}(x, y, z) = \frac{\sum_{x_j \in (x_-, x_+)} \sum_{y_j \in (y_-, y_+)} \sum_{z_j \in (z_-, z_+)} \mu(x_j, y_j, z_j)(x_{\bar{j}} - x)(y_{\bar{j}} - y)(z_{\bar{j}} - z)}{(x_+ - x_-)(y_+ - y_-)(z_+ - z_-)} \tag{2.61}$$

with $\sum_{\mathcal{L}_i}$ running over the path of X-ray i with steps of Δl, (x, y, z) a coordinate in the sample volume, x_- the x-coordinate of the voxels whose x-coordinate is the highest still lower than x, x_+ the x-coordinate of the voxels whose x-coordinate is the lowest still higher than x. When x_j is x_-, then $x_{\bar{j}}$ is x_+ and the other way around. Analogously for y and z.
For the back projection, the approximation is voxel-based. The fraction $\frac{s_{ij}}{\sum_m s_{im}^2}$ has two possible values in this approximation: 0 for the voxels not in the path of X-ray i and $1/L_i$ for the others, with L_i the total X-ray length through the sample. This assumes each s_{ij} is approximately equal to the

voxel size. The back projection step becomes:

$$\mu_j^k = \mu_j^{k-1} + \frac{p_i - q_i}{L_i} \tag{2.62}$$

Noise will negatively influence the results of the CT scan. To decrease its influence, there are two approaches inherent to iterative reconstruction: a relaxation factor or combining the back projection for multiple i.
A relaxation factor α is a number between 0 and 1 that decides how strong the back projection is:

$$\mu_j^k = \mu_j^{k-1} + \alpha \frac{p_i - q_i}{\sum_m s_{im}^2} s_{ij} \tag{2.63}$$

$$\approx \mu_j^{k-1} + \alpha \frac{p_i - q_i}{L_i} \tag{2.64}$$

It decreases noise, but slows down convergence.
Combining the back projection for multiple i means waiting to update μ_j^k. The algorithm calculates $p_i - q_i$ as usual, but instead of updating with equation 2.64, the next i is chosen and the next $p_i - q_i$ calculated and added to the previous. Only after doing this for several i, is the update step performed.
The difference with updating immediately is that the second, third,... correction factors are calculated with μ^{k-1} instead of with a new version of μ that already got the influence of the first correction factor. The back projection becomes:

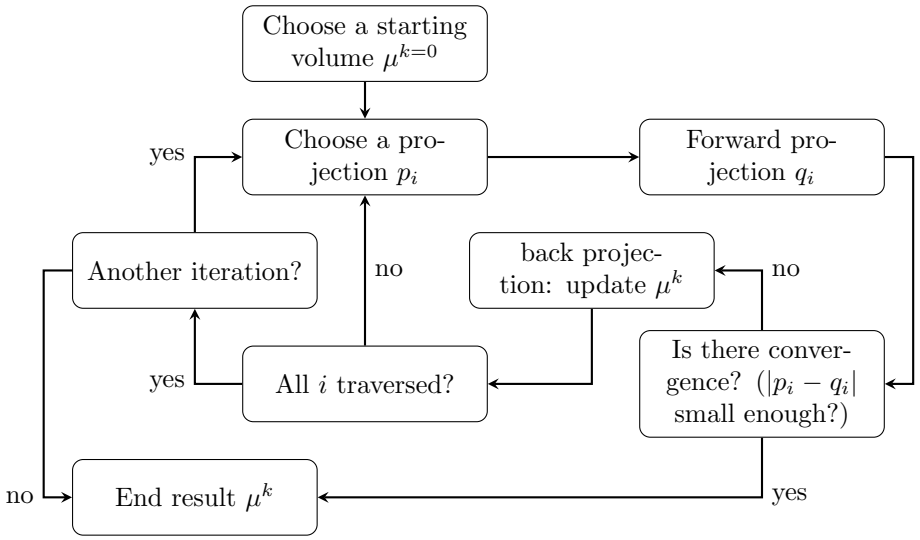
$$\mu_j^k = \mu_j^{k-1} + \alpha \sum_{i \in S} \left(\frac{p_i - q_i}{\sum_m s_{im}^2} s_{ij} \right) \tag{2.65}$$

$$\approx \mu_j^{k-1} + \alpha \sum_{i \in S} \left(\frac{p_i - q_i}{L_i} \right) \tag{2.66}$$

With S the set of i over which the correction factors are calculated before updating. This becomes equation 2.64 when S only contains one projection i.
The name of the iterative reconstruction algorithm changes according to the size of S:

- **ART (Algebraic reconstruction techniques)**: There are no sets, i.e. S has a size of 1.
- **SART (Simultaneous algebraic reconstruction technique)**: S contains all the pixels belonging to one radiograph.
- **SIRT (Simultaneous iterative reconstruction technique)**: S has a size of N and contains all i, i.e. all pixels of the detector and all radiographs.

Ordered subsets is the general term for SART, SIRT and every size of S in between. The bigger S is, the slower the convergence and the lower the noise. [4, 58]
Iterative reconstruction takes more computer time in comparison with analytical reconstruction (section 2.4.1). Parallel programming on the GPU (graphical processing unit) can partially compensate for this [4]. The work in this thesis uses GPU programming to speed up iterative reconstruction. Iterative reconstruction has the advantage over analytical that it allows to model physical effects and take into account these during the reconstruction [70]. An example is the polychromatic nature of the X-ray beam, which causes beam hardening, an unwanted reconstruction artefact (section 2.5.2).
Another possibility is to include prior knowledge about the sample into the algorithm. The simplest example is an initial volume. When you already have a volume that resembles the sample (for example from a previous scan), this can be used as μ_j^0 instead of 0. A previous scan could have been taken when the sample is undergoing a dynamic process. A more thorough explanation of this can be found in section 3.2.
In short, iterative reconstruction follows the following flowchart:



2.5 Artefacts

Imaging artefacts are unwanted effects in the final reconstructed volume, which cause a deviation from the perfect representation of the sample. There are several causes and therefore types of artefacts. This section presents a

list of the artefacts most important for this work.

2.5.1 Noise

Noise [56, 63, 71, 72] is a well known effect in almost all scientific data collection, including in μCT. Noise means that the radiograph pixels suffer from statistical deviation from the grey value they should theoretically have. Instead, they display a value lower or higher, with the additional difficulty that the size of this deviation is not fixed, but fluctuating. Noise is unavoidable, i.e. while it can be lowered, it can never be completely eliminated.
The amount of photons detected per detector pixel per time unit is subject to Poisson statistics. Therefore, the resulting measured intensity in a projection always contains some amount of noise. Since this is governed by Poisson statistics, more statistics (longer projection time or brighter source) mean a higher signal to noise ratio.
In a Poisson process, the chance P of k events happening is [73]:

$$P(k) = \frac{\lambda^k}{k!} e^{-\lambda} \tag{2.67}$$

with λ the expected number of events.
In the case of μCT an event is the detector measuring an X-ray photon. Since the amount of photons is directly proportional to the intensity of the X-ray beam, we can express P as a function of intensity instead of number of photons. Therefore, λ is I_{expect}, the expected intensity from the Lambert-Beer equation (equation 2.3) and k is $I_{measured}$, the measured intensity.
Starting from $k = 0$, $P(k)$ rises until it reaches a maximum at $k = \lambda$, after which it starts to drop again, asymptotically going to 0. For higher λ, this curve approaches a Gaussian curve, which follows the equation:

$$P(k) = \frac{e^{(k-\lambda)^2/(2\lambda)}}{\sqrt{2\pi\lambda}} \tag{2.68}$$

In this equation, the gaussian mean equals λ and the standard deviation is $\sqrt{\lambda}$.
The standard deviation of the Poisson peak or Gaussian peak, i.e. $\sqrt{\lambda}$ is an indication of the magnitude of the noise, i.e. how far $k = I_{measured}$ can deviate from $\lambda = I_{expect}$. Therefore, a higher intensity I_{expect} will increase the signal to noise ratio, which is $\frac{I_{expect}}{\sqrt{I_{expect}}} = \sqrt{I_{expect}}$. This can be achieved by increasing the flux of the source or increasing the time during which the detector is recording a radiograph. At the same time, samples that have a very high attenuation coefficient or thickness will attenuate most incoming photons, leaving few to reach the detector. Such a sample would result in a

low I_{expect} and thus in high noise. This situation is called photon starvation. Most notably metals, such as lead - typical materials to shield X-ray sources - are difficult to scan with μCT.
The important signal for reconstruction is not just I, but the normalised $\frac{I}{I_0}$. Both the numerator $I = I_{measured}$ and the denominator I_0, i.e. the flat field (see section 2.3.5), are measured projections subject to Poisson noise. If the normalisation of all I happens with the same I_0, the noise in this flat field becomes a fixed pattern through the dataset, leading to ring artefacts (ring shaped artefacts in the reconstructed images). To avoid this, many flat fields are recorded and averaged, increasing λ. Since the noise is Poisson, this will reduce the noise in the I_0. The same applies to dark field images, but their contribution to the noise in the final reconstruction is smaller compared to the flat fields. The PhD dissertation [40] presents a more thorough study of the noise in flat fields and filters specifically designed to reduce ring artefacts in chapter 3.
Due to the energy sensitivity of the detector, i.e. having different sensitivities to different energies, in combination with a polychromatic X-ray beam, the noise on the radiographs will not exactly follow Poisson statistics. However, it is a reasonable approximation. In addition, the exact relationship between noise on the radiographs and noise in the final reconstruction (i.e. noise propagation) is complicated and depends multiple factors, including the reconstruction algorithm [70].

2.5.2 Beam hardening

In short, beam hardening is the change in energy spectrum of an X-ray beam before and after traversing the sample [4, 70, 56, 71]. Since low-energy X-rays are typically attenuated more than higher energy photons, the final spectrum will have changed compared to the original spectrum: the amount of low energy X-rays will have decreased more than the amount of high energy X-rays. The mean of the outgoing spectrum thus shifts to higher energies. Most reconstruction algorithms do not take this into account, leading to an image artefact known as the beam hardening effect.
Therefore, the beam hardening effect is a consequence of the polychromatic nature of the X-ray beam, or more precisely: it is the consequence of the reconstruction algorithm not accurately taking into account this polychromaticity. The effect is not present when using monochromatic X-ray beams, as is often done at synchrotron beamlines.
The change in intensity of a polychromatic X-ray beam after passing through the sample is described by equation 2.3:

$$I = \int_{\mathcal{E}} I_0(E) \cdot \exp\left(- \int_{\mathcal{L}} \frac{\mu}{\rho}(s, E)\rho(s)ds \right) dE \tag{2.69}$$

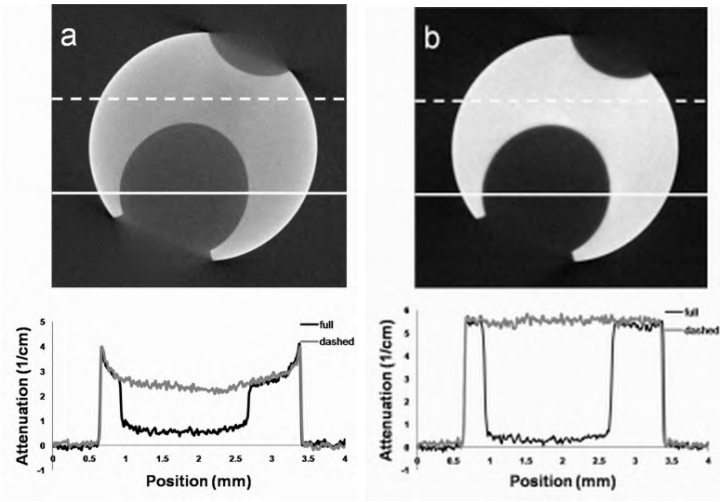


Figure 2.15: A slice through the reconstruction of a tooth implant and a plot of the grey value along two indicated lines. Left is a regular reconstruction, right is a reconstruction with beam hardening correction. [4]

in which $I = \int_{\mathcal{E}} I(E)dE$ because the detector only measures one intensity for all energy bins together, instead of one for each energy bin. This integration takes into account the initial energy spectrum and the different energy response the detector has for different photon energies.
The reconstruction algorithms, however, are based on the monochromatic Lambert-Beer equation:

$$I = I_0 \cdot \exp\left(-\int_{\mathcal{L}} \frac{\mu}{\rho}(s)\rho(s)ds\right) \tag{2.70}$$

in which we pretend none of the variables are dependent on the energy E.
In practice, this means the voxels near the centre of the sample will be reconstructed with a lower attenuation coefficient. The result is a gradient from centre to edges, even when it is a single material. Figure 2.15 shows an example, which includes the result of a beam hardening correction algorithm as described in [4].

2.5.3 Limited angle

Limited angle artefacts [74, 72] or missing wedge artefacts [75] appear in the reconstruction when the radiographs are taken from an insufficient angular range, which means there is not enough information to correctly reconstruct

the three dimensional volume. This can happen when the geometry of the scanner or the size and shape of the sample only allow a (very) limited rotation.
A reconstruction displays limited angle artefacts as prominent distortions in the image. Figure 2.16 shows an example of a limited angle reconstruction.

The Tuy-Smith condition [3, 76] states that an object's attenuation coef-

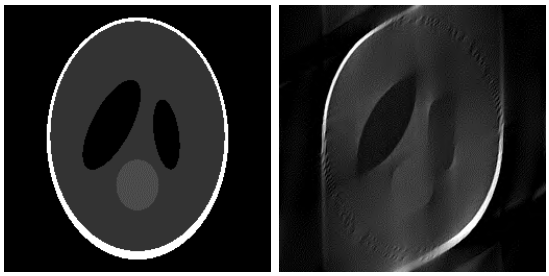


Figure 2.16: Left: a slice of the shepp-logan phantom. Right: the same slice of a reconstruction using only a 60° rotation of the (simulated) μCT scan. [72]

ficients can only be reconstructed exactly when each plane intersecting the sample contains at least one line from X-ray source to a detector pixel, somewhere along the rotation trajectory of the CT scan. It is clear that many of these planes are 'missing' their source point if the rotation was not sufficient. For example, a rotation around the z-axis over 60°, starting at the x-axis, will not be sufficient for the yz-plane. A larger rotation diminishes the limited angle artefacts.
A complete μCT scan contains a rotation over at least $180° + 2\alpha$ with α being half the opening angle of the cone. For a parallel beam geometry (see section 2.3.2), a rotation over 180° is sufficient, since each X-ray at a scanning angle θ has a 'mirror' X-ray at $\theta + 180°$ and the opening angle α is close to zero. On the other hand, for cone beam, $180° + 2\alpha$ is necessary [63]. To understand why, consider figure 2.17. The middle circle is the scanned area in which the sample resides. In the left figure, a rotation of 180° is performed, but some paths through the sample (planes when considering as their second axis the rotation axis) have not been traversed, such as the dashed line. For that, an additional rotation of 2α is necessary, as shown in the right figure.
If the rotation is performed over an angle smaller than $180° + 2\alpha$, limited angle artefacts occur.

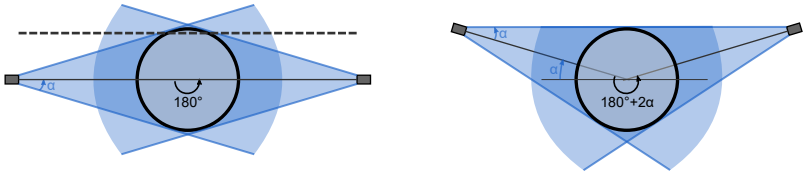


Figure 2.17: An illustration of the minimum required rotation for a cone beam geometry. More explanation on this figure can be found in section 2.5.3.

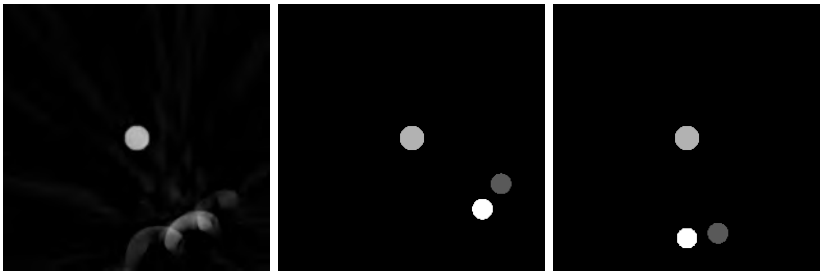


Figure 2.18: Left: slice of the reconstruction of a moving phantom. Middle: starting situation of the movement. Right: ending situation of the movement.

2.5.4 Motion artefacts

A CT scan requires a finite amount of time for the gathering of the projections. Even recording one radiograph takes a finite time, and this is multiplied with the number of projections for one scan, optionally adding the time required for rotation if the scanner does not record radiographs and rotate simultaneously. During this time, the sample may have moved unintentionally, or, when considering a dynamic process (as in this thesis), it may have changed according to this dynamic process. Since such a change is presumably continuous, no matter how small the time required for a CT scan, there will still be a change in the sample.
Motion or movement artefacts [56, 71, 72, 41, 77] occur when trying to reconstruct a three dimensional volume from the radiographs of such a changing sample (see section 2.6). 4D in this case means there was movement inside the sample during the scan, i.e. every radiograph was taken from a slightly different volume, with a bit of movement from the previous one. However, the reconstruction algorithm assumes every radiograph originated from the same, static volume.
The result is a smeared effect at the location of the movement. This blur is similar to photography of fast moving objects, when the object moves faster than the shutter time of the camera. Since the resolution in μCT can be

as good as the order of 1 µm, even really small changes or movements cause motion artefacts.
Motion artefacts can be caused by unwanted motion, for example a small tilting of the sample during the scan. However, of interest here are the motions that are part of a dynamic process that is the subject of the investigation, i.e. the reason the sample was scanned was to investigate this dynamic process. The scans are typically taken as fast as possible, to minimise these artefacts, while still slow enough to get a good signal to noise ratio. A fast rotation keeps the movement within one minimum angle of rotation (see section 2.5.3 for the minimum angle) to a minimum. However, since the changes in the sample are typically continuous, there is no finite time in which nothing in the sample changes. Therefore, the motion artefacts are still present even when scanning fast. They are inherent to the scanning of a dynamic process. The smeared effect of a motion artefact is the result of back projecting all the 'static' volumes that each radiograph imaged, i.e. the different time instances. The visual appearance of the artefact depends on the sample, the movement and even the trajectory of the scan. An example with a phantom of spheres is shown in figure 2.18. In this phantom, two spheres rotated clockwise around a static central sphere.

2.5.5 Undersampling

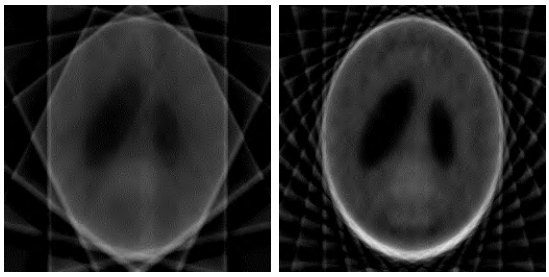


Figure 2.19: Slices of a reconstruction of a modified Shepp-Logan phantom. Left was with 8 projections, right with 31. About 250 projections are needed to avoid the undersampling effect here. [72]

Undersampling [56, 63, 71, 72] happens when a reconstruction is performed with an inadequate amount of radiographs (even though the radiographs can still span a whole rotation, see section 2.5.3) or an inadequate amount of pixels per radiograph. Sharp edges and small features of the sample will then cause a pattern of lines, as in figure 2.19.
The lines are caused by the backprojection (see section 2.4): a feature is backprojected along a line, both on its real location and other points in this

line. Since there are too few projections to correct these wrong parts, the lines remain in the reconstruction.
The outside edges of an object typically get more undersampling artefacts, since two subsequent projections (i.e. the rotation angle between them) cover a larger distance when considering a point further away from the rotation axis. To reconstruct the outside edges, the required number of projections according to the Nyquist theorem is $\frac{\pi}{2}N$, with N the number of detector pixels in one row [58].
Below this limit, a difference between step-and-shoot tomography and smooth rotation arises. In a step-and-shoot set-up, each radiograph is taken with a motionless scanner and the rotations are performed in between. In a smooth acquisition, the scanner rotates during the acquisition of radiographs. This last method causes angular smoothing [78], but is faster and therefore often more useful for 4D μCT. All set-ups of the UGCT use smooth acquisitions.

2.5.6 Partial volume effect

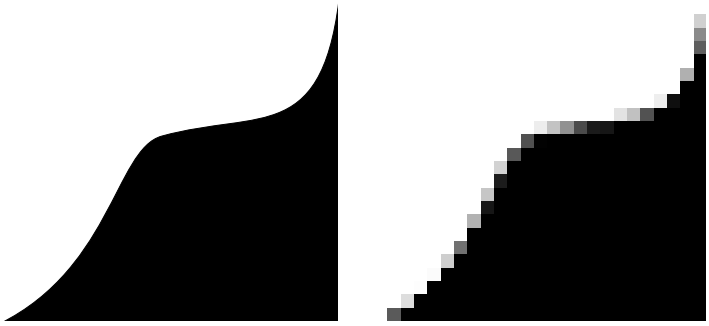


Figure 2.20: A theoretical example of the partial volume effect in two dimensions. Left: the sample. Right: a reconstruction suffering from the partial volume effect. We refer the reader to the online version to view the left figure as a vector image (i.e. without pixels).

The partial volume effect [79, 56, 71, 72] applies when more than one material occupies one voxel in the sample. This happens in the sample at the locations where two different materials meet, since these material interfaces are highly unlikely to coincide precisely with the independently chosen voxel boundaries. This is especially true if features in the sample may be smaller than the voxel size, i.e. the resolution of the CT scan is not sufficient to discern these features. It is clear that the partial volume effect at visible edges diminishes when the voxel size is reduced, which can only be done up until the resolution limit of the μCT scan. This limit depends the spot size (the

size of the X-ray source) and the resolving power of the detector, as explained in [40].
In such a voxel that contains more than one material, each with a different attenuation coefficient, the reconstructed voxel grey value will display a weighted average between these attenuation coefficients. Figure 2.20 displays an example: left is the perfect sample and right is the same sample with square voxels. The voxels on the edge between both materials display a grey value between white (the first material) and black (the second material), which will be darker or lighter depending on the fraction of each material in that particular voxel.
When the resolution is low or the features are smaller than the voxel size, this average attenuation coefficient in one voxel will represent the density ρ in that voxel, when combined with the mass attenuation coefficient of the material (see equation 2.3). So, counter-intuitively, the partial volume effect is in some cases less of a problem at really low resolutions compared to high ones. For example, a highly porous material with small pores will have pores (air bubbles) smaller than the voxel size. By measuring the density in such voxels, the porosity can be calculated, which is a property of interest.
Visually, the partial volume effect causes fuzzy edges. More importantly for data analysis, it can cause voxels to display grey values not present in the real sample. For example: if a sample consists of two materials, which have grey value 0 and 1, the voxels on the edge between these two materials might have grey value 0.25 or any other value between both materials. However, there is no material with grey value 0.25 in the real sample. This causes difficulties with segmentation and further analysis of the reconstructed sample.

2.5.7 Cone beam artefacts

Cone beam CT (see section 2.3.2) creates an additional kind of artefact, the cone beam artefact or the cone beam effect [56], which is not present in other geometries. This artefact can be explained by the Tuy-Smith condition [3, 76] (which is also mentioned in section 2.5.3). This condition requires that each plane intersecting the sample contains at least one line from X-ray source to a detector pixel, somewhere along the rotation trajectory of the CT scan. In a cone-beam geometry, the gantry (source and detector) rotate around the sample in one plane. When the rotation axis is the z-direction, this plane is the xy-plane with $z = 0$. Therefore, the xy-plane fulfils the Tuy-Smith condition, but any plane parallel to the xy-plane with $z \neq 0$ does not and can therefore not be reconstructed perfectly.
This artefact appears most at the top and bottom of the reconstructed sample (when rotation happened around the axis that connects this top and bottom)

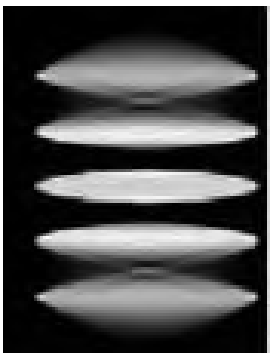


Figure 2.21: Example of a cone beam artefact. This is an xz-slice of the reconstruction of a phantom of five identical disks [80].

and can be quite severe.

2.6 What is dynamic micro computed tomography?

Dynamic micro computed tomography or 4D μCT is μCT with the addition of a fourth dimension: time. The sample being scanned can undergo a dynamic process which causes it to change during the scan. In this work, it is precisely this change, the dynamic process, we want to observe to be able to study it. Note that in some fields the dynamics are something to be avoided, for example the breathing motion in a medical chest CT scan. These fields are not considered in this thesis.
Since we want to observe the dynamic process itself, any destructive technique is not usable. This would disturb the very process we wish to see. Therefore, CT, being non-destructive, is a good candidate.
The end result of a 4D μCT scan should be a 4 dimensional volume, i.e. a time series of 3D reconstructions. The most simple way to get there is to divide the series of projections into a number of parts and perform a static 3D reconstruction (section 2.4) on each part. For example if there were N_{proj} projections and one 3D reconstruction required N_{rec} projections, it is possible to create $\frac{N_{proj}}{N_{rec}}$ parts back-to-back, or more if one projection is allowed to be used by multiple 3D reconstructions, i.e. if the parts may overlap.
The blurring effect in photographs when the subject is moving is widely known, for example when you try to photograph a moving car. This is because the detector pixels that detect the light and record it, are detecting for a certain time $\Delta t > 0$. During this time, the subject has moved. This causes one pixel to detect different parts of the subject. It can not know it has measured multiple things instead of one. This causes the blurring effect, also described in section 2.5.4.
What happens in 4D μCT is similar. In this case $\Delta t = N\Delta t_{radiograph}$ is the time it takes to scan enough data for one 3D reconstruction, using N projections that each require an illumination time $\Delta t_{radiograph}$. In order to avoid limited angle artefacts (see section 2.5.3), N is a number of projections taken over $180° + 2\alpha$, with α half the opening angle of the cone beam. Depending on the scanning system, the amount of time for a full 360° rotation could be 0.1 s on a synchrotron ([81] mentions twenty scans per second, each half a rotation instead of a full rotation because synchrotrons work with parallel beams), to a few seconds on a lab-based 4D μCT scanner (12 s on EMCT [46]). This is plenty of time for a fast dynamic process to change the sample. For example, a fly moves its wings at 50 to 2000 Hz [82], a crack in plexiglass can propagate faster than $330\,\mathrm{m\,s^{-1}}$ [83] and a fluid propagating through a porous medium makes so called "Haines jumps" within a timespan of the order of magnitude of milliseconds [84].
Figure 2.22 presents an overview of the temporal and spatial resolutions of

several real-world synchrotron and laboratory systems, including the EMCT (section 2.7) used in this work. Note that for practical reasons, these resolutions are estimated from different samples, scanned at these machines and published in literature. Each scanner can handle a range of resolutions, while the resolution for only one sample scanned is displayed in this figure. Therefore, the figure does not reflect an overall comparison but more a general idea.
The blurring caused by the Δt is called a motion artefact and it is of course

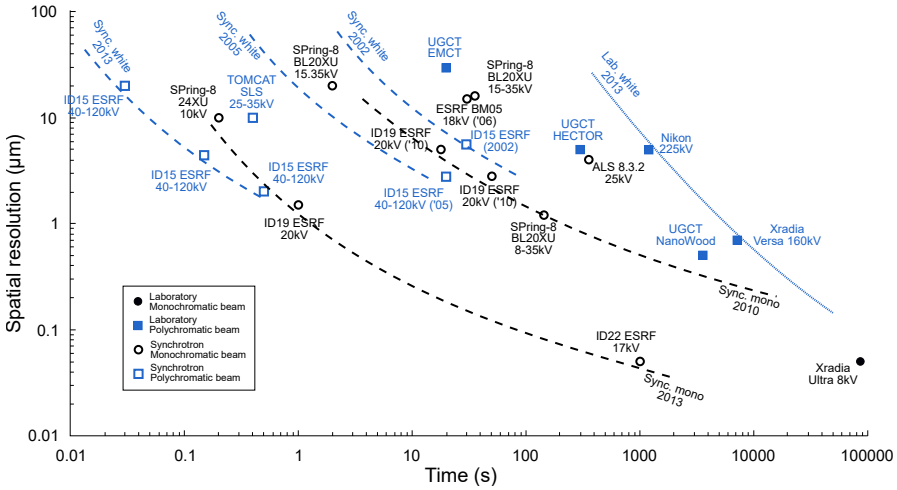


Figure 2.22: Temporal and spatial resolution at different synchrotron beamlines and laboratory μCT scanners. The spatial resolution, when not cited, is estimated as twice the used voxel size. The temporal resolution is the time needed to gather 1000 projections. The lines indicate the state of the art in resolutions of synchrotron (sync.) or laboratory (lab.) and for monochromatic (mono) or polychromatic beams (white) for a certain year. This means μCT scanners in that year of that type could be anywhere to the right of and above this line. Adapted from [19].

an unwanted effect in a reconstruction. To minimise motion artefacts, the scan should go as fast as possible (though this still might not be fast enough). However, a faster scan means less photons got the chance to reach the detector in this time,therefore there are less statistics in the detected projections and hence more noise. Noise in CT scans is an unwanted effect as well (see section 2.5.1). To minimise noise, the scan should be as slow as possible.
Clearly these two image artefacts (noise and motion blurring) counteract each other [85]. It is impossible to minimise one without worsening the other. This is called the law of conservation of misery [86]. Therefore, dynamic μCT is challenging and warrants further research.

2.7 The EMCT: scanner for 4D μCT

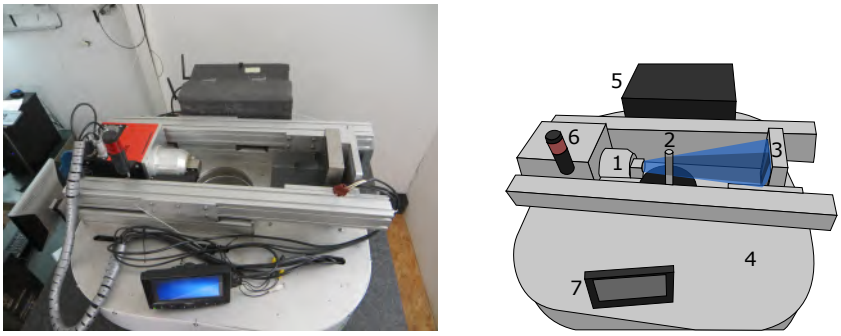


Figure 2.23: The environmental micro-CT system (EMCT [46]), one of the CT scanners at the UGCT ([2]), with some components indicated as explained in section 2.7. The blue cone shows the path of the X-rays.

In this research, the environmental micro-CT system or the EMCT [46, 3], one of the μCT scanners of the UGCT (section 1.1), provided all the experimental data.
A picture and a schematic representation of the EMCT is depicted in figure 2.23. The indicated components are:

1. The X-ray tube. This is a 130 kV source with integrated high-voltage power supply, a Hamamatsu L9181. It has a minimum spot size of 5 µm and a maximum power output of 39 W.

2. The object under investigation, i.e. the sample. Since this picture was taken when no real sample was present, the schematic shows a fictious sample at the location where a sample would normally be. Samples are mounted through the hole visible between X-ray tube and detector. Under this hole, beneath the granite table on which the EMCT rests, is a vertical stage that can move the sample up and down. In addition, a piezo stage can move the sample horizontally in both dimensions, in order to position it precisely on the center of rotation.

3. The detector, a Xineos1313. This is a CMOS flat panel with 1316 by 1312 pixels, each 100 µm high and wide. In order to scan strongly attenuating objects while maintaining a high resolution, a thick structured CsI-scintillator is used.

4. The rotation stage. This is the entire structure on which both the X-ray tube and the detector are mounted.

5. The computer saving the radiographs.

6. Warning light that goes on together with the X-ray tube. This is one of the safety mechanisms to ensure no people are inside the shielded area when the X-rays are on.

7. Screen to view the controls when near the scanner (for example for mounting a sample).

The EMCT is the only UGCT scanner with a rotating gantry. This means the source and detector rotate around the sample instead of vice versa, which is a huge advantage for performing dynamic scans. Since the sample is now stationary, periferal equipment (add-on modules) can be attached to it and the scanner can perform its continuous rotation without entangling cables and tubes running to the sample. Such equipment is meant to change the environment of the sample (hence the name of the EMCT), for example the pressure, humidity or temperature. In addition, the gantry is set up in a horizontal position (in contrast to most medical CT scanners), so the rotational accuracy is not affected by changing gravitational forces on the components. The gantry is mounted on a slip ring [87] embedded in the granite table on which the EMCT is mounted. Consequently the EMCT can perform multiple rotations without having to return to its original position. This is a big advantage for 4D μCT as well, where multiple rotations are a requirement.
In addition, the components of the EMCT were chosen to allow for fast scanning: a full rotation can be completed in 12 s, which is very fast for a high resolution μCT scanner.

2.8 Experimental data

The methods investigated in this thesis are demonstrated on a 4D μCT dataset obtained to study the behaviour of fluids in the pores of porous geomaterials, such as groundwater flow in rocks and sediments.
This dataset considered fluid flow through a Bentheimer sandstone. Figure 2.24 shows a picture of a Bentheimer sandstone sample, albeit a different (bigger) one than the one that was used as described further. This experiment was performed to investigate the intrusion of oil into a water-filled porous rock, with relevance to the study of groundwater pollution, CO_2 storage and petroleum reservoirs [19, 84]. The sandstone was roughly cylindrical, with a height (z-axis) of 10 mm and a diameter (x and y-axes) of 6 mm. It was encompassed in a viton sleeve [88].

Figure 2.24: Picture of a Bentheimer sandstone sample.

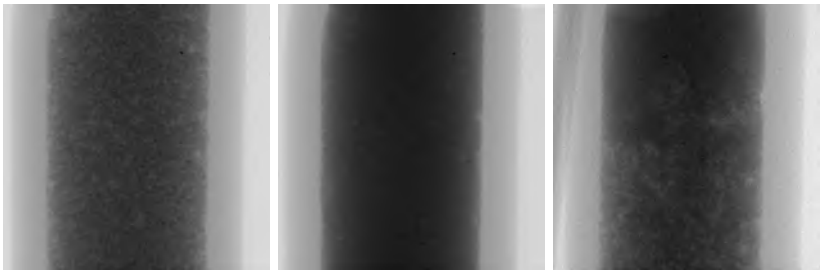


Figure 2.25: Projections of the Bentheimer sandstone scans. Left is the first scan, where pores were visible. In the middle is the scan for the initial volume. Right is a projection from the dynamic scan.

The experimental data from this sandstone was acquired using the EMCT micro-CT scanner [46] (see section 2.7). In total, three separate μCT scans were performed. One projection of each is displayed in figure 2.25. The first scan was a high quality static scan of the rock in which the pores were filled with water. This scan took 17 min and 52 s and consisted of 2201 projections of 487 ms exposure each, obtained in one full rotation. This resulted in a volume in which the pores were clearly visible. The second scan was a high quality static scan of the rock, but now the pores in the rock were filled with a aqueous CsCl solution (10 wt% CsCl), made to match the attenuation coefficient of the rock's solid quartz grains. As such, the dynamic process of oil intrusion is imaged as a binary volume with only two attenuation coefficient values. Over 14 min, 38 s and 1 full rotation, 1801 projections were taken. The reconstruction of this scan can be used as the initial volume in an itera-

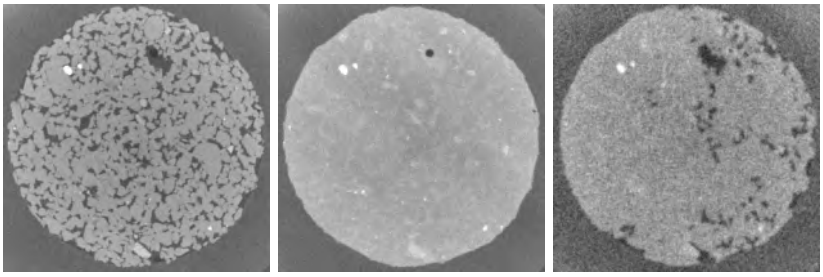


Figure 2.26: Slices of reconstructions of the Bentheimer sandstone scans. Left is the first scan, where pores were visible. In the middle is the scan for the initial volume. Right is a slice from the dynamic scan.

tive reconstruction process. Both these scans were performed with more time for one rotation compared to the third, dynamic scan described next, so they could be reconstructed with a higher signal-to-noise ratio.
Finally, a fast dynamic scan was taken of the rock while oil (n-decane [89]) was pumped in from below at $0.005\,\mathrm{ml\,min^{-1}}$. This scan took 17 min, 6 s and covered 79.8 rotations during which 48 000 projections were acquired (not including dark images and flat fields for normalisation). One rotation took 12.86 s. This sample is an example in which a dynamic process occurs within a static grid, in this case the pores of an otherwise static rock.
Figure 2.25 shows the central slice of an iterative reconstruction of each of these three scans (in the case of the dynamic scan: a subset of the scan consisting of one rotation).
A time series of phantoms, i.e. simulated samples, was also created, based on this Bentheimer sandstone. The first phantom was a reconstruction of the high quality static scan of the rock, in which the pores in the rock were filled with a aqueous CsCl solution. In each subsequent phantom, 0 to 3 randomly chosen pores (segmented as described in section 6.1) were filled. In total, 100 different phantom volumes were created, corresponding to 100 separate moments in time. The last and first phantoms were re-used 10 times, resulting in a time series of 120 phantoms in total. From these phantoms, two dynamic CT scans were simulated by calculating the projections and adding for each dynamic scan different amounts of Poisson noise. The total time series covered 80 rotations (similar to the real scan), meaning each phantom generated 400 projections and the projection parameters were similar to those of the physical dynamic scan, i.e. 600 projections per rotation. When using these projections for a reconstruction, the phantoms can be used as the ground truth for quality testing purposes.

References

[1] B. Masschaele, M. Dierick, D. Van Loo, M. N. Boone, L. Brabant, E. Pauwels, V. Cnudde, and L. Van Hoorebeke. *HECTOR: A 240kV micro-CT setup optimized for research.* Journal of Physics: Conference Series, 463(1):012012, 2013.

[2] B. C. Masschaele, V. Cnudde, M. Dierick, P. Jacobs, L. Van Hoorebeke, and J. Vlassenbroeck. *UGCT: New X-ray radiography and tomography facility.* Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 580(1):266–269, 2007. Proceedings of the 10 th International Symposium on Radiation Physics.

[3] T. De Schryver. *Fast imaging in non-standard X-ray computed tomography geometries.* PhD thesis, Ghent University, 2017.

[4] L. Brabant. *Latest developments in the improvement and quantification of high resolution X-ray tomography data.* PhD thesis, Ghent University, 2013.

[5] K. Sansare, V. Khanna, and F. Karjodkar. *Early victims of X-rays: a tribute and current perception.* Dentomaxillofac Radiol., 40:123–125, 2 2011.

[6] A. De Muynck, S. Bonte, J. Dhaene, M. Dierick, K. Bacher, L. Van Hoorebeke, and M. N. Boone. *Fast method for the estimation of the absorbed dose in X-ray microtomography.* Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 452:40–47, 2019.

[7] S. Itoh, T. Ishiguchi, T. Ishigaki, S. Sakuma, K. Maruyama, and K. Senda. *Mucin-producing pancreatic tumor: CT findings and histopathologic correlation.* Radiology, 183(1):81–86, 1992. PMID: 1312735.

[8] C. Peng, J. Qin, B. Zhou, Q. Chen, M. Shen, M. Zhu, X. Lu, and X. Shi. *Targeted tumor CT imaging using folic acid-modified PEGylated dendrimer-entrapped gold nanoparticles.* Polym. Chem., 4:4412–4424, 2013.

[9] G. Flitton, A. Mouton, and T. P. Breckon. *Object classification in 3D baggage security computed tomography imagery using visual codebooks.* Pattern Recognition, 48(8):2489–2499, 2015.

[10] P. Tafforeau, R. Boistel, E. Boller, A. Bravin, M. Brunet, Y. Chaimanee, P. Cloetens, M. Feist, J. Hoszowska, J. . Jaeger, R. F. Kay, V. Lazzari, L. Marivaux, A. Nel, C. Nemoz, X. Thibault, P. Vignaud, and S. Zabler. *Applications of X-ray synchrotron microtomography for non-destructive 3D studies of paleontological specimens.* Applied Physics A, 83(2):195–202, 5 2006.

[11] D. W. Holdsworth and M. M. Thornton. *Micro-CT in small animal and specimen imaging.* Trends in Biotechnology, 20(8):S34 – S39, 2002.

[12] R. D. B. Sevenois, D. Garoz, F. A. Gilabert, S. W. F. Spronk, S. Fonteyn, M. Heyndrickx, L. Pyl, D. Van Hemelrijck, J. Degrieck, and W. Van Paepegem. *Avoiding interpenetrations and the importance of nesting in analytic geometry construction for Representative Unit Cells of woven composite laminates.* Composites Science and Technology, 136:119–132, 2016.

[13] M. Boone. *3D Visualization of mineral carbonation: μCT as a tool to understand pore scale processes.* PhD thesis, Ghent University, 2015.

[14] V. Cnudde. *Exploring the potential of X-ray tomography as a new non-destructive research tool in conservation studies of natural building stones.* PhD thesis, Ghent University, 2005.

[15] S. N. Saleem and Z. Hawass. *Brief Report: Ankylosing Spondylitis or Diffuse Idiopathic Skeletal Hyperostosis in Royal Egyptian Mummies of the 18th–20th Dynasties? Computed Tomography and Archaeology Studies.* Arthritis & Rheumatology, 66(12):3311–3316, 2014.

[16] M. Sedighi Moghaddam, J. Van den Bulcke, M. E. Wålinder, P. M. Claesson, J. Van Acker, and A. Swerin. *Microstructure of chemically modified wood using X-ray computed tomography in relation to wetting properties.* Holzforschung, 71(2):119–s128, 2016.

[17] K. Van Tittelboom, B. Van Belleghem, M. Boone, L. Van Hoorebeke, and N. De Belie. *X-ray radiography to visualize the rebar-cementitious matrix interface and judge the delay in corrosion through self-repair by encapsulated polyurethane.* Advanced materials interfaces, 5(3):10, 2018.

[18] F. Vanommeslaeghe, W. Van Biesen, M. Dierick, M. Boone, A. Dhondt, and S. Eloot. *Micro-computed tomography for the quantification of blocked fibers in hemodialyzers.* Scientific reports, 8:9, 2018.

[19] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, L. Van Hoorebeke, and V. Cnudde. *Fast laboratory-based micro-computed*

tomography for pore-scale research: Illustrative experiments and perspectives on the future. Advances in Water Resources, 95:341–351, 9 2016.

[20] X. Zhang and L. N. Y. Wong. *Cracking Processes in Rock-Like Material Containing a Single Flaw Under Uniaxial Compression: A Numerical Study Based on Parallel Bonded-Particle Model Approach.* Rock Mechanics and Rock Engineering, 45(5):711–737, 2012.

[21] V. Cnudde, J. P. Cnudde, C. Dupuis, and P. J. S. Jacobs. *X-ray micro-CT used for the localization of water repellents and consolidants inside natural building stones.* Materials Characterization, 53(2):259–271, 2004.

[22] J. Dewanckele, T. De Kock, G. Fronteau, H. Derluyn, P. Vontobel, M. Dierick, L. Van Hoorebeke, P. Jacobs, and V. Cnudde. *Neutron radiography and X-ray computed tomography for quantifying weathering and water uptake processes inside porous limestone used as building material.* Materials Characterization, 88(Supplement C):86–99, 2014.

[23] J. Van Stappen, T. Bultreys, F. A. Gilabert, X. K. D. Hillewaere, D. G. Gómez, K. Van Tittelboom, J. Dhaene, N. De Belie, W. Van Paepegem, F. E. Du Prez, and V. Cnudde. *The microstructure of capsule containing self-healing materials: A micro-computed tomography study.* Materials Characterization, 119(Supplement C):99–109, 2016.

[24] J. A. Elliott, A. H. Windle, J. R. Hobdell, G. Eeckhaut, R. J. Oldman, W. Ludwig, E. Boller, P. Cloetens, and J. Baruchel. *In-situ deformation of an open-cell flexible polyurethane foam characterised by 3D computed microtomography.* Journal of Materials Science, 37(8):1547–1555, 4 2002.

[25] F. Koksel, S. Aritan, A. Strybulevych, J. H. Page, and M. G. Scanlon. *The bubble size distribution and its evolution in non-yeasted wheat flour doughs investigated by synchrotron X-ray microtomography.* Food Research International, 80:12 – 18, 2016.

[26] A. Shastry, P. Palacio-Mancheno, K. Braeckman, S. Vanheule, I. Josipovic, F. Van Assche, E. Robles, V. Cnudde, L. Van Hoorebeke, and M. Boone. *In-Situ High Resolution Dynamic X-ray Microtomographic Imaging of Olive Oil Removal in Kitchen Sponges by Squeezing and Rinsing.* Materials, 11(8), 2018.

[27] V. G. Ruiz de Argandoña, A. Rodriguez Rey, C. Celorio, L. M. S. del Río, L. Calleja, and J. Llavona. *Characterization by computed X-ray tomography of the evolution of the pore structure of a dolomite rock during freeze-thaw cyclic tests.* Physics and Chemistry of the Earth, Part A: Solid Earth and Geodesy, 24(7):633–637, 1999.

[28] Rijksmuseum van Oudheden (Leiden), *Young adult male of the olive or anubis baboon.*

[29] Rijksmuseum van Oudheden (Leiden), 4 2008, *X-ray scan of a baboon mummy.*

[30] G. N. Hounsfield. *Computerized transverse axial scanning (tomography): Part 1. Description of system.* The British Journal of Radiology, 46(552):1016–1022, 1973.

[31] W. Röntgen. *Sur une nouvelle sorte de rayons.* Sitzungsberichten der Würzburger Physik. medic. Gesellschaft, 1:1–10, 1895.

[32] Royal College of Radiologists - London - UK, 2007, *Basics of X-ray Physics: Tissue densities.*

[33] V. Blacus, 10 2012, *Wikipedia: Electromagnetic spectrum.*

[34] S. E. Technologies, 2004, *ISO 21348 Definitions of Solar Irradiance Spectral Categories.*

[35] F. H. Attix. *Introduction to radiological physics and radiation dosimetry.* John Wiley & Sons, 1986.

[36] P. Bouguer. *Essai d'optique sur la gradation de la lumière.* chez Claude Jombert, 1729.

[37] J. H. Lambert. *Photometria sive de mensura et gradibus luminis, colorum et umbrae.* Klett, 1760.

[38] A. Beer. *Bestimmung der absorption des rothen lichts in farbigen flussigkeiten.* Ann. Physik, 162:78–88, 1852.

[39] M. J. Berger, J. H. Hubbell, S. M. Seltzer, J. Chang, J. S. Coursey, R. Sukumar, D. S. Zucker, and K. Olsen. *XCOM: photon cross sections database.* NIST: physical measurement laboratory, 2010.

[40] M. Boone. *New imaging modalities in high resolution X-ray tomography.* PhD thesis, Ghent University, 2013.

[41] J. Vlassenbroeck. *Advances in laboratory-based X-ray microtomography.* PhD thesis, Ghent University, 2009.

[42] O. Klein and Y. Nishina. *Über die Streuung von Strahlung durch freie Elektronen nach der neuen relativistischen Quantendynamik von Dirac.* Zeitschrift für Physik, 52(11):853–868, 11 1929.

[43] Wikipedia, 2016, *Klein-Nishina formula.*

[44] M. Kissick, 2016, *Lecture 8 MP501 (Medical Physics - University of Wisconsin).*

[45] Jarekt, 10 2007, *Wikipedia: Mass Attenuation Coefficient of Iron.*

[46] M. Dierick, D. Van Loo, B. Masschaele, J. Van den Bulcke, J. Van Acker, V. Cnudde, and L. Van Hoorebeke. *Recent micro-CT scanner developments at UGCT.* Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 324:35 – 40, 2014. 1st International Conference on Tomography of Materials and Structures.

[47] A. Bhardwaj, R. F. Elsner, G. R. Gladstone, T. E. Cravens, C. M. Lisse, K. Dennerl, G. Branduardi-Raymont, B. J. Wargelin, J. H. Waite, I. Robertson, N. Østgaard, P. Beiersdorfer, S. L. Snowden, and V. Kharchenko. *X-rays from solar system objects.* Planetary and Space Science, 55(9):1135–1189, 2007. Highlights in Planetary Science.

[48] H. Feng and P. Kaaret. *Spectral State Transitions of the Ultraluminous X-Ray Sources X-1 and X-2 in NGC 1313.* The Astrophysical Journal Letters, 650(1):L75, 2006.

[49] E. Pauwels. *Optimal use of the polychromaticity in X-ray microtomography for dual energy methods and contrast agent imaging.* PhD thesis, Ghent University, 2017.

[50] G. V. Marr. *Handbook on Synchrotron Radiation: Vacuum Ultraviolet and Soft X-ray Processes*, volume 2. Elsevier, 2013.

[51] Wikipedia, 2017, *Synchrotron light source.*

[52] R. Talman. *Accelerator X-Ray Sources.* Wiley, 2007.

[53] D. Elektronen-Synchrotron, 2018, *How does a synchrotron radiation source work?*

[54] Museum voor de Geschiedenis van de Wetenschappen. *Picture of a Coolidge tube*, 10 2017.

[55] LinguisticDemographer, 1 2011, *Wikimedia: Tube spectrum.*

[56] J. F. Barrett and N. Keat. *Artefacts in CT: recognition and avoidance.* RadioGraphics, 24(6):1679–1691, 2004.

[57] R. Behling. *Modern Diagnostic X-Ray Sources: Technology, Manufacturing, Reliability.* CRC Press, 2015.

[58] Y. De Witte. *Improved and practically feasible reconstruction methods for high resolution X-ray tomography.* PhD thesis, Ghent University, 5 2010.

[59] R. Ballabriga, J. Alozy, M. Campbell, E. Frojdh, E. Heijne, T. Koenig, X. Llopart, J. Marchal, D. Pennicard, T. Poikela, et al. *Review of hybrid pixel detector readout ASICs for spectroscopic X-ray imaging.* Journal of Instrumentation, 11(01):P01007, 2016.

[60] M. C. Veale, S. J. Bell, D. D. Duarte, M. J. French, A. Schneider, P. Seller, M. D. Wilson, A. D. Lozinskaya, V. A. Novikov, O. P. Tolbanov, A. Tyazhev, and A. N. Zarubin. *Chromium compensated gallium arsenide detectors for X-ray and γ-ray spectroscopic imaging.* Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 752:6–14, 2014.

[61] B. G. Lowe and R. A. Sareen. *Semiconductor X-Ray Detectors.* Series in Sensors. CRC Press, 2013.

[62] M. Dierick. *Tomographic Imaging Techniques using Cold and Thermal Neutron Beams.* PhD thesis, Ghent University, 2005.

[63] A. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging.* Society for Industrial and Applied Mathematics, 2001.

[64] S. Pan and A. Kak. *A computational study of reconstruction algorithms for diffraction tomography: Interpolation versus filtered-backpropagation.* IEEE Transactions on Acoustics, Speech, and Signal Processing, 31(5):1262–1275, 10 1983.

[65] Wikipedia, 2018, *Nyquist frequency.*

[66] L. A. Feldkamp, L. C. Davis, and J. W. Kress. *Practical cone-beam algorithm.* J. Opt. Soc. Am. A, 1(6):612–619, 6 1984.

[67] S. Karczmarz. *Angenäherte Auflösung von Systemen linearer Gleichungen.* Bull. Int. Acad. Pol. Sic. Let., Cl. Sci. Math. Nat., pages 355–357, 1937.

[68] M. Beister, D. Kolditz, and W. A. Kalender. *Iterative reconstruction methods in X-ray CT.* Physica Medica, 28(2):94 – 108, 2012.

[69] K. Mueller, R. Yagel, and J. F. Cornhill. *The weighted-distance scheme: a globally optimizing projection ordering method for ART.* IEEE Transactions on Medical Imaging, 16(2):223–230, 4 1997.

[70] J. Dhaene. *Development and application of a highly accurate polychromatic X-ray microtomography simulator.* PhD thesis, Ghent University, 2017.

[71] R. Schulze, U. Heil, D. Groβ, D. Bruellmann, E. Dranischnikow, U. Schwanecke, and E. Schoemer. *Artefacts in CBCT: a review.* Dentomaxillofacial Radiology, 40(5):265–273, 2011.

[72] M. Heyndrickx. *Evaluatie van iteratieve reconstructie in dynamische tomografie door gebruik van initiële oplossingen.* Master's thesis, Ghent University, 2015.

[73] Wikipedia, 2018, *Poisson distribution.*

[74] J. Frikel and E. T. Quinto. *Characterization and reduction of artifacts in limited angle tomography.* Inverse Problems, 29(12):125007, 2013.

[75] D. M. Pelt, K. J. Batenburg, and J. A. Sethian. *Improving Tomographic Reconstruction from Limited Data Using Mixed-Scale Dense Convolutional Neural Networks.* Journal of Imaging, 4(11), 2018.

[76] B. D. Smith. *Image Reconstruction from Cone-Beam Projections: Necessary and Sufficient Conditions and Reconstruction Methods.* IEEE Transactions on Medical Imaging, 4(1):14–25, 3 1985.

[77] C. Jailin and S. Roux. *Dynamic Tomographic Reconstruction of Deforming Volumes.* Materials, 11(8), 2018.

[78] J. Cant, W. J. Palenstijn, G. Behiels, and J. Sijbers. *Modeling blurring effects due to continuous gantry rotation: Application to region of interest tomography.* Medical physics, 42(5):2709–2717, 2015.

[79] M. Soret, S. L. Bacharach, and I. Buvat. *Partial-volume effect in PET tumor imaging.* Journal of Nuclear Medicine, 48(6):932, 2007.

[80] G. L. Zeng, 2007, *A Skew-Slit Collimator for Small Animal SPECT.*

[81] S. C. Garcea, Y. Wang, and P. J. Withers. *X-ray computed tomography of polymer composites.* Composites Science and Technology, 156:305–319, 2018.

[82] N. Lee, 2000, *Frequency Of Fly Wings.*

[83] S. Chasteen. *Focus: Cracking the Story of Fracture.* Physics, 25:5, 2010.

[84] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, D. Van Loo, L. Van Hoorebeke, and V. Cnudde. *Real-time visualization of Haines jumps in sandstone with laboratory-based microcomputed tomography.* Water Resources Research, 51(10):8668–8676, 2015.

[85] T. De Schryver, M. Dierick, M. Heyndrickx, J. Van Stappen, M. A. Boone, L. Van Hoorebeke, and M. N. Boone. *Motion compensated micro-CT reconstruction for in-situ analysis of dynamic processes.* Scientific Reports, 8:7655, 2018.

[86] Wikipedia, 2011, *The law of conservation of misery.*

[87] Wikipedia, 2018, *Slip ring.*

[88] Wikipedia, 2018, *Viton.*

[89] Wikipedia, 2018, *Decane.*

3

Established approaches to 4D μCT

How other people solve this problem

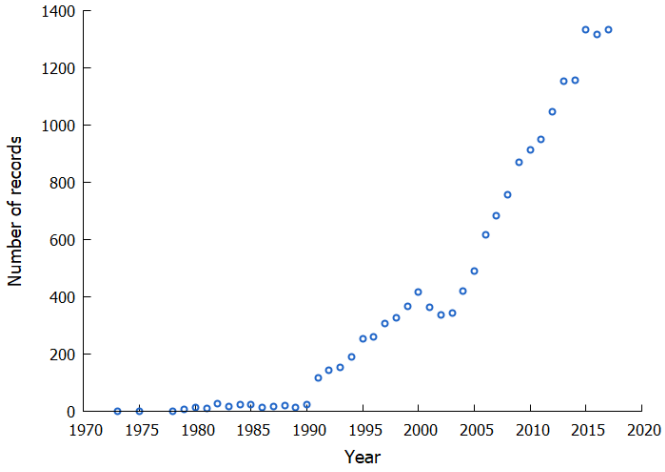


Figure 3.1: The number of articles, proceedings papers and other scientific publications per year with 4D μCT (in different spelling variations) as a topic. Data taken from Web of Science [1].

Interest in 4D μCT as a scientific research topic and tool is increasing, as shown in figure 3.1. As such, there is a lot of literature available on different techniques to handle the difficulties of 4D μCT, either in general or for specific types of samples. Some of them use prior knowledge or a priori knowledge, which is information known about the sample independent of the 4D μCT scan itself. This can for example be information on which materials are present in the sample, or an approximation of the volume (from design or from a previous scan) or the limitation that the attenuation coefficient can only increase or decrease due to the nature of the dynamic process. Other techniques do not use prior knowledge or combine it with another approach. Below is an overview of some of the techniques used to perform 4D μCT.

3.1 Hardware improvements

Although this thesis does not implement any hardware improvements, this research would not have been possible without them. Over the history of CT scanning, all components of a μCT scanner (as explained in section 2.3) have improved, causing μCT scanners to become more suitable for 4D μCT. The evolution and improvements remain ongoing [2, 3].
The original CT scanning system as proposed by Hounsfield [4] used a pencil beam, requiring a translation as well as a rotation to scan the entire 3D volume (see section 2.3.2). Since then, **cone beam CT** has been developed [5],

providing a huge advantage for 4D μCT. The cone beam geometry is depicted in figure 2.7. In this set-up, an entire 2D radiograph can be recorded at once, resulting in a large speed-up with respect to pencil beams or fan beams. This faster scan causes less motion artefacts in dynamic CT. Note that cone beam CT suffers from cone beam artefacts at the top and bottom of the reconstruction (see section 2.5.7). However, the sample can be positioned so the interesting part is in the middle or even so the top and bottom parts contain surrounding air instead of sample material.
All experimental data used in this thesis were acquired with a cone beam set-up.
The geometry of a cone beam can be further extended. Instead of just recording an entire radiograph at once, some scanners can record multiple radiographs at once, at different angles, thus eliminating part of the rotation. Reducing the need for rotation reduces the scanning time just as eliminating the translation movement did. An example is a **dual source CT scanner**, where only half the rotation time is needed, since two perpendicular sources and detectors operate simultaneously [6]. More sources and detectors are possible, further reducing the rotation time, but few such systems exist in reality. An example of such a multi-source, multi-detector scanner is the Rapiscan RTT 110, equipped with 900 sources and over 12 000 detectors [7, 8], which is depicted schematically in figure 3.2.
The UGCT has no multi-source CT scanners available. Therefore, none of

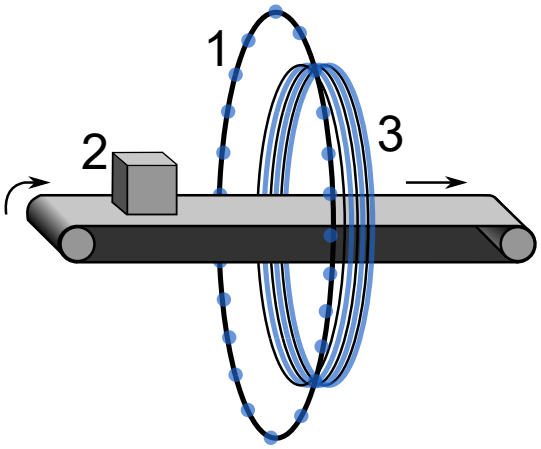


Figure 3.2: Schematic representation of the Rapiscan RTT 110. 1 is a ring of sources. 2 is the sample, located on a conveyor belt. 3 is a ring of detectors. The reconstruction is approximately helical. Adapted from Warnett *et al.* [9].

the data in this thesis was acquired with multi-source or multi-detector CT. The methods developed in this thesis, and more specifically all methods based on iterative reconstruction techniques, are usable on multi-source/multi-detector set-ups without any adaptations.

Advances in detector technology are advantageous for 4D μCT. In contrast to the older xenon gas detector systems or x-ray image intensifiers (XRII) coupled to a video read-out, many current scanners use highly efficient scintillation detectors coupled to photodiodes or to cooled charge-coupled detector (CCD) arrays. This caused an improvement in spatial resolution and a decrease in noise [2, 10]. This last improvement means that, for the same signal to noise ratio, less photon statistics are needed, i.e. less time per radiograph, with a faster scan as a result. A different class of detector systems has no scintillator, but converts the absorbed X-rays directly into electron-hole pairs, which increases the response speed, allowing for even faster readout [11]. There is, however, a limit in how fast a detector can record one radiograph. For example, the Xineos-1313, one of the faster detectors of the UGCT and therefore the one used at the EMCT, can acquire a radiograph in about 30 ms [12].

Typically, the limitation to the speed of a CT scan is not how fast a detector can acquire radiographs, but how much flux the X-ray source provides. Therefore, **evolutions in X-ray source technology** are also important to consider. For example, a smaller focal spot size results in a better resolution. However, this smaller spot size means a lower flux, hence a longer exposure time is needed for the same photon statistics [10], which is detrimental for 4D μCT. Other types of sources are also becoming available for μCT, such as liquid metal-jet sources, which can achieve a higher flux but may be limited in resolution [11].

The speed of the rotation stage has increased since the original invention of CT. This is mostly due to the improvement in detector and X-ray source technology. For medical scanners, rotation times from 270 ms to 350 ms can be reached [2]. At the UGCT, a complete rotation of the EMCT (the scanner most often used for dynamic scans, see section 2.7) takes at least 12 s [13].

A rotating stage for a gantry typically reaches slower speeds than one for a sample. For the latter, the rotation speed can be a problem when scanning big or bulky objects, especially when peripheral equipment is connected to it. When the system is not perfectly aligned, fast rotation can in that case break the rotation stage, the sample or other equipment.

The development of **add-on modules** allows for more applications to be scanned with 4D μCT. These are devices to accurately control the conditions in which a sample is scanned, for example the temperature, pressure, humidity, flow or compression [14]. Two of these add-on modules, one for temperature and one for compression, are described in detail in De Schryver

[14]. A discussion on flow cells, which control fluid movement through a porous material, can be found in Bultreys *et al.* [11].
All of these improvements have made CT with spatial resolutions below 100 nm possible, as well as acquiring radiographs fast enough to get multiple 3D images per second at synchrotrons [15]. High-resolution dynamic μCT is also becoming possible in lab-based systems [13, 16]. An extensive review of both spatial and temporal resolutions at CT scanners is available in Maire and Withers [15], while figure 2.22 shows a selection of spatio-temporal resolutions of systems around the world.

3.2 Initial volume

Using an initial volume [12, 17] is perhaps the most straightforward way to incorporate prior knowledge into the reconstruction mechanism. It works by taking advantage of a property of iterative reconstruction algorithms (see section 2.4.2): they can start from any 3D volume. Usually the starting volume contains all zeros, but in this case, the starting point is a so-called initial volume [17].
An initial volume is a volume that should resemble the real sample-volume. This means that the difference $\sum_j \mu_{j,real} - \mu_{j,init}$ between the grey values will be smaller when choosing a good initial volume than it would be when choosing $\mu_{j,init} = 0$ for all voxels j. The iterative reconstruction will therefore have to reconstruct less difference and will converge faster to the correct solution. This not only means there is less reconstruction time needed, but also that the noise, that tends to come in stronger after a few iterations, has less chance to increase. In addition, less projections are needed for the same result, which means less time to record these projections and a faster scan, which is beneficial for 4D μCT. [12]
As was proven by Jiang and Wang [18], an iterative reconstruction such as SART converges to a volume close to the real volume irrespective of the initial volume chosen for theoretically perfect data. In practice, noise and a limited amount of projections may make the correct solution unreachable. In this case, the resulting volume is sensitive to the initial volume [19], which should therefore be chosen wisely.
The initial volume is usually a static μCT scan performed before or after the dynamic process under consideration. Since the dynamic scan will be of the same sample, only changed by whatever dynamic process is imaged, the volumes μ_{real} and μ_{init} can be close together when the dynamic process does not consist of movement. An example is fluid flow through rock: an initial volume could be a high resolution scan performed before adding the fluid.
An initial volume works great when the dynamic process is taking place in a static grid. This means a large part of the volume remains static during the

scan and is therefore almost perfectly imaged by the initial volume already. Real movement of the sample is not addressed by an initial volume technique. Therefore, whether an initial volume is a good choice depends on the type of dynamic process.
A caveat when using an initial volume is the visual appeal of wrong reconstructions: when the reconstruction, for any reason (for example too few iterations), is still close to the initial volume instead of having reconstructed the $\mu_{j,real}$, the visual end result seems reasonable to the human eye. Therefore, the observer might not realise the reconstruction was wrong. In contrast, when starting from an empty volume, this is immediately clear. [12]
An initial volume combined with compressed sensing (more on that in section 3.4) is called prior image constrained compressed sensing (PICCS) [20]. Any CT reconstruction approach allowing for the use of iterative reconstruction can be complemented with the use of an initial volume.

3.3 Motion registration

Motion registration is a technique which decreases the motion artefacts, making it suitable for dynamic processes consisting of movement. Since weighted back projection (see chapter 4) handles dynamic processes occurring within a static grid, the combination of both would be an exciting future prospect, able to handle even dynamic processes that are happening within a deforming grid.
Motion registration estimates the motion of all parts of the volume. Some algorithms go further and incorporate this motion into the reconstruction. This results not only in the reconstructed volume(s), which will be of higher quality than without motion correction, but also in the motion estimate itself, which may provide valuable information about the dynamic process. For example, the displacement vectors from this motion can result in strain maps [15, 21].
The estimated motion is called a displacement field $\mathcal{M}$ which maps the coordinates of a 'fixed' frame $\vec{x_f}$ to the coordinates in a moving frame $\vec{x_m}$ [14]:

$$\mathcal{M}(\vec{x_f}) = \vec{x_m} = \vec{x_f} + \vec{u}(\vec{x_f}) \tag{3.1}$$

The deformation $\vec{u}(\vec{x_f})$ is known as a 'dense displacement field' or 'Motion Vector Field' (MVF). The motion registration technique now consists of two major parts: estimating the MVF (the motion registration itself) and incorporating the MVF in the reconstruction to reduce motion blurring [21]. Both have different approaches described in literature.

3.3.1 Motion vector field estimation

The estimation consists of 3 essential components which may change depending on the used algorithm [14]. First is the choice on the similarity metric, which is used to compare the moving and fixed volumes. The aim of the registration algorithm is to deform (equation 3.1) the moving volume $\mu(\vec{x_m})$ in such a way as to maximize the similarity with the fixed volume $\mu(\vec{x_f})$ [22, 23, 24, 25, 26, 27]. Second is the choice on how the MVF is parametrized. This is equivalent to choosing the basis functions that represent the displacements. Estimating the displacement then means finding the best values for the parameters of this function [28, 29, 30, 31, 32].
Third is the strategy used to find an optimal set for the deformation parameters. The optimal parameters are often found through a hierarchical or resolution-pyramid scheme: the parameters are first searched at larger scales and the results are used as starting points to search at increasingly smaller scales (possibly down to voxel level, but not necessarily) [14, 30, 22, 28, 33]. A thorough explanation of the motion estimation and aspects of it can be found in [14, 27, 25, 15] amongst others.

3.3.2 Incorporating the displacement field in the reconstruction

Van Nieuwenhove *et al.* [34] and De Schryver *et al.* [14, 21] describe two methods that incorporate an MVF into an iterative reconstruction algorithm (see section 2.4.2).
The first step is to reconstruct certain time steps in a static way, resulting in a number of 3D volumes each corresponding to a certain time point and (probably) containing motion artefacts, since they were reconstructed using projections that were taken over a certain time period. Then, a motion registration is performed for each adjacent pair of 3D volumes. Finally, an iterative reconstruction is performed while deforming the volume according to the found MVF.
Remember, when there is only a fixed state $\vec{x_f}$, the steps in an iterative reconstruction are equations 2.60 and 2.64, respectively the forward and the backward projection. While reconstructing a certain time point T with coordinates $\vec{x_m}$, the algorithm uses other nearby time points t (coordinates $\vec{x_f}$) as well, since every radiograph corresponds to a different time point. These time points are connected through the known or estimated MVF $\vec{u}$ (see section 3.3.1 and equation 3.1). In the projection step, the volume is transformed according to $\vec{x_f} + \vec{u}(\vec{x_f})$ to the projection time t. In the back projection step, the volume is transformed in the opposite direction $\vec{x_f} - \vec{u}(\vec{x_f})$ to get back to the wanted time point T. A thorough explanation of this is found in De Schryver *et al.* [14, 21] and Van Nieuwenhove *et al.* [34] and a schematic

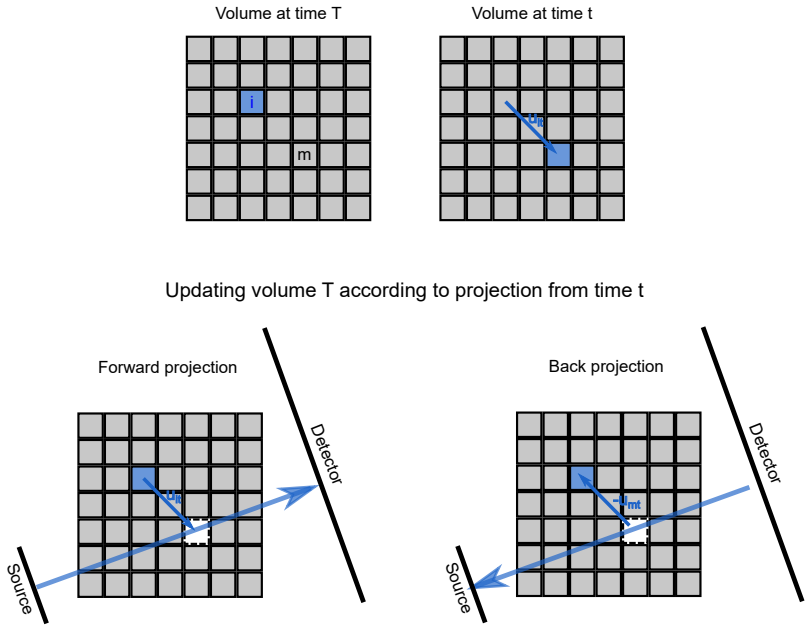


Figure 3.3: Schematic 2D representation of an interative reconstruction using a displacement field $\vec{u}$. The pixel under reconstruction, i, is indicated in blue. At time T (the time being reconstructed), it is at its 'fixed' position, at time t it moved to position m. A projection at time t has X-rays passing through m, they use μ_i from the current reconstructed volume T instead of μ_m in the forward projection. When back projecting this same X-ray, instead of applying the correction to μ_m, it should be applied to μ_i.

representation in figure 3.3.

3.4 Total variation minimisation

Total variation minimisation or compressed sensing is an increasingly popular technique in sparse-data CT and 4D μCT [35]. It was first developed for CT with theoretically insufficient data, i.e. few projections. It combines an iterative reconstruction (see section 2.4.2) with the total variation minimisation noise filter, which assumes the sample consists of constant regions, i.e. the sample is spatially piecewise constant [36, 35].
The total variation minimisation minimises the ℓ_1-norm of an image by using the the gradient descent method [37]. The ℓ_1-norm of an image is the sum of the absolute grey values of the pixels. This minimisation leads to sparse solutions. Since (ideal) reconstructions are not necessarily sparse in the the ℓ_1-norm of the volume itself, the norm that is minimised is actually the ℓ_1-norm of the gradient of the volume. For a piecewise constant volume, the gradient volume *is* sparse. [36]
By combining the iterative reconstruction (for example ART) and the total variation minimisation, the result is a volume consistent with the projection data which has a gradient volume with a sparse ℓ_1-norm. The algorithm will first perform an ART iteration, then a total variation minimisation and repeat from the start until a predetermined criterium is met. This can be a fixed number of iterations or when the changes made in each iteration are smaller than some threshold.
The total variation minimisation uses the gradient descent method, which is in itself an iterative method with a predetermined number of iterations. The main step in one iteration is the following [36]:

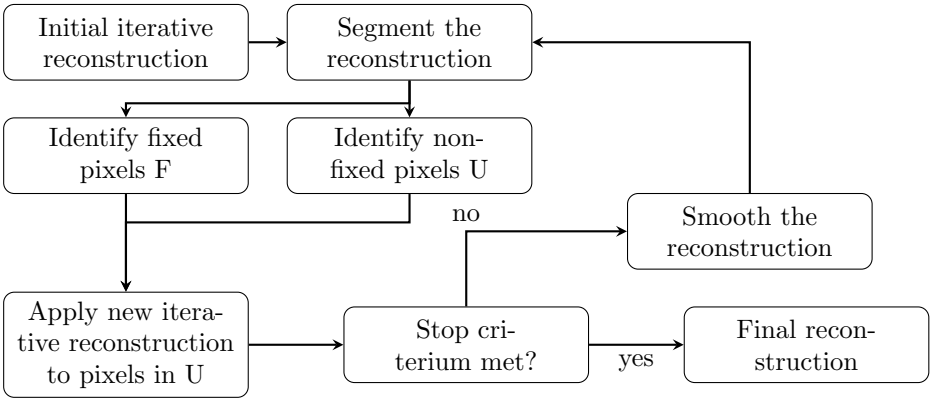
$$\mu_j^k = \mu_j^{k-1} - a d_j v_j^{k-1} \tag{3.2}$$

in which μ is the voxel grey value, k is the iteration step of the gradient descent method, j is the voxel number, a is a parameter between 0 and 1 determining the 'strength' of this noise filter, d_j is how much this voxel grey value changed starting from the beginning of the last ART-iteration to the end of the same ART-iteration and v_j^{k-1} is the gradient between the grey value of voxel j and its neighbours.
This reconstruction algorithm can handle sparse projection data. Specifically in 4D μCT, the sparseness is caused by the rapid scanning.
A significant amount of literature builds upon total variation minimisation or adapts it to obtain better results. Xu *et al.* [35] add a weight dependant on the anisotropic variation to better suppress the noise along the edges. The technique described by Grasmair [38] combines the total variation minimisation with an anisotropic diffusion filter to get the benefits of both (though this is not in the context of CT). Chen *et al.* [20] combine compressed sensing with gating (section 3.7) and an initial volume (section 3.2). The result,

with or without gating, is called prior image constrained compressed sensing (PICCS). Lee *et al.* [39] build further upon PICCS by performing a segmentation into air, soft tissue and bone and performing a registration between the current volume and the prior image based on this. In contrast to section 3.3, this is not used to find the dynamic process, but simply to align scans that are taken days apart. Regions where a mismatch of the segmented regions occurred between prior image and current volume get a higher weight, used in the total variation minimisation step. The algorithm is called adaptive prior image constrained compressed sensing (APICCS).
One property of total variation minimisation is the assumption of a piecewise constant sample, resulting in 'cartoon-like' images with a 'staircase' effect: at a gradient, the grey value makes discrete jumps [38], which is a wanted effect for many samples.

3.5 Discrete reconstruction

When a sample consists of a limited amount of materials, each with a fixed density, there is only a limited number of attenuation coefficients. In this case, the fact that there is a discrete amount of attenuation coefficients can be used as prior knowledge in the reconstruction, leading to what is called a discrete reconstruction.
With the attenuation coefficients of the sample known in advance, a discrete reconstruction technique such as DART, the discrete algebraic reconstruction technique [40], becomes possible. In this technique, a well known iterative reconstruction technique (see section 2.4.2) is used to create a starting volume. In this volume, fixed pixels are those with a grey value that is already correct, i.e. that belongs to one of the known materials in the sample. Non-fixed pixels are those with a differing grey value or directly adjacent to one such pixel.
Next, the iterative reconstruction technique is repeated, but the fixed points are kept unchanged (see also chapter 4). After this, a smoothing filter is applied. This concludes an iteration, which continues until a certain stopping criterium is met (a fixed number of iterations or a certain minimal level of change during a complete iteration).
The following flowchart shows an overview of DART [40] :



A discrete algorithm provides an enormous improvement and allows for a good reconstruction even with few or noisy projections. The downside is that this is only possible for samples which consist of only a few known materials. Herman and Kuba [41] provide an extensive overview of discrete reconstruction techniques and the math behind them, including techniques where the precise attenuation coefficients of the materials are now known beforehand.

3.6 Machine learning

Machine learning is a programming technique applied in an increasing number of domains, such as face recognition, chat bots, handwriting recognition, spam filters, traffic prediction, product recommendations and μCT. Machine learning allows an algorithm to "learn", i.e. to get increasingly better at a given task, without explicit programming on how to perform this task [42, 43]. One subset of machine learning is a neural network, which is what this section will focus on.
A neural network is a sort of 'black box'. It reads the input data, makes some internal decisions and outputs what it considers to be the correct solution. These internal decisions are not created by the programmer. Instead, the network has undergone a training phase. During training, the neural network is presented with many examples of input combined with the correct output. For example, in μCT, the input could be the radiographs and the output the reconstruction volume. Or the input could be the reconstruction and the output the analysis result. This training data consists of objectively correct outputs for each given input, otherwise the network will train itself to produce wrong results. Therefore, training data for μCT is typically based on high-quality scans. The input (radiographs) may be truncated to simulate a 'bad scan' that the neural network should eventually be able to handle [3].

The output, however, should be the 'correct' reconstructions from the complete, high-quality scan. Another possibility is to reconstruct the training data with a different method that is able to handle the low-quality scans, but might not be the preferred method for all performed scans because it requires a large amount of computer time. [44]
The neural network learns from these examples. It does this by trying to predict the output for a given input and comparing this with the given output. The difference is propagated back through the network. The network becomes iteratively better at predicting the correct output.
After training, new input data from low-quality scans can be fed to the neural network, that will (ideally) produce the correct reconstruction volume.

3.6.1 Internal mechanism

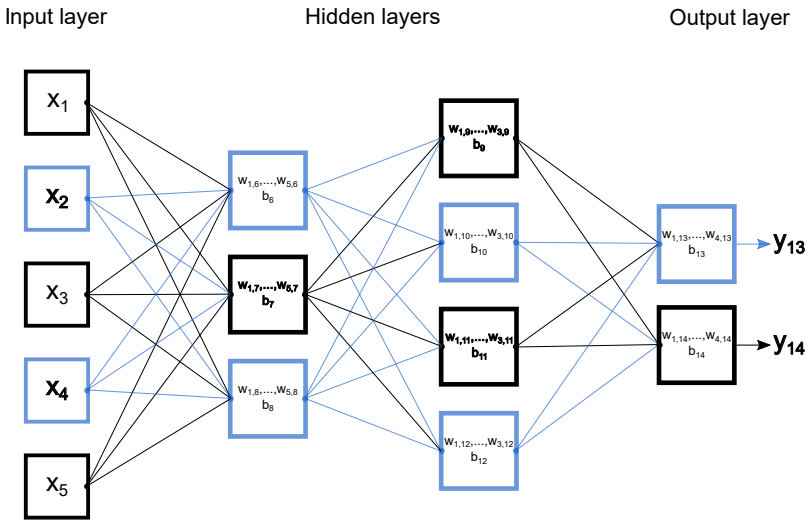


Figure 3.4: Schematic representation of a neural network with 5 input numbers, 2 hidden layers and 2 output numbers. The two different colors serve only to discern the connecting lines from different nodes easier from each other.

While the neural network is often called a 'black box', it still relies on known mathematics.
A network consists of a number of layers as represented in figure 3.4. One layer is the input layer, one is the output layer and all the other layers between them (there's at least one) are called hidden layers. The amount of layers can be chosen by the user or the programmer. There are no fixed rules dictating

how many layers each application would need.
A number of nodes make up each layer. For the hidden layers, the number of nodes in any layer is not governed by clear rules, so the programmer or user fixes this before starting the algorithm. It is hard to determine the ideal number of nodes: too few and the network will not be able to correctly map the input to the output. Too many and the chance of overfitting increases, as explained in section 3.6.2 [44].
Each node j has N_j input numbers (x_{1j}, x_{2j}, etc.), one output (y_j) which is sent to multiple nodes in the next layer, N_j internal weights (w_{1j}, w_{2j}, etc.) with N_j being the number of nodes in the previous layer, and a threshold value b_j, . Note that while a node only has one output, the complete network can have several. The input numbers of a node are the output numbers of the nodes of the previous layer. The j'th node transforms input to output as follows:

$$y_j = f(b_j + \sum_{i=1}^{N_j} w_{ij} \cdot x_{ij}) \tag{3.3}$$

f is a function that is the same for each node, but may be different between two neural networks, depending on the choice of the programmer. It is called the activation function and sometimes called σ instead of f. The most commonly used is the sigmoid function:

$$f(x) = \frac{1}{1 + \exp(-\beta x)} \tag{3.4}$$

β is a parameter chosen beforehand for this neural network. It determines the slope of the function [45]. For example, in Pelt *et al.* [44], β is 1.
The output layer often has a different activation function than the other layers, for example $f(x) = x$ as in Pelt *et al.* [46].
A lot of networks use the simplest way to connect the nodes from different layers: a node gets input from each node in the previous layer and from none of the other layers. However, it is possible to connect a node to nodes from several layers before, for example in Mixed-Scale Dense networks, allowing for sufficient training with few data available [46].
During training, the network will improve the weights w_{ij} and thresholds b_j parameters. Improvement in this case means the correct output, the final y's, will be produced for an input x_j within a certain application. There are several training methods. A supervised learning training method presents the network with a set of correct outputs mapped to their input. These inputs and outputs are called the training data. [46]
The goal of the training phase is to minimise the total mean square error between calculated output and given output. There are several minimisation

methods, such as gradient descent [45, 43], the Levenberg-Marquardt algorithm [44], the proximal point algorithm and the Chambolle-Pock algorithm [47].
The initial weights and thresholds are picked randomly [45] or set to 0 [47]. The training phase works iteratively. It calculates the output from a training input data set with its current weights and thresholds (parameters). It then compares this output with the correct training output and adapts its parameters according to the chosen minimisation method. Gradually, this minimises the total mean square error between calculated output and given output. The number of iterations can be fixed [47] or based on a stopping criterium such as an error computed with a validation data set (more on this in section 3.6.2) [44].
After training, the neural network produces output for a given input, never seen in the training phase, with the final parameters it obtained from training.
A neural network can be a very powerful tool to process new data from applications where the relation between input and output is not easily described [43]. The network can find patterns between input and output that the programmer or researcher couldn't notice or formulate. In addition, while some neural networks take a long time to train and to compute the output, others have the advantage of the fast reconstruction computation time that is typical for analytical reconstructions. A notable example is a neural network that is a weighted sum of filtered back projections (see section 2.4.1) [44].

3.6.2 Disadvantages and dangers

While the advantages of a neural network are clear, i.e. fast processing of incomplete or otherwise imperfect data while maintaining a very high reconstruction quality [44, 46], there are also some disadvantages to keep in mind when considering a neural network for an application.

- There is a risk of underfitting if there is not enough training data. This means the model will not be trained enough and won't produce the correct output for a new input.
- There is a risk of overfitting if there is too much training data. The neural network is no longer sufficiently generic, and it can not handle completely new input data well.
- The training data has to include all possible situations the real data can encounter.
- The training data has to resemble the real input data the neural network will have to process.

- Since the neural network is a black box, i.e. it is unknown what patterns it actually recognises in the data, there is no certainty it will always produce the correct result. When dealing with new data, this uncertainty is a major disadvantage.

A notable example of an incomplete training data set, combined with the uncertainty of the neural network can be found in Wolfson [48], where a simple neural network discerned dogs from wolves. Eventually, it turned out to recognise snow instead of wolves, since most wolf training data were pictures from wolves with a snowy background.
Overfitting can be prevented by using a second set of training data, the *validation set*. The error between calculated output and validation output is calculated at each iteration. When this error stops improving and instead starts deteriorating, overfitting is occurring. The training therefore stops at that point. [44]
For the samples considered in this thesis, neural networks are not an option because there is not enough training data. The samples used for research are usually unique or there's only a small set of them to be scanned. Applications of neural networks within 4D μCT are still relatively rare because it requires many similar samples. However, it is also possible to increase the training data set by considering small parts of the scan (for example each set of 100 neighbouring pixels) as one input and output data set, or even, as in Pelt *et al.* [44] one voxel at a time.

3.7 Gating

When the dynamic component of the sample is a periodic movement, gating (or single exposure phase retrieval) becomes available for correctly reconstructing the different phases of the sample. A periodic or quasi-periodic movement is common in medical CT. Examples are the cardiac movement (heartbeat) and the respiratory movement (breathing). Sometimes the scan takes into account both the cardiac and the respiratory movement [10].
The CT scanner performs multiple rotations, as usual in 4D CT. Each projection therefore has a certain projection angle associated with it, which is important for the reconstruction. In addition, since the dynamic process is periodic, each projection was taken during a certain part or phase in this process. The same phase returns for a projection T time later, with T being the period of the movement. When $T \neq R$ (R being the duration of one rotation), the next projection that has the same phase will be taken at a different angle. [49]
Gating combines the projections belonging to the same phase into reconstruction sets. Each reconstruction set corresponds to one phase and contains

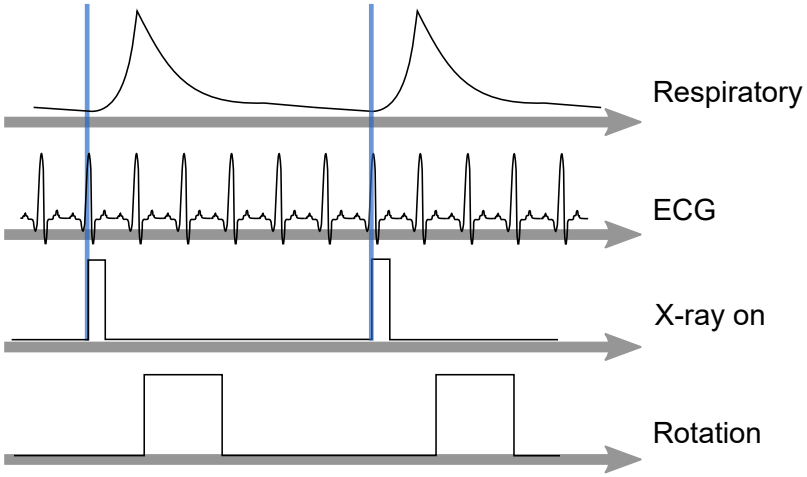


Figure 3.5: Procedure for rotation and radiograph recording ("X-ray on") in a discontinuous gating sequence in medical CT, dependant on both the heart phase (measured with an ECG) and the respiratory phase. Adapted from Guo *et al.* [49].

projections that were not taken sequentially, but should span the whole rotation domain. Since a single phase is static, each reconstruction set can be reconstructed without motion artefacts. Putting the reconstructions of the different phases together spans the complete dynamic process.
Gating can be performed non-continuously as in figure 3.5. This means a phase to reconstruct is chosen first. The projections are now taken exclusively at the times of this phase: a projection is taken at the correct time, the gantry rotates to the next position and then the scanner waits. When the phase occurs again, a new projection is taken and the process is repeated until the complete projection set is acquired. This can be repeated for other phases if they are of interest. [49]
Gating eliminates the difficulties of 4D CT, especially when the phase per projection can be determined accurately, for example by an outside measurement such as an ECG [50]. The major drawback is the limited applicability: the dynamic process must be periodic. This assumption is false for the dynamic processes considered in this thesis, hence the gating technique is not used on the samples presented in this work.
Gating is most relevant in medical CT, in studies of the heart [10, 51] and the lungs [52]. In high resolution μCT, the application of gating is rare, with one such rare example being the quasi-periodic wingbeat of a fly in motion [53]. In medical CT, gating can even be considered the 'default', since new

techniques for 4D cardiac or respiratory CT are often compared with it [54] or build upon it [55, 52].

References

[1] C. Analytics, 2019, *Web of Science*.

[2] S. Halliburton, A. Arbab-Zadeh, D. Dey, A. J. Einstein, R. Gentry, R. T. George, T. Gerber, M. Mahesh, and W. G. Weigold. *State-of-the-art in CT hardware and scan modes for cardiovascular CT*. Journal of Cardiovascular Computed Tomography, 6(3):154–163, 2012.

[3] D. Parkinson, D. Ushizima, H. Krishnan, T. Perciano, and J. Sethian. *Achieving fast high-resolution 3D imaging by combining synchrotron x-ray microCT, advanced algorithms, and high performance data management (Conference Presentation)*. page 26, 5 2018.

[4] G. N. Hounsfield. *Method and apparatus for measuring x- or y-radiation absorption or transmission at plural angles and analyzing the data*, 12 1971.

[5] R. A. Robb. *The Dynamic Spatial Reconstructor: An X-Ray Video-Fluoroscopic CT Scanner for Dynamic Volume Imaging of Moving Organs*. IEEE Transactions on Medical Imaging, 1(1):22–33, 7 1982.

[6] B. Krauss, B. Schmidt, and T. G. Flohr. *Dual source CT*. In Dual energy CT in clinical practice, pages 11–20. Springer, 2011.

[7] R. systems, 2018, *Rapiscan RTT*.

[8] J. M. Warnett, V. Titarenko, E. Kiraci, A. Attridge, W. R. B. Lionheart, P. J. Withers, and M. A. Williams, 2018, *Towards in-process X-ray CT for dimensional metrology*.

[9] J. Warnett, V. Titarenko, E. Kiraci, A. Attridge, W. R B Lionheart, P. Withers, and M. Williams. *Towards in-process x-ray CT for dimensional metrology*. Measurement Science and Technology, 27, 1 2016.

[10] C. T. Badea, M. Drangova, D. W. Holdsworth, and G. A. Johnson. *In vivo small-animal imaging using micro-CT and digital subtraction angiography*. Physics in Medicine & Biology, 53(19):R319, 2008.

[11] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, L. Van Hoorebeke, and V. Cnudde. *Fast laboratory-based micro-computed tomography for pore-scale research: Illustrative experiments and perspectives on the future*. Advances in Water Resources, 95:341–351, 9 2016.

[12] M. Heyndrickx. *Evaluatie van iteratieve reconstructie in dynamische tomografie door gebruik van initiële oplossingen.* Master's thesis, Ghent University, 2015.

[13] M. Dierick, D. Van Loo, B. Masschaele, J. Van den Bulcke, J. Van Acker, V. Cnudde, and L. Van Hoorebeke. *Recent micro-CT scanner developments at UGCT.* Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 324:35 – 40, 2014. 1st International Conference on Tomography of Materials and Structures.

[14] T. De Schryver. *Fast imaging in non-standard X-ray computed tomography geometries.* PhD thesis, Ghent University, 2017.

[15] E. Maire and P. J. Withers. *Quantitative X-ray tomography.* International Materials Reviews, 59(1):1–43, 2014.

[16] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, D. Van Loo, L. Van Hoorebeke, and V. Cnudde. *Real-time visualization of Haines jumps in sandstone with laboratory-based microcomputed tomography.* Water Resources Research, 51(10):8668–8676, 2015.

[17] L. Brabant. *Latest developments in the improvement and quantification of high resolution X-ray tomography data.* PhD thesis, Ghent University, 2013.

[18] M. Jiang and G. Wang. *Convergence of the simultaneous algebraic reconstruction technique (SART).* IEEE Transactions on Image Processing, 12(8):957–961, 8 2003.

[19] D. Van de Sompel and M. Brady. *A systematic performance analysis of the simultaneous algebraic reconstruction technique (SART) for limited angle tomography.* In 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pages 2729–2732, 8 2008.

[20] G. Chen, J. Tang, and S. Leng. *Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets.* Medical Physics, 35(2):660–663, 2008.

[21] T. De Schryver, M. Dierick, M. Heyndrickx, J. Van Stappen, M. A. Boone, L. Van Hoorebeke, and M. N. Boone. *Motion compensated micro-CT reconstruction for in-situ analysis of dynamic processes.* Scientific Reports, 8:7655, 2018.

[22] P. Thevenaz, U. E. Ruttimann, and M. Unser. *A pyramid approach to subpixel registration based on intensity.* IEEE Transactions on Image Processing, 7(1):27–41, 1 1998.

[23] R. Berthilsson. *Affine correlation.* In Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170), volume 2, pages 1458–1460, 8 1998.

[24] A. Simper. *Correcting general band-to-band misregistrations.* In Proceedings of 3rd IEEE International Conference on Image Processing, volume 1, pages 597–600, 9 1996.

[25] G. P. Penney, J. Weese, J. A. Little, P. Desmedt, D. L. G. Hill, and D. J. hawkes. *A comparison of similarity measures for use in 2-D-3-D medical image registration.* IEEE Transactions on Medical Imaging, 17(4):586–595, 8 1998.

[26] P. Thevenaz and M. Unser. *Optimization of mutual information for multiresolution image registration.* IEEE Transactions on Image Processing, 9(12):2083–2099, 12 2000.

[27] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. *Mutual-information-based registration of medical images: a survey.* IEEE Transactions on Medical Imaging, 22(8):986–1004, 8 2003.

[28] J. Magarey and N. Kingsbury. *Motion estimation using a complex-valued wavelet transform.* IEEE Transactions on Signal Processing, 46(4):1069–1084, 4 1998.

[29] I. W. Selesnick, R. G. Baraniuk, and N. C. Kingsbury. *The dual-tree complex wavelet transform.* IEEE Signal Processing Magazine, 22(6):123–151, 11 2005.

[30] H. Chen and N. Kingsbury. *Efficient Registration of Nonrigid 3-D Bodies.* IEEE Transactions on Image Processing, 21(1):262–272, 1 2012.

[31] U. Kjems, S. C. Strother, J. Anderson, I. Law, and L. K. Hansen. *Enhancing the multivariate signal of [/sup 15/O] water PET studies with a new nonlinear neuroanatomical registration algorithm [MRI application].* IEEE Transactions on Medical Imaging, 18(4):306–319, 4 1999.

[32] T. Gaens, F. Maes, D. Vandermeulen, and P. Suetens. *Non-rigid multimodal image registration using mutual information.* In William M. Wells, Alan Colchester, and Scott Delp, editors, Medical Image Computing and Computer-Assisted Intervention — MICCAI'98, pages 1099–1106, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[33] C. Jailin and S. Roux. *Dynamic Tomographic Reconstruction of Deforming Volumes.* Materials, 11(8), 2018.

[34] V. Van Nieuwenhove, J. De Beenhouwer, J. Vlassenbroeck, M. Brennan, and J. Sijbers. *MoVIT: a tomographic reconstruction framework for 4D-CT.* Optics Express, 25:19236–19250, 8 2017.

[35] Q. Xu, X. Mou, S. Tang, and Y. Zhang. *CT reconstruction based on improved total variation minimization.* In IEEE Nuclear Science Symposuim Medical Imaging Conference, pages 3241–3244, 10 2010.

[36] E. Y. Sidky, C. Kao, and X. Pan. *Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT.* Journal of X-ray Science and Technology, 14(2):119–139, 2006.

[37] E. J. Candes, J. Romberg, and T. Tao. *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information.* IEEE Transactions on Information Theory, 52(2):489–509, 2 2006.

[38] M. Grasmair.

[39] H. Lee, L. Xing, R. Davidi, R. Li, J. Qian, and R. Lee. *Improved compressed sensing-based cone-beam CT reconstruction using adaptive prior image constraints.* Physics in Medicine & Biology, 57(8):2287, 2012.

[40] K. J. Batenburg and J. Sijbers. *DART: A Practical Reconstruction Algorithm for Discrete Tomography.* IEEE Transactions on Image Processing, 20(9):2542–2553, 9 2011.

[41] G. T. Herman and A. Kuba. *Discrete tomography: Foundations, algorithms, and applications.* Springer Science & Business Media, 2012.

[42] Wikipedia, 2018, *Machine learning.*

[43] Y. LeCun, Y. Bengio, and G. Hinton. *Deep learning.* nature, 521(7553):436, 2015.

[44] D. M. Pelt and K. J. Batenburg. *Fast tomographic reconstruction from limited data using artificial neural networks.* IEEE Transactions on Image Processing, 22(12):5238–5251, 2013.

[45] R. Hambli, H. Katerchi, and C. Benhamou. *Multiscale methodology for bone remodelling simulation using coupled finite element and neural network computation.* Biomechanics and modeling in mechanobiology, 10(1):133–145, 2011.

[46] D. M. Pelt, K. J. Batenburg, and J. A. Sethian. *Improving Tomographic Reconstruction from Limited Data Using Mixed-Scale Dense Convolutional Neural Networks.* Journal of Imaging, 4(11), 2018.

[47] J. Adler and O. Öktem. *Learned Primal-Dual Reconstruction.* IEEE Transactions on Medical Imaging, 37(6):1322–1332, 6 2018.

[48] W. Wolfson, 2017, *Husky or Wolf? Using a Black Box Learning Model to Avoid Adoption Errors.*

[49] X. Guo, S. M. Johnston, Y. Qi, G. A. Johnson, and C. T. Badea. *4D micro-CT using fast prospective gating.* Physics in Medicine & Biology, 57(1):257, 2012.

[50] S. Achenbach, T. Giesler, D. Ropers, S. Ulzheimer, H. Derlien, C. Schulte, E. Wenkel, W. Moshage, W. Bautz, W. G. Daniel, W. A. Kalender, and U. Baum. *Detection of coronary artery stenoses by contrast-enhanced, retrospectively electrocardiographically-gated, multislice spiral computed tomography.* Circulation, 103(21):2535–2538, 2001.

[51] A. C. S. Brau, C. T. Wheeler, L. W. Hedlund, and G. A. Johnson. *Fiber-optic stethoscope: A cardiac monitoring and gating system for magnetic resonance microscopy.* Magnetic Resonance in Medicine, 47(2):314–321.

[52] C. J. Ritchie, J. Hsieh, M. F. Gard, J. D. Godwin, Y. Kim, and C. R. Crawford. *Predictive respiratory gating: a new method to reduce motion artifacts on CT scans.* Radiology, 190(3):847–852, 1994. PMID: 8115638.

[53] S. M. Walker, D. A. Schwyn, R. Mokso, M. Wicklein, T. Müller, M. Doube, M. Stampanoni, H. G. Krapp, and G. K. Taylor. *In Vivo Time-Resolved Microtomography Reveals the Mechanics of the Blowfly Flight Motor.* PLOS Biology, 12(3):1–12, 3 2014.

[54] J. Liu, X. Zhang, X. Zhang, H. Zhao, Y. Gao, D. Thomas, D. A. Low, and H. Gao. *5D respiratory motion model based image reconstruction algorithm for 4D cone-beam computed tomography.* Inverse Problems, 31(11):115007, 2015.

[55] M. Dawood, N. Lang, X. Jiang, and K. P. Schafers. *Lung motion correction on respiratory gated 3-D PET/CT images.* IEEE Transactions on Medical Imaging, 25(4):476–485, 4 2006.

4

Weighted back projection

Reconstruction when the location of changes is known

To improve the reconstructions of 4D μCT scans, we developed weighted back projection. Weighted back projection is a technique that combines an initial volume (section 3.2) with a weight volume and is performed in the context of an iterative reconstruction (section 2.4.2).
Weighted back projection is used in the context of 4D μCT, when two types of prior knowledge are available:

- An initial volume representing the very same sample which is studied using 4D μCT, but without the dynamic process occurring. This can be obtained from a scan before the process was initiated or after it has concluded, or it can be a model which represents the real sample.

- Knowledge about which areas in the sample might change (in comparison to the initial volume) due to the dynamic process.

In weighted back projection, the prior knowledge of an initial volume is extended by the knowledge on where the dynamic process is most likely to occur in this volume, i.e. which voxels j of the initial volume will have the largest $|\mu_{j,measured} - \mu_{j,init}|$. Weighted back projection typically starts from a high-quality static scan, wherein the voxels that will be subject to the dynamic change are identified, and focusses on those voxels for reconstructing the dynamic scan. Without this prior knowledge, weighted back projection is not possible.
Other techniques have used this type of prior knowledge before, such as a method presented by Myers *et al.* [1] and region based 4D tomographic reconstruction [2]. In both these, the static regions are enforced to not deviate from their initial value. In Myers' method [1], the prior knowledge of the static regions is supplemented with prior knowledge about the physics of the dynamic process and the fact that only two attenuation coefficients are possible, i.e. the studied materials are incompressible and therefore do not change in attenuation coefficient. All this prior knowledge is implemented as corrections to the reconstruction volumes in between the iterations of an iterative reconstruction and after the reconstruction completes.
In region based reconstruction [2], two separate SIRT reconstructions are performed per iteration. One, called the static reconstruction, uses the complete projection data, the other uses a small fraction around the time which is being reconstructed and is performed for each time step. Both reconstructions in region based reconstruction [2], static and dynamic, reconstruct the entire volume, i.e. the static reconstruction also reconstructs the dynamic region and vice versa. The resulting volume for one time step uses as attenuation coefficients for the static regions those obtained by the first reconstruction and as attenuation coefficients for the dynamic regions those obtained by the second one. Therefore, in this technique, similarly to Myers *et al.* [1], the

prior knowledge of static and dynamic regions is incorporated in between iterations and at the end.
In reconstruction algorithms that use prior knowledge, the influence of this prior knowledge on the final resulting volume is of importance. If this influence is too strong, small errors in the prior knowledge could propagate to wrong results that might be hard to detect. Therefore, any algorithm must be able to robustly deal with at least a small margin of error in the prior knowledge. Weighted back projection can handle small errors in the prior knowledge, while similar 4D μCT techniques require the prior knowledge to be exactly correct. In both previously described techniques, the static regions are enforced not to deviate from their initial value. In contrast, the static regions can still deviate in weighted back projection, allowing for error correction in the initial volume.
Weighted back projection works by assigning a weight w_j to every voxel j: high for dynamic regions and low for static regions. The voxels expected to change with respect to their value in the initial volume will receive a high w_j, while those that are expected to remain static receive a w_j closer to zero. This results in a faster convergence to a correct volume and therefore fewer projections needed for one 3D reconstruction, granted that the w_j are chosen wisely.
This weight is included in the back projection step of an iterative reconstruction, instead of in between iterations as in most 4D μCT techniques, and assures most of the correction is assigned to dynamic voxels. Static voxels should already have their (almost) correct value from the initial volume, which means all changes to this initial volume in the update step should be due to the dynamic process which is being studied. Since the prior knowledge is incorporated in the reconstruction algorithm itself, less iterations are required to fully use it.
The simplest way to determine the w_j in samples where the dynamic region shares a single μ, including the sample we will use here (described in section 2.8), is based on the attenuation coefficients of an input volume (possibly, but not necessarily the initial volume) with the assistance of a Gauss function. This assures that the slightly different attenuation coefficients in a reconstruction (since the input volume is probably a reconstruction from another scan, reconstructed attenuation coefficients from the same material can have differences due to noise and other imaging artefacts) do not have widely different w_j. This also means there is a wide range of possible w_j.
If there are only two w_j possible, 0 and 1, weighted back projection becomes similar to region based iterative reconstruction [2] or Myers' technique [1]. As shown in Van Eyndhoven *et al.*[2], this already provides a tremendous improvement for 4D μCT. When extending this to more possible weights, there is more room for error upon determining the w_j and there is no requirement

for a segmentation step [3]. Weighted back projection can be combined with other techniques such as compressed sensing or using other prior knowledge [4, 5].
Dividing a reconstruction volume in a static and a dynamic region can also happen without prior knowledge on these regions. Part of the MOCDOR framework [6] is to iteratively determine the changing regions within a certain projection time range. Note that, even though Kumar [6] uses the terminology 'motion model', their framework is about dynamic regions instead of motion registration (section 3.3). This subtle distinction in terminology is important when comparing techniques. MOCDOR combines these dynamic regions with a certain acquisition scheme for the projections, a probability maximisation algorithm and some smoothing filters.
Both weighted back projection and region based iterative reconstruction can only be used when the volume has enough static regions. Therefore it is best to use different techniques (for example motion registration, as in section 3.3) when the dynamic process consists of movement.

4.1 The reconstruction algorithm

Weighted back projection is an adapted version of the 3D iterative reconstruction for computed tomography. In this study, we use SART, but any other iterative reconstruction algorithm can be adapted in the same way to result in weighted back projection. We first explain the 3D reconstruction and subsequently this can be integrated in a 4D reconstruction.
We assume that the initial volume is a good approximation of the sample in the static regions of the sample and hence most of the back projected information in the update step physically originates from the dynamic regions. Therefore, weighted back projection back projects mostly in the dynamic regions, by using the prior knowledge on where these regions occur in the sample.
To achieve this, an additional volume is introduced into the reconstruction process: the weight volume. This volume has the same amount and configuration of voxels as the reconstruction volume, although it may be stored as a different data type. It would be possible to create a lower resolution weight volume, i.e. with a lower amount of voxels, and use interpolation to get voxel values at higher resolution. This approach was not tested in this manuscript, but it is a promising method to decrease the memory load.
Each voxel j in the reconstruction volume therefore has a corresponding voxel value w_j in the weight volume. Voxels that are present in a dynamic region should have a high w_j and voxels in a static region should have a low w_j. The exact values for these will be discussed below.
The iterative reconstruction now starts with an initial volume. Ideally, the

static regions from the prior knowledge will already have the correct attenuation coefficient μ_j in this initial volume. In theory, all changes $p_i - q_i$ that are back projected will be a result of changes in the dynamic regions due to the dynamic process. However, in reality these changes also contain contributions from noise in both dynamic and static regions and potentially other imaging artefacts. The dynamic regions are those voxels where we expect the dynamic process to occur, while the static region is its complement.
We want the correction $p_i - q_i$ to be back projected mainly in the dynamic voxels, instead of evenly over dynamic and static voxels as is the case in a regular iterative reconstruction. The weight factors w_j will ensure a larger part of the correction is distributed to those voxels with a larger weight, i.e. that are known to be more likely to be dynamic.
The weight volume is used during the back projection step of an iterative reconstruction algorithm (section 2.4.2), equation 2.64. The back projection step without a weight volume is given by the following equation, which is simply equation 2.64.

$$\mu_j^k = \mu_j^{k-1} + \alpha \frac{p_i - q_i}{L_i} \tag{4.1}$$

We propose to add a weight volume to transform this equation into:

$$\mu_j^k = \mu_j^{k-1} + \alpha \frac{w_j}{W_i} \frac{p_i - q_i}{L_i} \tag{4.2}$$

W_i is the normalisation factor, which depends on the X-ray i. When $W_i < \epsilon$, the attenuation coefficients are unchanged in the implementation, i.e. $\mu_j^k = \mu_j^{k-1}$. ϵ is a very small number determined by the precision of the used data type (unsigned 8-bit int, 32-bit float,...). This is to prevent infinities resulting from division by 0. W_i ensures that the same total amount $p_i - q_i$ is back projected over the complete X-ray, compared to a back projection in a conventional iterative reconstruction, where all weights are 1. It is defined by the following equation:

$$W_i = \frac{\sum_{j \in X_i} w_j}{\sum_{j \in X_i} 1} \tag{4.3}$$

To demonstrate that the total amount back projected remains unchanged, we calculate the sum along an X-ray i of the back projected values with a weight volume:

$$\sum_{j \in X_i} \left(\frac{w_j}{W_i} \frac{p_i - q_i}{L_i} \right) = \frac{p_i - q_i}{L_i} \frac{1}{W_i} \sum_{j \in X_i} (w_j) \tag{4.4}$$

And the same sum without a weight volume:

$$\sum_{j \in X_i} \left(\frac{p_i - q_i}{L_i} \right) = \frac{p_i - q_i}{L_i} \sum_{j \in X_i} (1) \tag{4.5}$$

In both these, we removed any factor not depending on j from the sum over j. These two summations should be equal, which can only be true if equation 4.3 holds.
The consequence of using equation 4.2 as the back projection step in an iterative reconstruction is that the voxels with a high w_j change more as a result of one back projection than voxels with a low w_j. Over the scope of a full iteration, low w_j voxels will keep a value close to μ_j^0, their value in the initial volume. This behaviour is desirable for static voxels, i.e. those regions in the sample where no dynamic process (section 2.6) takes place.
While the physical dynamic process in the sample will not affect the static voxels, the voxels where we expect the dynamic process to occur should have a high w_j, so they have to opportunity to start deviating from their initial attenuation coefficient μ_j^0. Obviously, voxels in the dynamic region do not necessarily change. For example, a dynamic process might occur only in certain pores of a geological material and not all of them. Since it is not a priori known exactly which pores will change, all of them receive a high w_j.
The benefit of fast convergence and the requirement of a previously acquired initial volume makes the technique perfectly suited for 4D μCT imaging. A weighted back projection for a 4D μCT scan starts by subdividing the radiographs into a number of subsets which may overlap, each spanning at least a complete angular range. Each subset is reconstructed separately with the above described 3D weighted back projection reconstruction algorithm. The combination of these 3D volumes forms the resulting 4D volume.
An additional benefit of weighted back projection is that it can be combined with most other 4D μCT techniques, as long as they are using iterative reconstruction.

4.2 The weight volume

4.2.1 Generic description

The choice of the weight volume has an important influence on the result, as will be discussed in section 4.4. In this work, we consider 4 different cases, but more methods to determine the weight volume are possible. In this study, a weight volume can be continuous (many possible weight values) or discrete (only two possible weight values), and it can allow weights of zero or require all weights to be higher than zero.

All used weight volumes in this work are derived from a high quality reconstruction of an additional scan. The attenuation coefficient of this reconstructed volume at voxel j is μ_j^w and the weight of this voxel is w_j. For the continuous weight volumes, both are connected by:

$$w_j = b + v \cdot \exp\left(-\frac{(\mu_j^w - \mu_{centr})^2}{2\sigma^2}\right) \tag{4.6}$$

This is a Gaussian function [7] with four parameters: b, v, μ_{centr} and σ. This

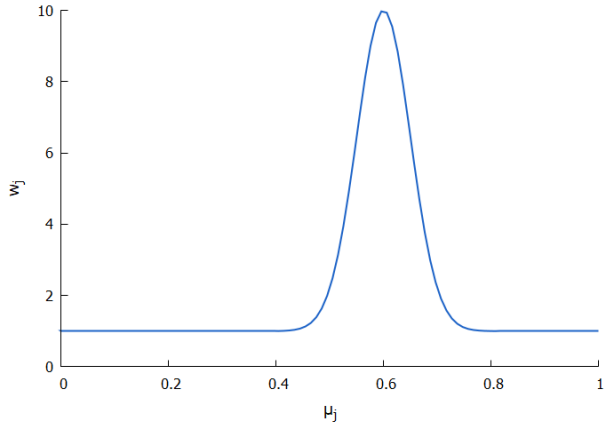


Figure 4.1: Graph respresenting equation 4.6 with $\mu_{centr} = 0.6$, $\sigma = 0.05$, $b = 1$ and $v = 9$.

method of determining the weight volume can be used on the Bentheimer sandstone sample (described in section 2.8) since we know the dynamic process, i.e. the oil displacing brine in the rock, will only occur in the pores. Therefore, all dynamic regions have the same attenuation coefficient μ of the pore space. Figure 4.1 shows an example plot of this function. b is a constant determining a minimum weight for the static voxels. It is put to 0 or 1 depending on whether zero weights are allowed or not. v is a factor determining the height of the peak, μ_{centr} is the central value of the peak and σ determines its width. When used to determine a weight volume, μ_{centr} is the average attenuation coefficient of the dynamic region, i.e. the region inside the initial volume most likely to undergo a dynamic change, σ determines how much μ_j^0 can differ from μ_{centr} while still receiving a high weight and v is a measure of how certain we are that dynamic and static regions are well described by this Gauss. v regulates how much of the back projection ends up in dynamic regions and how much in static regions.

The baseline, when $b = 1$, is an important aspect of this function. It assures no w_j is ever zero. A voxel j with $w_j = 0$ would be unable to change from its value μ_j^0 in the initial volume. Therefore, any mistakes in the initial volume or in the determining of the dynamic region are fixed and can not be corrected. To allow some lenience for these mistakes, b can be set to 1.
This method of determining the weight volume is a very simple segmentation. The dynamic and the static regions are determined based on their attenuation coefficient in the initial volume. This is similar to a single threshold segmentation algorithm ([8], section 6.5), which is the simplest segmentation technique in existence, with the difference that the cut-off for weighted back projection is not 'hard': this function is a Gaussian distribution, instead of a delta function. For the discrete weight volumes, the segmentation will be a single threshold, resulting in two possible weights: w_{high} and w_{low}.
In contrast, region based iterative reconstruction [2] is similar to allowing only two different w_j, 0 or 1. This is a hard division between static and dynamic regions, in which there is no room for error in determining the static region. If any dynamic voxel ends up with $w_j = 0$, it will keep its faulty initial value. While there is less uncertainty allowed, this method provides a simpler weight volume than the continuous weight volume as described above. A voxel can either change, or it can not.
In region based iterative reconstruction [2] the static and dynamic regions are either known beforehand, or they are estimated during the reconstruction iterations with a projection distance minimisation scheme. For the interested reader, Van Eyndhoven *et al.* [2] provide a detailed explanation of this estimation method.
A weight volume can be derived from an initial volume or another representation of the sample with most segmentation methods. Vlassenbroeck [8] lists a number of these methods, which are used for 3D analysis.

4.2.2 Experimental study

We created multiple weight volumes for the experiments on the sample of the Bentheimer sandstone, described in section 2.8. For all of these weight volumes, the starting point was a high resolution reconstruction of a scan of the sample, taken before an attenuation-matching CsCl solution filled the pores. In other words, this is a scan of the sample where the pores are clearly visible, but it is not the initial volume, as described in section 2.8. Figure 4.2 shows a slice of this volume.
Also in figure 4.2 is the porous volume on which a straightforward cylindrical mask has been applied. The cylinder encompassed the sample as close as possible and any voxel outside of it got a grey value of 1, which is white in the visual representation shown.

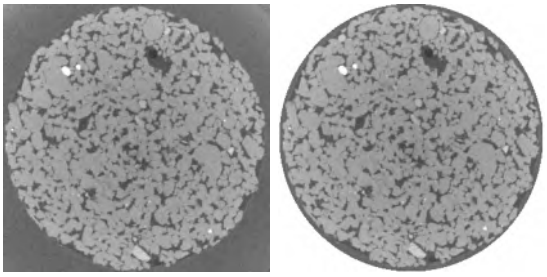


Figure 4.2: The central slice of the reconstruction of the Bentheimer sandstone where the pores were visible. To the right a cylindrical mask was applied to single out the sample from the background viton sleeve.

This masked volume was the starting point for the continuous weighted back

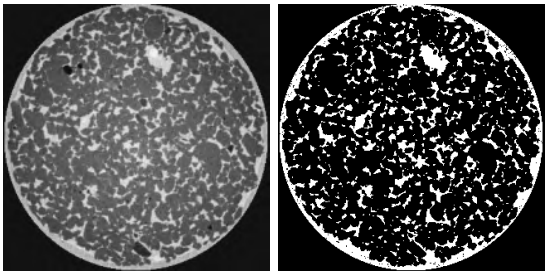


Figure 4.3: Slice of the weight volume created for weighted back projection (left) and a discrete weight volume (right). White areas are the pores, which have higher weight.

projection weight volume, shown in figure 4.3 (left). For this weight volume, we used equation 4.6, with $\mu_{centr} = 0\,/mm$, $\sigma = 0.03\,/mm$, $b = 1$ and $v = 20$. Depending on whether weights of zero were allowed or not, b got a value of respectively 0 or 1. With these parameters, any voxel outside of the cylinder got $w_j = b$, which is the lowest weight.
A weight volume with two discrete w_j (one high and one low) was created from the same starting volume: the cylindrical masked volume represented in figure 4.2. The pores, segmented with a simple threshold on their attenuation coefficient, received $w_j = w_{high}$ while the grey rock structure had $w_j = w_{low}$. The threshold was chosen manually, as the quality of the initial scan was sufficiently high to have a clear distinction between both phases. The resulting weight volume can be found in figure 4.3 (right).
Both these weight volumes can allow $w_j = 0$ (in case of the discrete weight volume, this is $w_{low} = 0$, in the continuous case, $b = 0$) or only allow non-zero

weights ($w_{low} = b = 1$). Both possibilities visually look similar, hence they are not shown in separate figures here.
In the case of the Bentheimer sandstone, we set $\mu_{centr} = 0\,/mm$ and $\sigma = 0.03\,/mm$. These parameters should be determined for each application individually, based on the histogram of the high-quality dataset.
For discrete weight volumes, the weights are either $w_j = 0$ and $w_j = 20$ or $w_j = 1$ and $w_j = 20$. The absolute values 1 and 20 are not important, it is only the ratio between the highest and the lowest weight which influences the results.

4.3 The benchmark volume

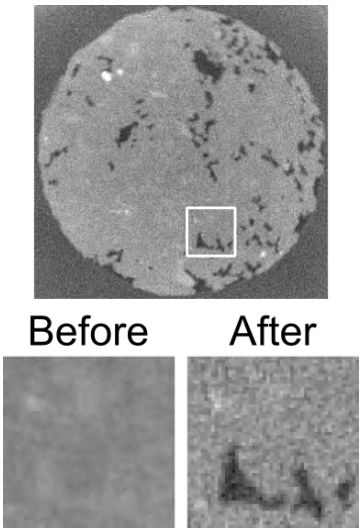


Figure 4.4: The central slice of the benchmark reconstruction with a zoomed-in area focussing on a specific pore shown in the bottom figures. This reconstruction is an averaged reconstruction without prior knowledge using all projections. The zoomed-in area shows the situation of the pore before filling with oil and after filling with oil. The complete slice shows the after-filling situation. The diameter of the slice is 6 mm and the physical size of the zoomed-in region is 0.95 mm×0.95 mm.

All reconstructions in this chapter use one full iteration of the SART algorithm, i.e. use every projection once. We compare the reconstructions with a benchmark volume, for which we reconstruct two static situations, representing an average over the first 8 rotations and the last 8 rotations of

the dynamic scan. As described in section 2.8, the total scan consisted of 79.8 rotations. The projections corresponding to these 8 rotations were averaged for each projection angle and the resulting dataset was reconstructed with conventional SART. The projections of each rotation were shifted 0.096° compared to the previous rotation, which resulted in angular smoothing when averaging over many rotations. This shift is due to rounding errors in the positions of the rotation motor and does not influence a reconstruction, only an averaging such as done here. Such a shift can even be on purpose, since it is beneficial to have data at slightly different angles in subsequent rotations. Multiple rotations can then be combined for one reconstruction, each having additional information.
The number of 8 was chosen to get significant noise reduction while keeping the angular smoothing limited. The pore on which the figures in the following sections zoom in was static in both benchmark reconstructions and got filled with oil in between. The resulting benchmark reconstructions are shown in figure 4.4.
The reconstructions with which weighted back projection was tested used only 1 out of every 10 projections. One time step was therefore reconstructed with 60 projections, uniformly spread over a full rotation, instead of the 600 available per rotation. This simulates a dynamic process that is ten times faster with a 4D-μCT scan that is performed ten times faster (since there is still the same amount of change within one rotation). Since the benchmark uses the full available data, we can compare such reconstructions of such a 'fast' scan with limited available data with a higher quality scan.

4.4 Comparison between weight volumes

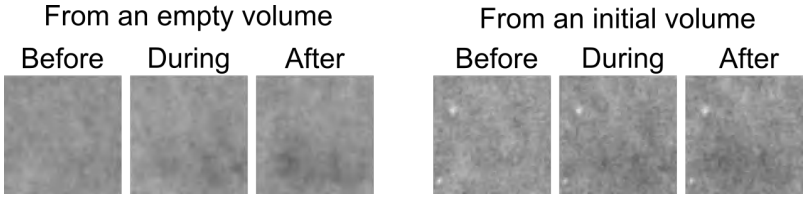


Figure 4.5: Zoom-in of a pore in the central slice of two different reconstructions: the conventional SART reconstruction starting from an empty volume (left) and starting from an initial volume (right) with only $1/10^{\text{th}}$ of available projections used. Three reconstructed time steps are shown for each one: immediately before the filling of this pore with oil, during filling and immediately after. The physical size of the displayed region is $0.95\,\text{mm}\times0.95\,\text{mm}$.

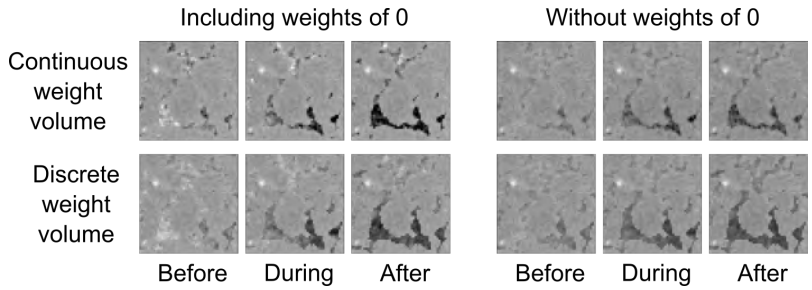


Figure 4.6: Zoom-in of a pore in the central slice of weighted back projection reconstructions with different weight volumes with only $1/10^{\text{th}}$ of available projections used. Three reconstructed time steps are shown for each one: immediately before the filling of this pore with oil, during filling and immediately after. The physical size of the displayed region is $0.95\,\text{mm}\times0.95\,\text{mm}$. The left part of the pore in the before-filling time step shows a lot of noise when including weights of 0.

In this section, we will present the resulting reconstructions when using either of the weight volumes described in section 4.2. Figure 4.4 shows the benchmark reconstruction, while figure 4.5 shows reconstructions using conventional SART. The result of the weighted back projection reconstructions with different weight volumes are shown in figure 4.6, where three time steps are shown for each reconstruction, zoomed in on one large pore with a size of 2292 voxels: immediately before the filling of this pore with oil, during filling and after filling. A quantitative analysis of one pore is displayed in figure 4.7. For this figure, the pore volume was segmented into individual pores using the software octopus analysis (formerly known as Morpho+ [9]) as described in section 6.1. Using the information on which voxels belonged to which pore, we viewed the results within one pore, the same pore visualised in figure 4.4. We calculated the attenuation coefficient μ of a pore by taking the average over all voxels within this pore. Similarly, a standard deviation σ was calculated over all these voxels. The result is a μ and a σ for each pore and for each reconstructed time step. We display, for one dynamic pore, the average over all time steps before this specific pore was invaded by oil (this moment of invasion is called the Haines jump) and after the Haines jump in figure 4.7. The slices show that both the reconstruction from an empty or from an initial volume in figure 4.5, i.e. the reconstructions that did not use a weight volume, fail completely at reconstructing the sample correctly. As visible in figure 4.7, the conventional SART reconstruction and the reconstruction from an initial volume fail to reach the benchmark μ after filling of the pore with oil. It is clear that 60 projections is far too few. However, the weighted back

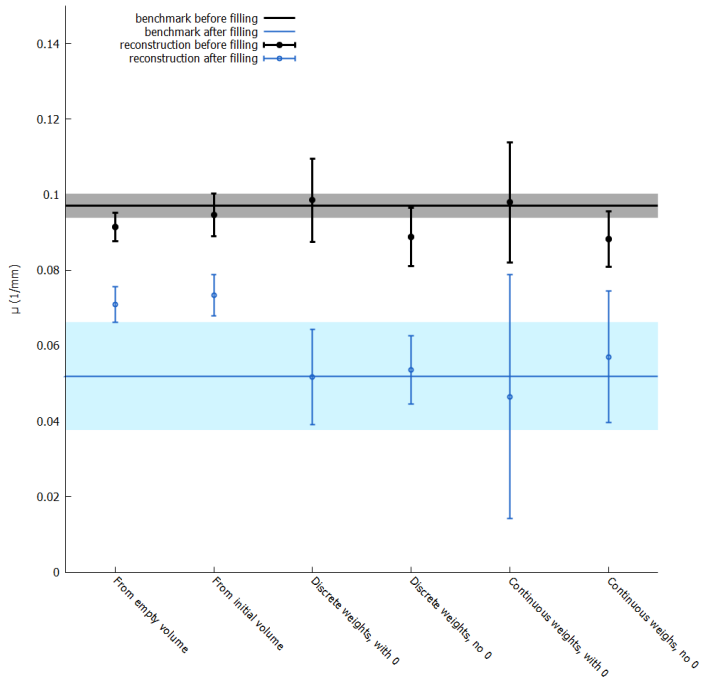


Figure 4.7: The average attenuation coefficient and average standard deviation in a dynamic pore (the bigger pore in the zoomed-in region in figure 4.4) in the reconstructions which used $1/10^{\text{th}}$ of projections. Data from both before the filling with fluid and after are shown. The benchmark μ are shown as a constant function, with the σ on the benchmark reconstruction denoted by the shaded areas around the benchmark constant.

projection reconstructions in figures 4.6 and 4.7 perform well, clearly showing the pore getting filled for all four weight volumes and the attenuation coefficients being close to the benchmark. From visual assessment, there is little difference between discrete or continuous weight volume reconstructions. Similarly, the attenuation coefficients in figure 4.7 are similar, although the noise for a continuous weight volume (the error bars) is larger when considering the graph. The difference between including or excluding $w_j = 0$ is clear: the noise is stronger when including $w_j = 0$, as shown in the size of the error bars. This noise is mainly visible in function of time, when visually scrolling through slices: the grey value of many smaller pores is fluctuating around the attenuation coefficient of the surrounding rock. This is an unphysical effect: these pores are not rapidly decreasing and increasing in attenuation

coefficient. Using piecewise linear fitting [10] (chapter 6), these small pores were found to be static and should therefore remain on their initial attenuation coefficients. This strong noise behaviour is much more nuanced in the weighted back projection reconstructions that used a minimal weight larger than 0.

When performing these reconstructions using the projections generated from

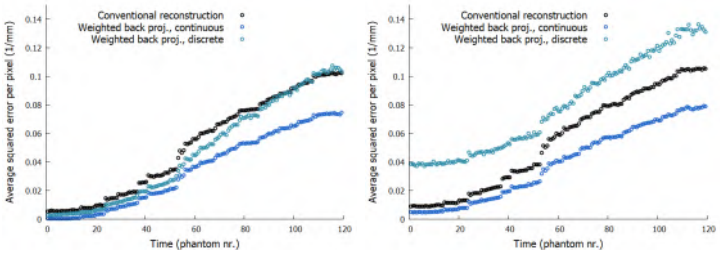


Figure 4.8: The root mean squared pixel error between reconstruction and phantom for reconstructions starting from the simulated projections of the phantoms (described in section 2.8). The left figure corresponds to low simulated noise, the right figure to high noise.

the phantoms (as described in section 2.8), we can calculate the root mean squared error between the ground truth (the phantoms) and the reconstructions. A graph of these mean squared errors is displayed in figure 4.8. A remarkable difference between a continuous and discrete weight volume is visible: a discrete weight volume suffers more from noise, even resulting in a higher mean squared error than the conventional method when reconstructing from high-noise projections. The continuous weight volume, on the other hand, has a mean squared error that remains below the mean squared error of a conventional reconstruction. In general, all errors increase with time. This can be explained due to the increasing number of small features (filled pores) that should be reconstructed, rendering the reconstruction with any technique increasingly difficult.

The high noise is caused by the nature of weighted back projection: the pores, which have a higher weight than the surrounding rock, receive the high amount of noise that would otherwise be spread out over a much higher amount of voxels. This amount of voxels is the voxels in the complete transmission length in the case of conventional SART reconstructions, or the voxels in the complete transmission length with different contributions from pores or non-pores in case of weighted back projections where $w_j > 0$. When $w_j = 0$ is possible, the amount of voxels is smallest, explaining why the noise in the pores is the most pronounced in these reconstructions.

This higher noise in the dynamic regions is the largest disadvantage of the

weighted back projection technique. It is stronger when zero weights are allowed, since the static regions in that case cannot act as a buffer for some of this back projected noise. At the other hand, not only the propagation of noise is stronger in weighted back projection, but also the signal itself. This is why the filled pores are clearly visible in the weighted back projection reconstructions when using only a limited amount of projections.

4.5 Effects of errors in prior knowledge

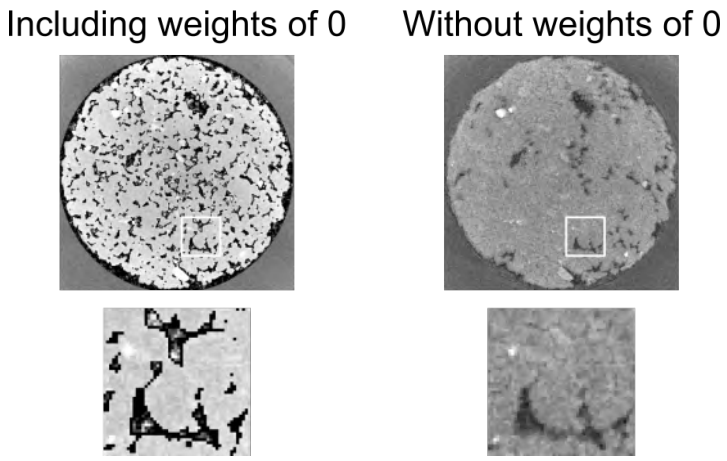


Figure 4.9: Weighted back projection reconstructions starting from a wrong initial volume using a weight volume where $w_j = 0$ is possible (left) and another where $w_j > 0$ for all voxels j (right). These reconstructions used all available projections, i.e. 600 (the full amount in one rotation) per time step. The shown time step is after the filling of the chosen pore with oil. All shown slices and zoom-ins are scaled equally. A benchmark reconstruction for comparison can be found in figure 4.4.

In general, reconstruction methods which incorporate prior knowledge are susceptible to errors in this prior knowledge, such as noise and imaging artefacts. In this case, weighted back projection methods, particularly with zero weights, result in strongly deteriorated reconstruction quality when the static regions of the initial volume, i.e. the voxels where $w_j = 0$, are wrong. To evaluate this, we consider an initial volume in which the attenuation coefficients are about 33% higher than they are in the reconstructions of the dynamic scan. This can occur due to different filtering, beam fluctuations or a bad normalisation. It may not always be obvious at first sight that this initial

volume is wrong, which is why we want a reconstruction algorithm robust enough to handle this, as said in the introduction.
As visible in the figure 4.9, allowing zero weights in this case renders the reconstruction useless: the dynamic pores at the bottom of the zoomed-in region are shown as filled, but so is every other pore in the volume, including those that should have remained on their initial value. On the other hand, when the static regions still have a low, non-zero weight, the reconstruction can handle a small error in the prior knowledge.
We can conclude, so far, that weight volumes without zero weights, both discrete or continuous, provide the best reconstruction of the four possibilities. Zero weights, while useful in theory, produce too much noise to safely discern the real change to the initial volume, even when there are no mistakes in the initial volume.

4.6 Dynamic relaxation factor

Weighted back projection causes a noise increase in the dynamic regions, which is a disadvantage of the technique. Additionally, the dynamic regions are precisely those of interest: a 4D μCT scan is performed to study the dynamic process taking place in the sample, which is why it is of such great importance to correctly reconstruct the dynamic regions.
The idea of weighted back projection is to back project the difference between simulated and measured projection values only to those voxels that can cause this difference: the voxels in a dynamic region. However, this back projected difference is not only caused by real attenuation changes, i.e. the dynamic process, but also by noise. Back projecting this difference over a lower amount of voxels causes these voxels to suffer more from noise than they would have in a conventional reconstruction.
In this section, some approaches to decrease the noise are investigated. Two methods are often used to combat noise in a reconstruction: noise filters and the relaxation factor α. The second one is inherent to iterative reconstruction and described in section 2.4.2, in equation 2.64. α is a number between 0 and 1. A lower relaxation factor leads to less noise and a slower convergence.
Another method tried here is the use of a dynamic relaxation factor. Instead of using a relaxation factor α that is fixed for the entire reconstruction, the relaxation factor will depend on the X-ray i that is being back projected. More specifically, we want α to be lower when the X-ray passes through few dynamic voxels and higher when it passes many of them. Therefore, the total relaxation factor will consist of two parts, α_i and α_{fixed} as follows:

$$\alpha = \alpha_i \cdot \alpha_{fixed} \tag{4.7}$$

with α_{fixed} a constant and α_i a number between 0 and 1 that depends on the X-ray i. The back projection is therefore slightly adapted from equation 4.2:

$$\mu_j^k = \mu_j^{k-1} + \alpha_i \cdot \alpha_{fixed} \frac{w_j}{W_i} \frac{p_i - q_i}{L_i} \tag{4.8}$$

Since α_i should depend on the amount of dynamic voxels traversed, it is correlated with W_i and we should take care to not negate the normalisation effect of W_i. A high W_i means many dynamic voxels are present in this X-ray, so α_i may be high as well. The opposite is true for a low W_i.
The simplest relation between α_i and W_i that matches the above description is linear. However, since the back projection contains $\frac{\alpha_i}{W_i}$, a simply linear relationship such as $\alpha_i = k \cdot W_i$ (with k some constant number) would cancel out the normalising effect of W_i. Therefore, we will use a slightly more complicated version of α_i, which is still not optimised since this is a proof of concept. This simple version is used to investigate the feasibility of the dynamic relaxation factor. α_i will follow the next equations:

$$\alpha_i = 1 \text{ if } W_i > c \tag{4.9}$$

$$\alpha_i = \frac{W_i}{c} \text{ if } W_i \leq c \tag{4.10}$$

This is the linear equation with the addendum that there will be no lowering of the relaxation factor when the cumulative weight on X-ray i is high enough. This only affects the relaxation factor when $W_i \leq c$, i.e. when the X-ray passes through a relatively static X-ray. The constant c is a rough estimate of a dynamic X-ray, i.e. containing many dynamic voxels:

$$c = (1 + b) w_{min} N_{voxels} \tag{4.11}$$

with w_{min} the minimum possible weight (we use a weight volume with $w_{min} > 0$), N_{voxels} the number of voxels along one side of the volume cube, as an estimation of the average X-ray length, and b the fraction of these voxels that we want to be dynamic, i.e. have a higher weight than w_{min}, to label this a 'dynamic X-ray'. In the case of the Bentheimer sandstone and similar samples, b will be related to the porosity, i.e. the fraction of pore volume with respect to the total sample volume. We set b to 0.3. This number was determined by trying out several different b and picking the one which gave the best result. As mentioned before, this is an extremely simple formula for α_i and, in extension, c, because this is a proof of concept test to gauge the ability of a dynamic relaxation factor to improve the weighted back projection method. In addition, b is set to an ad hoc number for the same reason. After determining the feasibility of the technique, we want to determine the

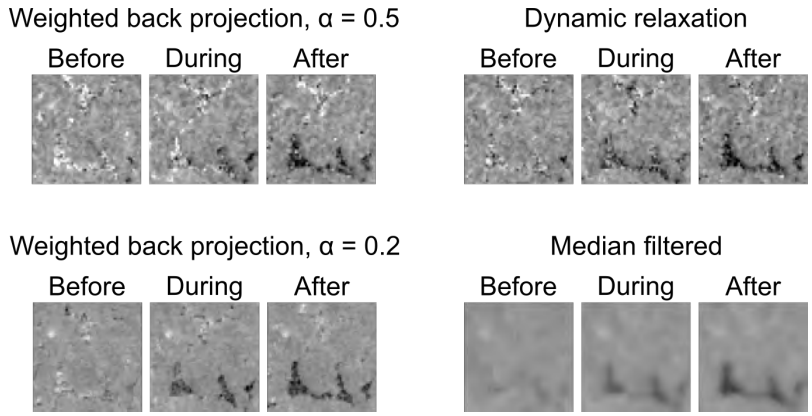


Figure 4.10: Zoom-ins of a pore on the slices of weighted back projection reconstructions using all available projections (600 per rotation) with different options to reduce the noise, including dynamic relaxation. The benchmark reconstructions can be found in figure 4.4.

best value for any application.
Dynamic relaxation is an addition to weighted back projection which adapts the update step in an iterative reconstruction based on the weight volume, similarly to weighted back projection itself. The adaptation to the update step is slightly different compared to weighted back projection, which can hopefully provide a result with less noise.
Figure 4.10 shows zoom-ins of slices of the results. In this figure, the weighted back projection is compared to a dynamic relaxation reconstruction, a weighted back projection with a lower relaxation factor and a weighted back projection on which a simple median noise filter was applied.
While the dynamic relaxation provides a small improvement in the noise level of the pores, simply selecting a lower, static relaxation factor achieves a better result. The noise lowering of the relaxation factor is also demonstrated in figure 4.11: a lower α decreases the noise within one pore. The median filter decreases the noise, but blurs the pores. A more complex noise filter will provide better results.
In conclusion, we recommend to use a lower relaxation factor for the weighted back projection than would be used for a conventional reconstruction, to use a noise filter which is more tailored to the specific sample compared to the demonstrated median filter or to use piecewise linear fitting as explained in chapter 5. This approach will keep the noise in the small dynamic regions adequately small.

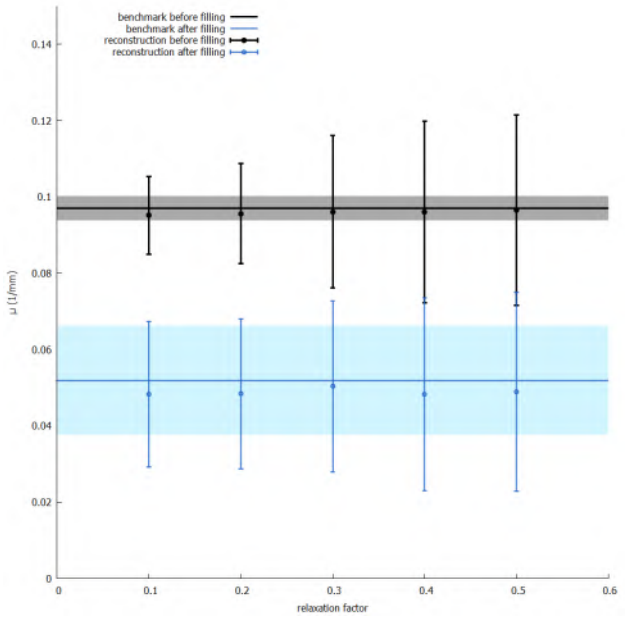


Figure 4.11: The average attenuation coefficient and average standard deviation in a dynamic pore (the bigger pore in the zoomed-in region in figure 4.4) in the reconstructions which used all available projections, using different relaxation factors. Data from both before the filling with fluid and after are shown. The benchmark μ are shown as a constant function, with the σ on the benchmark reconstruction denoted by the shaded areas around the benchmark constant.

4.7 Back projected weights

Back projected weights is a reconstruction technique which is an adapted version of the weighted back projection as described earlier in this chapter. In this technique, the weights from the weight volume in back projected weights can be changed before the reconstruction, instead of being determined solely from prior knowledge, as in weighted back projection.
We can estimate which X-rays pass through only static voxels and lower the weights on these X-rays. The result should be a lower amount of dynamic voxels (voxels with a high weight) and therefore the reconstruction can back project the dynamic changes in a more focussed region. In short, we use an iterative approach to improve the weight volume, in order to strengthen the weighted back projection. This is possibly at the cost of more noise, as we

expect a weight volume with fewer dynamic regions to suffer more from image noise.
In back projected weights, an additional step is added in the reconstruction. First an initial volume and a weight volume are determined, then comes the additional step: the weight tuning, which can be seen as the fine-tuning of the weight volume. Finally, a weighted back projection reconstruction is performed, exactly as explained earlier in this chapter.
The weight tuning starts from an initial volume and an initial weight volume and tunes the weights based on the first steps in a 'normal' iterative reconstruction, using the initial volume as the volume of attenuation coefficients. It performs the first steps of an iterative reconstruction for a chosen set of projections from the data: the simulation of a projection of the initial volume and the calculation of a correction based on the measured radiographs, called C in the following. The actual back projection step, however, is adapted.
The algorithm needs a certain threshold variable, which is called g. We will discuss a good choice for the value of g later in this section. With this variable and the correction factor C (as in equation 2.64) back projected weights fine-tunes the weight volume:

$$C = \frac{p_i - q_i}{L_i} \tag{4.12}$$

$$f = 0.999 + 0.001 \exp(-\exp(1 - \frac{C}{g})) \tag{4.13}$$

$$w_i^k = f \cdot w_i^{k-1} \text{ if } f < 0.9998 \tag{4.14}$$

$$w_i^k = w_i^{k-1} \text{ if } f \geq 0.9998 \tag{4.15}$$

The numbers in this equation are determined ad hoc in order to test this technique as a proof of concept. More rigorous determination of the constants will be necessary if the technique proves viable. The w_i^k are assured never to drop below w_{min}, which is 1 for our demonstration dataset. We discussed the possibility of zero weights in section 4.4 and determined a weight volume without zero weights was a better option.
Since f always has a value between 0.999 and 1, the weights will be lowered on an X-ray i with a small correction C, while they will retain their original value on X-rays with high correction factors. The double exponential in the calculation of f results in a fairly 'gentle' rising between 0.999 and 1, as shown in figure 4.12.

We use an estimation of the noise level in the projection images for determining the value of g. An X-ray that went through a set of only static voxels will have a correction factor C that only differs from 0 because of noise. Therefore, the threshold should be bigger for a bigger noise level.
The noise level can be determined by selecting an outer region on the normalised projections, where the sample does not enter the image. Here, all

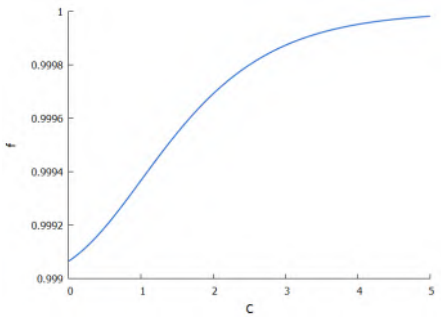


Figure 4.12: Plot of equation 4.13 with $g = 1$.

contributions to the pixels are due to noise, since the air should, normalised on the incoming intensity I_0, theoretically have an attenuation coefficient of 0. The noise level is determined to be the standard deviation σ of a Gaussian distribution fitted to the projection values in this region.
We set $g = 1.35\sigma$. This 1.35 was determined by manually tuning this variable until the resulting weight volume for the Bentheimer sandstone sample seemed reasonable while using 600 projections to determine the weight volume (the entire last rotation, assumed to differ the most from the initial volume). If this proof of concept technique proves viable, a more rigorous approach to determining g will be necessary to apply the technique to more samples.

There is a big disadvantage of back projected weights: the algorithm is not

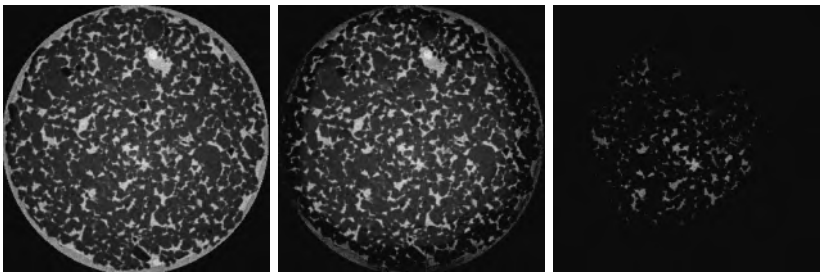


Figure 4.13: Slices of possible weight volumes after the weight back projections. Left: $g = 1.2\sigma$, middle: $g = 1.35\sigma$, right: $g = 1.5\sigma$.

robust. A small change in the parameter g drastically changes the resulting weight volume, as visible in figure 4.13. The left weight volume almost does not differ from the weight volume without using back projected weights. The algorithm did try to lower the weight of the voxels outside of the rock, but

these were already at the minimum because of prior knowledge. The weight volume at the right on the other hand, overestimates the lowering and only left a few pores in the middle of the sample at a non-minimal weight. This does not correspond to the pores that are actually changing, i.e. not all of the pores that receive oil in this scan, are kept at a high weight by back projected weights. It seems mostly the edge weights are lowered, which raises the suspicion that there is an issue with the normalisation. Further research is necessary to investigate this.

The non-robustness of back projected weights is a consequence of the fact

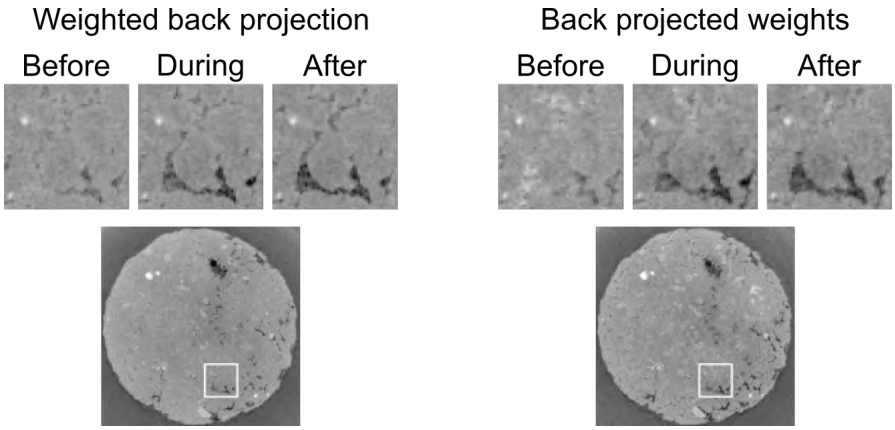


Figure 4.14: Comparison of a weighted back projection and a back projected weights reconstruction when only $1/10^{\text{th}}$ of the projections were used. A zoom-in of a pore before, during and after filling with oil is shown, as well as the full slice during filling with oil.

that the weights can only go down, never up. After enough iterations, all weights that are on even a single X-ray i with $f < 0.9998$ will reach the minimal value. In order to still use a significant amount of projections for determining the weight volume, i.e. use more available information, the minimal factor f is fairly high, i.e. 0.999, so a single X-ray i has little influence. If the weights were allowed to go up as well, the initial information on the weight volume would be eroded and the algorithm would be similar to a conventional reconstruction, with one extra iteration.

The results of a back projected weights reconstruction are shown in figures 4.14 and 4.15. Visually, there is some, but little difference between the weighted back projection and the back projected weights reconstruction. Quantitatively, the reconstruction of an empty pore (i.e. the zoomed-in pore before filling) seems to be closer to the benchmark in back projected weights, while the noise is in between the reconstructions with a continuous and a

discrete weight volume. Overall, both reconstruction techniques result in a similar reconstruction quality.

Since this algorithm needs careful fine-tuning of the threshold value in order

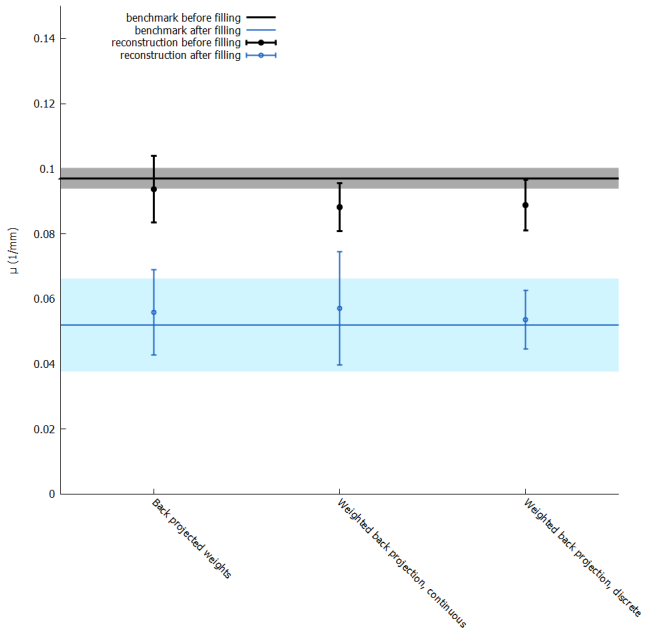


Figure 4.15: The average attenuation coefficient and average standard deviation in a dynamic pore (the bigger pore in the zoomed-in region in figure 4.4) in reconstructions which used $1/10^{\text{th}}$ of projections, both with weighted back projection and back projected weights. Data from both before the filling with fluid and after are shown. The benchmark μ are shown as a constant function, with the σ on the benchmark reconstruction denoted by the shaded areas around the benchmark constant.

to provide a decent weight volume and the resulting reconstruction is not a significant improvement upon standard weighted back projection, we do not advise to use it in its current form.

While back projected weights is not an improvement upon weighted back projection, it is not impossible to automatically estimate the dynamic and stationary regions from the projection data. As demonstrated by Van Eyndhoven *et al.* [2], it is possible to estimate a static (and by its complement, dynamic) region by minimising a measure called *region inconsistency* iteratively. The static region, in this case, is described by a B-spline closed curve

model, so all stationary regions are connected. Anything in between static and dynamic is not possible, so this algorithm is more suited when using a discrete weight volume and it can be used in combination with weighted back projection.

4.8 Conclusion

We presented a new technique to include prior knowledge on dynamic and static regions directly into an iterative reconstruction of 4D μCT. This technique uses an extra volume, the weight volume, which outlines the regions in the sample that are more likely to undergo the dynamic process and slightly adapts the back projection equation. In order to use this information, an initial volume is also required.
The resulting reconstruction, weighted back projection, improves the slices of the Bentheimer sandstone visually by more sharply showing the borders of the pores. Given that only a fraction of the projections ($1/10^{\text{th}}$) were used, the difference between weighted back projection and a conventional reconstruction is striking: a conventional reconstruction can not handle this, but the fast convergence of weighted back projection assures that the reconstructions still show the relevant dynamic changes.
Weighted back projection can be used with any iterative reconstruction. While SIRT theoretically converges, where SART does not, it requires many iterations, over which the influence of the initial volume will decrease slowly. Therefore, when using an initial volume such as in weighted back projection, a technique using a low number of iterations, such as SART, is well suited.
Note that decreasing the number of projections in a continuous scan means that each projection spans a larger angle, thereby increasing the angular smoothing. There will be a certain maximum angle one projection can span to keep a sufficient image quality, given by the spatial resolution, and therefore a minimum number of projections per rotation. Weighted back projection is useful for set-ups where this minimum number of projections is not reached yet. It is also possible to reduce the number of projections without increasing the angular range per projection by using a step-and-shoot mechanism, where the scanner rotates between projections but not during.
There are multiple possible weight volumes, even when the prior knowledge remains the same. Weights of zero can be allowed or disallowed, and weights can be discrete (only two possible values) or continuous. We found that weights of zero should not be used in the weight volume, while there was little difference between the results obtained with a continuous and a discrete weight volume. Theoretically, a continuous range of possible weights allows for the inclusion of uncertainty and a smaller dependence on a perfect segmentation into a static and a dynamic region. Voxels can now have a weight

in between the maximum and minimum, signifying a probability this voxel is dynamic. This advantage of a continuous weight volume is not large enough to generate a significant difference on the tested reconstructions.
Weights of zero should not be allowed, because they completely rule out any errors in both the weight volume and the static component of the initial volume. However, since we work on real data, there will always be artefacts and noise that do cause certain errors.
Weighted back projection can result in very noisy data in the (small) dynamic regions when not paying attention to the relaxation factor and not using a noise filter, since the back projected correction is due to a combination of the dynamic process, originating only from the dynamic regions, and noise, originating from the entire volume. This noise now gets back projected into a smaller region and therefore has a larger influence there. We tested a technique to combat this, dynamic relaxation. However, it turned out that simply using a lower relaxation factor in the iterative reconstruction provided more improvement than dynamic relaxation. Therefore, we advise to use weighted back projection with a lower relaxation factor compared to what a conventional iterative reconstruction would require, or use a noise filter tailored to the specific application.
There is the possibility to improve the weight volume further, after constructing it from the available prior knowledge, by using the projection data. However, the technique we tested for this, back projected weights, was not a good choice. It lowered the weights on 'static' X-rays iteratively. The threshold parameter required careful fine-tuning and the resulting reconstruction did not show significant improvement. Other techniques to improve the weight volume, such as iteratively minimising the *region inconsistency*, will be more robust and therefore a better choice.
The weight volume for the Bentheimer sandstone could be improved further by including the prior knowledge that pores can only be filled when located at the border, or when one of the adjacent pores was filled earlier. Using this knowledge requires extracting a pore network (as in section 6.3) and providing a different weight volume for each time step. After reconstruction of the first time step with a weight volume having only the border-pores as dynamic regions, a new weight volume can be created based on this first reconstruction. This second weight volume will have border-pores and pores neighbouring already filled pores as the dynamic region. Subsequent timesteps are reconstructed similarly. Due to the segmentation step (yielding which pores are filled already) and the re-calculation of the weight volume at each time step, this technique would be computationally more complex. However, the weight volume at each time step will have a smaller dynamic region when compared to the case of using only one weight volume as done in this chapter, so the reconstruction will converge even faster and require even less projections.

An important class of research problems covers dynamic processes occurring within a static grid, which is where weighted back projection is applicable. An example of such a process is fluid flow through the pores of a geomaterial [11]. In contrast, weighted back projection is not suited for dynamic processes which consist mainly of deformation, such as the deformation of foams [12], since in those cases no dynamic regions can be identified. As a future prospect, weighted back projection could be combined with motion registration [13, 14, 15] (see section 3.3), deforming not only the volume itself but the weight volume with it, to include more possible applications.
While weighted back projection was demonstrated on only one dataset in this chapter, the technique has been tried on other samples with similar results. However, for all these samples one was dealing with fluid flow through geomaterials. In general, as the ratio between dynamic regions and static regions decreases, weighted back projection needs less projections to converge and the noise in the dynamic regions increases. In order to further investigate and improve weighted back projection, i.e. evaluate its strengths and weaknesses, future work should investigate weighted back projection on a more diverse set of samples.
In conclusion, weighted back projection is a useful technique for 4D μCT when the necessary prior knowledge is available. When using it, it is best to choose a weight volume without zeroes: if the static region has a weight value higher than zero, noise has the chance to 'dissipate' in the larger static region and it allows room for error correction in the initial volume. Weighted back projection is more likely to suffer from noise in comparison to a regular reconstruction. This can be compensated by using a lower relaxation factor than for a conventional reconstruction.

References

[1] G. R. Myers, A. M. Kingston, T. K. Varslot, M. L. Turner, and A. P. Sheppard. *Dynamic tomography with a priori information.* Appl. Opt., 50(20):3685–3690, 7 2011.

[2] G. Van Eyndhoven, K. J. Batenburg, and J. Sijbers. *Region-Based Iterative Reconstruction of Structurally Changing Objects in CT.* IEEE Transactions on Image Processing, 23(2):909–919, 2 2014.

[3] M. Heyndrickx. *Evaluatie van iteratieve reconstructie in dynamische tomografie door gebruik van initiële oplossingen.* Master's thesis, Ghent University, 2015.

[4] G. R. Myers, A. M. Kingston, T. K. Varslot, M. L. Turner, and A. P. Sheppard. *Dynamic X-ray micro-tomography for real time imaging of*

drainage and imbibition processes at the pore scale. In International Symposium of the Society of Core Analysts, Austin, TX, 18-21 September, pages 1–12, 2011.

[5] G. Van Eyndhoven, K. J. Batenburg, and J. Sijbers. *Region based 4D tomographic image reconstruction: Application to cardiac x-ray CT.* In 2015 IEEE International Conference on Image Processing (ICIP), pages 113–117, 9 2015.

[6] A. S. Kumar. *Computational Methods for Nanoscale X-ray Computed Tomography Image Analysis of Fuel Cell and Battery Materials.* PhD thesis, Carnegie Mellon University, 2017.

[7] Wikipedia, 2018, *Gaussian function.*

[8] J. Vlassenbroeck. *Advances in laboratory-based X-ray microtomography.* PhD thesis, Ghent University, 2009.

[9] L. Brabant, J. Vlassenbroeck, Y. De Witte, V. Cnudde, M. N. Boone, J. Dewanckele, and L. Van Hoorebeke. *Three-Dimensional Analysis of High-Resolution X-Ray Computed Tomography Data with Morpho.* Microscopy and Microanalysis, 17(2):252–263, 2011.

[10] M. Heyndrickx, M. Boone, T. De Schryver, T. Bultreys, W. Goethals, G. Verstraete, V. Vanhoorne, and L. Van Hoorebeke. *Piecewise linear fitting in dynamic micro-CT.* Materials Characterization, 139:259–268, 2018.

[11] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, L. Van Hoorebeke, and V. Cnudde. *Fast laboratory-based micro-computed tomography for pore-scale research: Illustrative experiments and perspectives on the future.* Advances in Water Resources, 95:341–351, 9 2016.

[12] J. A. Elliott, A. H. Windle, J. R. Hobdell, G. Eeckhaut, R. J. Oldman, W. Ludwig, E. Boller, P. Cloetens, and J. Baruchel. *In-situ deformation of an open-cell flexible polyurethane foam characterised by 3D computed microtomography.* Journal of Materials Science, 37(8):1547–1555, 4 2002.

[13] V. Van Nieuwenhove, J. De Beenhouwer, J. Vlassenbroeck, M. Brennan, and J. Sijbers. *MoVIT: a tomographic reconstruction framework for 4D-CT.* Optics Express, 25:19236–19250, 8 2017.

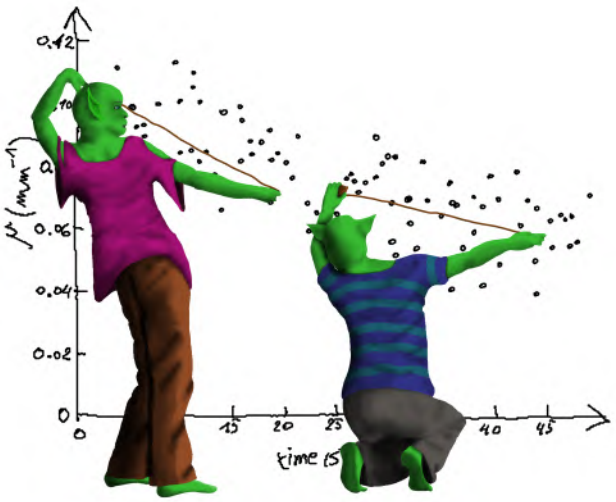
[14] T. De Schryver, M. Dierick, M. Heyndrickx, J. Van Stappen, M. A. Boone, L. Van Hoorebeke, and M. N. Boone. *Motion compensated micro-CT reconstruction for in-situ analysis of dynamic processes.* Scientific Reports, 8:7655, 2018.

[15] M. Odstrcil, M. Holler, J. Raabe, A. Sepe, X. Sheng, S. Vignolini, C. G. Schroer, and M. Guizar-Sicairos. *Ab initio nonrigid X-ray nanotomography.* Nature communications, 10(1):2600, 2019.

5

Piecewise linear fitting

Bridging 4D CT reconstruction and analysis



5.1 Starting situation

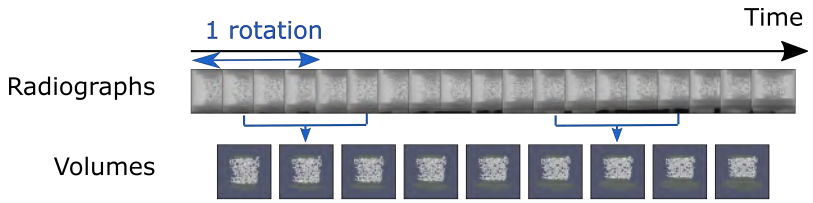


Figure 5.1: A schematic example of a dynamic scan, in this case the compression of an aluminium foam. Each time step or 3D volume is made out of a subset of the total amount of projections.

A dynamic CT scan (section 2.6) is a CT scan taking a long time in which the sample that is being scanned is undergoing changes. Taking a long time, in this case, means performing multiple rotations, even 50 or more. This is somewhat schematically presented in figure 5.1: each projection has both a time coordinate (presenting at which point in time this projection was recorded) and an angle coordinate (presenting at which point in the rotation the scanner was), that are linearly related to each other. There are no two projections with the same time coordinate, disregarding scanner systems with multiple detectors, but because $\alpha \equiv \alpha + n \cdot 360°$ for any $n \in \mathbb{Z}$, there are projections having the same angle coordinates.
After a reconstruction of a dynamic CT scan, the result is generally a series of 3D volumes. An example can be seen in figure 5.1. For clarity, not all of the projections or reconstructed 3D volumes are shown in figure 5.1, but only a representative sample. Each volume represents a specific time point or time range and is created from a set of consecutive projections, which is a subset of the total amount of projections taken during the dynamic scan. When interpreting the result in terms of time ranges, the range of a volume is the time needed from the first used projection to the last. When assigning a specific time point to a volume, the centre of this range is used.
A time range is usually a complete rotation, but it can also be smaller or larger. Time ranges may overlap or be consecutive. In theory, the time ranges are allowed to have gaps between them, but this means that some projections are not used, leaving the reader to wonder why these projections were taken at all. This is therefore not done in practice.
This series of 3D volumes after a reconstruction is the starting point of piecewise linear fitting as described in this chapter and in the article [1]. Several reasons led to the development of piecewise linear fitting in the scope of a 4D μCT reconstruction.

5.2 Motivation

5.2.1 Memory

A 4D μCT scan results in a lot of data. There are the projections themselves, the projection data after preprocessing (normalising, ring filtering, etc.) and of course the result: the reconstruction volumes for each time step.
To conserve computer memory, in this work all of this was saved as 8 bit data (1 byte per pixel). Generally, at least for the first tests, the size of all these datasets was limited. With small sizes, a typical memory size for all of this would be:

1. Projections after preprocessing: 512×512 pixels per projection, 8000 projections. These are taken as 16 bit images, so 2 bytes per pixel. This is $512 \times 512 \times 8000 \times 2 = 4\,194\,304\,000$ bytes or 4.2 GB.

2. Reconstruction volumes: 50 time steps. Each volume being $512 \times 512 \times 512$ voxels. This gives $6\,710\,886\,400$ bytes or approximately 6.7 GB.

This would give a total of 10.9 GB as an estimate of the memory needed for a very memory-gentle saving of the data. Note that it is possible to delete these first data set after use, leaving 6.7 GB of data. The reconstruction volumes are of interest and it is therefore these that we want to retain. Having a few of these dynamic scans quickly demands quite some disk space. Some way to store this data in less memory could therefore be beneficial.

5.2.2 Noise in the temporal dimension

After creating a time series of 3D volumes from a dynamic scan, the data is generally quite noisy, which yields uncertainty in the temporal dimension. In figure 5.2 a three dimensional rendering of one time step is represented. Three of the voxels in this volume have been selected and the time evolution of their grey value is plotted. This serves to illustrate the noise level of the temporal dimension, as a movie of the reconstructions can not be included here due to the restrictions of paper.
It is possible to battle this excess noise with a noise filter. Note that the 'connectedness' in the temporal dimension is probably higher than in the spatial dimensions - meaning that the volume one time step ahead is very likely to bear big resemblance to the volume at hand. A voxel in a 3D volume will also resemble its spatial neighbours, but there exist more small details that are present in just one or a tiny amount of voxels than there are features which appear in just one or two time steps, to disappear again almost directly after they appeared (e.g. these would be noise instead of features).
A simple smoothing filter employed over the temporal dimension will still

have the negative effects of blurring that it has in the spatial dimensions. While we expect features to be bigger in the temporal dimension, they still have edges. An edge will mostly appear due to movement in those dynamic processes where movement is present: water that moves into a voxel, a piece of the object that moves out of the voxel, etc. It is therefore best to pick a noise filter that aims to save edges.
While there are noise filters available in literature (such as an anisotropic diffusion filter with continuous switch [2]) and hence noise reduction was not the primary reason to implement piecewise linear fitting, it did do a good job at reducing the temporal noise too, as will be shown in this chapter.

5.2.3 Analysis

After reconstruction of the sample, we want to analyse the data to find out certain properties of the dynamic process. Since the goal of the analysis is different for each sample, the technique used is not a one-fits-all, even for analysing static 3D samples. With conventional 3D analysis methods, this is cumbersome, as the analysis has to be repeated for each time step and is computationally intensive. It involves manual work, fine-tuning the parameters for a specific application, that can not always be easily scripted [3].
A method to analyse a sample in one session - i.e. analyse the entire 4D volume instead of one 3D volume at a time - would certainly be beneficial. Since each sample and process is different and especially since the particular information we want to get from the sample or process is different, we cannot hope for an automatic analysis method applicable to each 4D volume. Therefore we will assume there is still some manual work needed for each sample. However, we want to minimise this manual work. Performing only one analysis on the entire sample, instead of combining multiple analyses from each time step, would therefore be a huge step forward. Piecewise linear fitting will aim to do just this for a wide variety of samples and processes.
Chapter 6 explains the analysis resulting from a piecewise linear fit in more detail.

5.3 Technique

Consider one 3D-voxel of a reconstructed dynamic scan. This voxel has a grey value for each reconstructed time step. Therefore its grey value can be plotted as a function of the time, as is done for three voxels in figure 5.2. Neighbouring points in this plot should have a high correlation to each other, as explained in section 5.2.2. Therefore it is possible to fit a function to this plot, a function that should reflect the real evolution of the attenuation coefficient of a voxel during the CT scan.

5.3.1 A suitable function

For determining a suitable function to fit, we consider the possible physical explanations for a certain grey value behaviour. Remember that a grey value represents a linear attenuation coefficient, which is determined by the material present and its density (section 2.3.1). The dependence on the density is linear. The dependence on the Z-value (the material) is a more complex function, including discontinuous jumps (absorption edges) as seen in figure 2.4. While the attenuation coefficient also depends on the energy of the X-rays, it is not considered here because the X-ray beam is assumed to remain unchanged during the CT scan. It will therefore not contribute to any change in grey value within one dynamic scan.
Possible behaviours of the attenuation coefficient in a voxel are:

- Nothing changes: the grey value should be a constant function. The CT scan is static (instead of dynamic) if all voxels are like this.

- The material in a voxel changes due to movement. This means that a certain material is moving in or out of the voxel under consideration. An example of this is fluid flow through rock: a pore voxel might have air in it at first and subsequently fluid as it moves through a pore. The attenuation coefficient will therefore change from that of air to that of the fluid and the change will be abrupt. The grey value will be a constant at first and a different constant later, with a discontinuity at the time the movement reaches this voxel. This is a piecewise constant function. If the partial volume effect (section 2.5.6) is significant - i.e. if there are distinct time steps where the voxel is filled partially with air and partially with water - values in between these two constants may be reached and the exact form of the function depends on how the voxel is filled.
  If the sample breaks, this is also a form of movement, where two parts of the sample that were previously stuck together now move apart while air (or whatever the surrounding gas or fluid is) moves in to fill the gap.

- The material changes due to a chemical reaction. A chemical reaction only changes the molecules involved. The atoms and their ratio are preserved [4]. Since μ, the attenuation coefficient, depends on the atomic number Z of the material components and on their ratio, but not on in which specific molecules they are bound [5], μ will not change in this case. It is a constant function.

- The material changes due to a nuclear reaction. In this case, Z of the participating molecules does change and therefore the attenuation coefficient does too.

Nuclear reactions are unlikely to occur in a CT sample. This is fortunate, since the evolution could not be easily described, but if the chosen function can still accommodate for this as well, that would be positive.

- The density changes. This can be due to a number of reasons, such as crystallization [6], temperature changes (material expands or contracts upon heating up [7]; note that this density change is combined with a movement at the outer region of the expanding object) or pressure changes (although this effect is small in solids and liquids [8]). Phase changes, such as melting, change the density radically.

- The concentration changes. An example of this is salt that is being added to water. Since the composition of the water changes, its attenuation coefficient changes as well.

The above changes that occur to the *real* attenuation coefficient are exactly the changes that the dynamic CT scan is supposed to reveal. On top of this, there are changes to the *measured* attenuation coefficient that are unwanted, such as noise or limited angle artefacts (see section 2.5.3). The function that is going to be fitted on the data should reflect the real changes as described above and neglect the false changes of noise and artefacts.
The chosen function is a piecewise linear function. This is a line (i.e. linear), with any slope and offset, that ends in a specific x-point, the breakpoint. From this breakpoint, a line with a new slope and offset starts until the next breakpoint, where another one starts. The function does not need to be continuous at the breakpoints. In theory the number of breakpoints is unlimited, but in practice it is bound by the number of data points we have to fit this function to.
A piecewise linear function may be used to approximate any function [9]. This is a great property, because we do not know the exact function the attenuation coefficient will follow. It is discontinuous, but only at specific points (the breakpoints), as was required by the 'changes due to movement'. The linear part will not allow for the sudden, drastic changes that noise induces, because the breakpoints will have to be a minimum distance apart, which is at least two fitting points.
Note that if the dynamic behaviour of the sample is known, a less general function that fits the specific sample better is preferable. For example, in [10], a piecewise constant function is used with three parts, in which the first and the third part have the same value. This will represent fewer samples, but the samples that can be described by this function, will be better reconstructed by using the more precise function, instead of a general piecewise linear function.

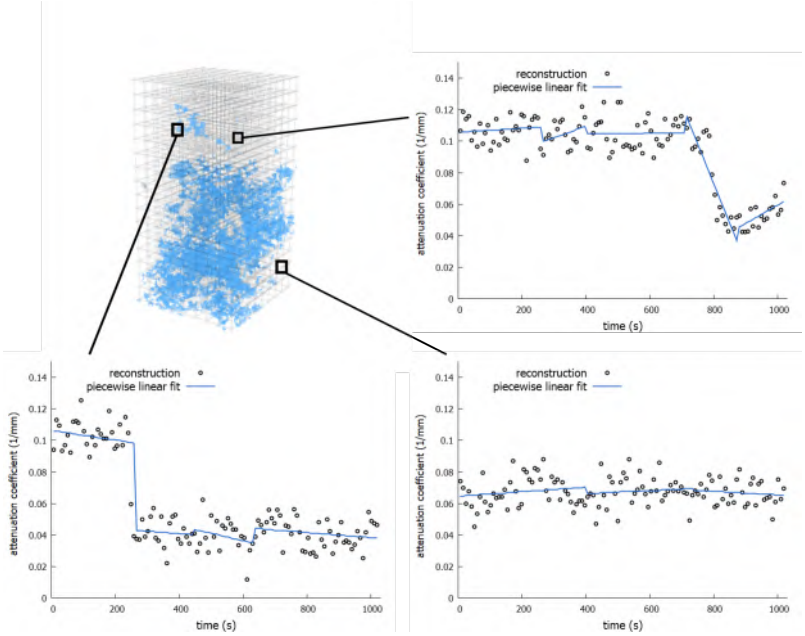


Figure 5.2: Top left: a 3D rendering of the pores in a Bentheimer sandstone. The temporal evolution of three separate voxels is plotted. The graphs depict their grey value μ in function of time t, along with a piecewise linear fit to this.

5.3.2 Fitting to data

Suppose there is data $\mu_i(t)$, meaning that for each time point t, there exists a grey value μ_i for voxel i. This is the data of *one* voxel in the reconstructed volumes. It is this data on which a fitted piecewise linear function will be fitted. Both this data and its fit are depicted for 3 separate voxels in figure 5.2. Note that the following fitting procedure should be repeated for each voxel in order to result in a complete 4D volume.
The fitting procedure should be translation-invariant, i.e. not influenced by the order at which the individual data points are visited. This is the case for the following procedure.
The fitting procedure consists of two main steps:

1. Determine which t are breakpoints

2. Use linear regression to determine the slope and offset of the lines between each two breakpoints

Since the resulting piecewise linear function does not need to be continuous, we can use linear regression on the data only present between two breakpoints, i.e. independent of adjacent data. The linear regression is a well known function [11]:

$$s = \frac{\overline{t\mu} - \overline{t} \cdot \overline{\mu}}{\overline{t^2} - \overline{t}^2} \tag{5.1}$$

$$f = \overline{\mu} - s \cdot \overline{t} \tag{5.2}$$

with the overline indicating an average over the data points in the time dimension. In this set of equations s is the resulting slope, f is the offset, t is the independent variable (in this case time) and μ is the time dependent variable (in this case attenuation coefficients). Every voxel in the resulting volume will have a set of slopes, offsets and breakpoints.
The main challenge to fit a piecewise linear function now is to define the breakpoints. For this we will use a criterium which indicates how good a linear regression result fits the data: R^2, the coefficient of determination [12]. R^2 will be determined by the following formula:

$$R^2 = 1 - \frac{\sum_k (\mu_k - \nu_k)^2}{\sum_k (\mu_k - \overline{\mu})^2} \tag{5.3}$$

$$\nu_k = s \cdot t_k + f \tag{5.4}$$

In this, the sum over k runs over all data points (time points) over which the linear regression is performed for this particular linear fit. μ_k is the 'measured' value of the linear attenuation coefficient and ν_k is the value predicted by this linear fit. $\overline{\mu}$ is the average μ_k over the fitted data range.
When a line is a good fit, R^2 should be close to 1. When it is a bad fit, R^2 will approach 0. In any case, $0 \leq R^2 \leq 1$.
We have a number of data points $\mu_k = \mu(t_k)$ in function of time coordinates t_k, sorted in ascending order, i.e. $t_{k-1} < t_k < t_{k+1}$ for all k. We choose a fixed line length L. L will be much smaller than the available time range, but L will encompass a reasonable amount of time points t_k. If $L = t_{k+1} - t_k$, i.e. L is the length between two adjacent data points, then a fitted line of length L will only be fitted to two data points, providing a perfect fit each time. If this would be the case, we can not use R^2 as a measure to search for breakpoints, since it would be 1 everywhere. Therefore, $L > t_{k+1} - t_k$. In order to choose a good L, we consider the specific sample and dynamic process of this scan. We expect a certain distance in time between two sudden dynamic events (breakpoints), which can be estimated from prior knowledge or from the 3D reconstructions before the piecewise linear fit. L should be a bit below this temporal distance. Within one 4D μCT scan, L is fixed.
Now we start fitting lines of length L to the data, calculating their R^2 each

time. These R^2 can be plotted in function of the first time point t_k to which this line was fitted. For clarity, we will also add the endpoint of the line, $t_k + L$ to the notation. We will have data $R^2(t_k; t_k + L)$ from $R^2(0; L)$ up until $R^2(T - L; T)$, T being the final time point t.
The goal is to search at which t_k the best breakpoints will be located, based on the behaviour of $R^2(t_k; t_k + L)$. Specifically (for reasons outlined in section 5.3.3), we will assume breakpoints are located at t_k for which R^2 displays the following characteristics:

- $R^2(t_k - L; t_k) \approx R^2(t_{k-1} - L; t_{k-1})$ and $R^2(t_k - L; t_k) > R^2(t_{k+1} - L; t_{k+1})$ (this means that $R^2(t_k - L; t_k)$ displayed a plateau to the left and starts to go down to the right)
- $R^2(t_k; t_k + L) \approx R^2(t_{k+1}; t_{k+1} + L)$ and $R^2(t_k; t_k + L) > R^2(t_{k-1}; t_{k-1} + L)$ (this means that $R^2(t_k + L)$ displayed a plateau to the right and starts to go down to the left)
- There exists a t_l for which $t_k - L < t_l < t_k$ and $R^2(t_l; t_l + L) < R_{threshold}$

$R_{threshold}$ is chosen to be $0.7 \cdot \frac{1}{2}(R^2(t_k - L; t_k) + R^2(t_k; t_k + L))$, but this is not a fixed number for this algorithm. The final criterium is to make sure there is a reasonably deep 'valley' between $R^2(t_k)$ and $R^2(t_k + L)$.
After applying these criteria, the result is a series of t_k that are possible breakpoints. To make sure the linear regression which will occur between breakpoints has enough data, we assume each line will be at least L_{min} long. We set $L_{min} = L$, but a different value might be chosen as well. Two breakpoint-candidates that are closer than L_{min} together will be replaced by their average, starting with the two t_k that are closest together, until every pair of t_k is at least L_{min} apart.

5.3.3 Reasoning behind the breakpoint detecting algorithm

The breakpoint detecting algorithm is described in section 5.3.2. As mentioned there, the breakpoints are picked based on the behaviour of $R^2(t_k; t_k + L)$, with L a constant line length and t_k the time coordinates which may be breakpoints. To see how we got to the previously given criterium for breakpoints, let's take a look in figure 5.3 at some $R^2(t_k; t_k + L)$ plotted in function of $t_k + L$ for perfect piecewise linear data. L was 10 in this case, while the minimum length of one line was 20, which we know because this is simulated data.
In figure 5.3 the piecewise linear function (left figure), which is our test data, has two different breakpoints: one continuous (at $t = 33$) and one discontinuous (at $t = 91$). Both are clearly visible on the R^2-graph (right figure), but

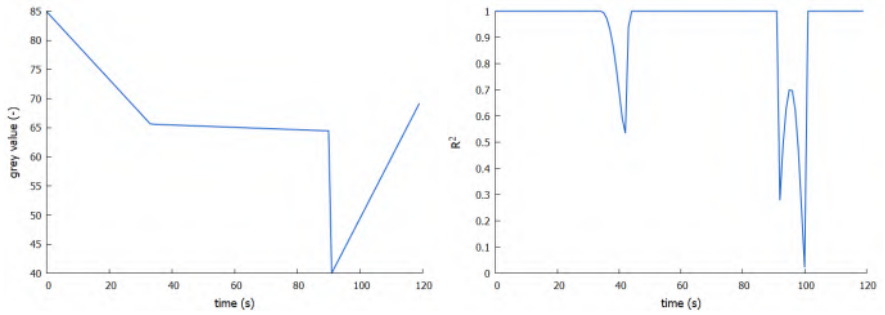


Figure 5.3: Left: a random piecewise linear function without noise. Right: the R^2 corresponding to the function depicted at the left.

display different behaviours. Let's move from left to right in the graph, so increasing t. When the data points in range $(t-L, t)$ belong to a perfect line, $R^2(t-L,t) = 1$. It is first when t reaches the breakpoint, that $R^2(t-L,t)$ begins to drop, since it reflects how well a line is fitted to a set of data points that are no longer described well by linear regression. When $t-L$ reaches the breakpoint (i.e. exactly L further), $R^2(t-L,t)$ will return to 1.
The basic behaviour around a breakpoint is therefore a large drop in R^2 of L width. Around a continuous breakpoint, this is the exact description of the behaviour. Around a discontinuous breakpoint, we see a (smaller) peak inside the valley of R^2. This is because the fitted line will not go straight from the line before the breakpoint to the one after. It will first attempt to fit the discontinuous jump with a very steep line. This will temporarily cause R^2 to rise.
Since we want to find both continuous and discontinuous breakpoints, the algorithm will search for a 'valley' of L wide and of a reasonable depth. The depth is checked to rule out some noise that would otherwise disguise itself as a breakpoint.
In figure 5.4 the criterium for finding breakpoints is used on data of a piecewise linear function with some added noise. The noise is very clear in the R^2-plot, yet the criterium finds the correct breakpoints and a few extra ones. The extra breakpoints are not a problem, since a breakpoint in the middle of a straight line results in about the same straight line fitted on both sides. The real breakpoints in this case were $t = 27$ and $t = 62$. The breakpoints found were $t = 27$, $t = 63$, $t = 79$, $t = 90$ and $t = 105$.
A reason to choose this breakpoint criterium instead of a more established fitting algorithm such as the Nelder-Mead simplex algorithm [13] or Powell's method [14] is mainly its simplicity and speed. Linear regression only considers each data point once and therefore the calculation of every R^2 is $\mathcal{O}(n)$,

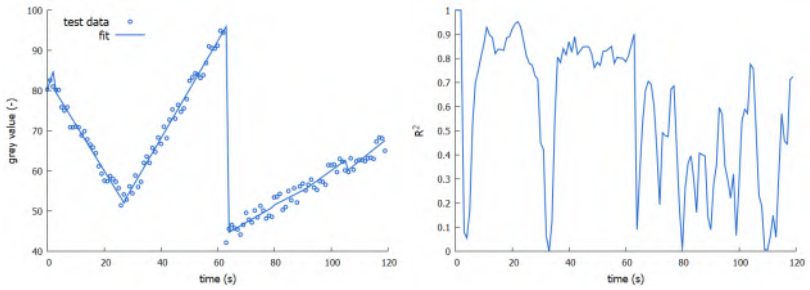


Figure 5.4: Left: a random piecewise linear function with noise and the fitted function to this data. Right: the R^2 corresponding to the function depicted at the left.

n being the number of data points, i.e. the number of 3D reconstructions. There are n R^2 to be calculated, but each one can use the results of the previous one, each time removing one data point and adding another one, so the complexity remains $\mathcal{O}(n)$. After finding these breakpoints, the final linear regression is again $\mathcal{O}(n)$, making the resulting complete fitting algorithm $\mathcal{O}(n)$ for one voxel.
By implementing this procedure on a GPU (graphics processing unit), each thread processing one voxel, the execution time is very short. For example, in a case where an original reconstruction of all time steps, also on the GPU, takes of the order of an hour, the fitting takes of the order of minutes.

5.4 Results

In the following, we denote the axis system by $xyzt$, with t the time dimension and the z-axis being parallel to the rotation axis of the tomographic acquisition. The scanned sample is described in section 2.8. We divided the dataset into time steps the length of one rotation, spaced overlappingly to have 158 time steps. As described in section 2.8, the total scan had about 80 rotations. Each time step was then reconstructed using SART [15] (see section 2.4.2) with the framework described in the PhD thesis of De Schryver [16].
After reconstruction of the time steps with SART (see section 2.4.2), a piecewise linear fit with $L = 13$ was performed on the time evolution of each voxel, as described in section 5.3.2. The original 4D reconstruction consisted of 158 time steps and therefore 158 volumes. By fitting, the amount of data was reduced to 32 parameter volumes, meaning the voxel with the most fitted breakpoints had 11 lines (each providing 2 parameters: a slope and offset)

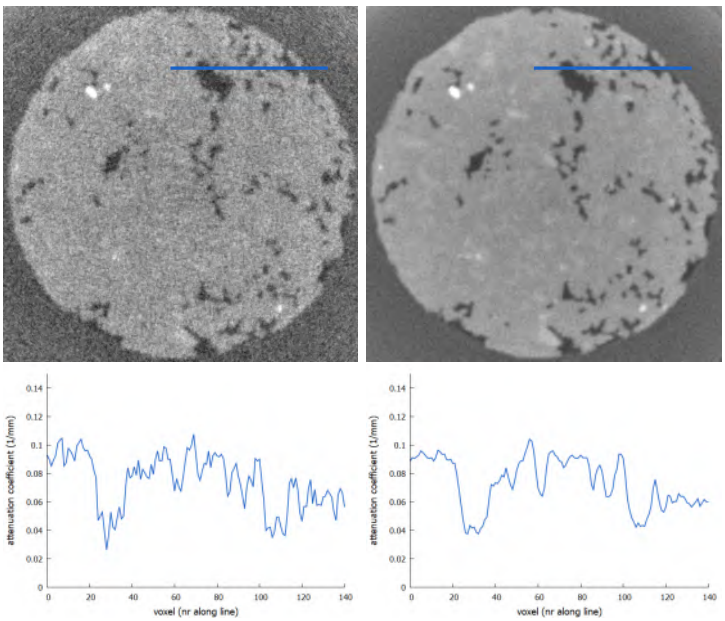


Figure 5.5: Left: xy-slice of a reconstruction of the Bentheimer sandstone. Right: slice of the same reconstruction after performing a piecewise linear fit on the dataset. The blue line indicated in the slices is plotted in the graphs below.

and 10 breakpoints (excluding the first and last time step, which are the limits of the time dimension. Each breakpoint is one parameter). Each line has three parameters: a slope, an offset and the breakpoint where it ends. The last line does not need a breakpoint, since it ends at the end of the time range in any case. Since the amount of breakpoints can be different for each voxel, each voxel will have a different amount of parameters to save. We saved as many volumes as the largest amount and padded those voxels that needed less parameters with zeros.
Decreasing the value of L increases the maximum amount of breakpoints fitted for one voxel, which is the total amount of time of the scan divided by L. In the total volume, there is almost always at least one voxel reaching this value. However, the final analysis results (see chapter 6) proved robust against changes in L, since extra breakpoints at static points in time do not influence the result.
Therefore, piecewise linear fitting reduced the amount of computer memory needed to store the results by a factor of $\frac{32}{158} = 0.203$. This is while saving a complete volume for each parameter - the data storage could be further

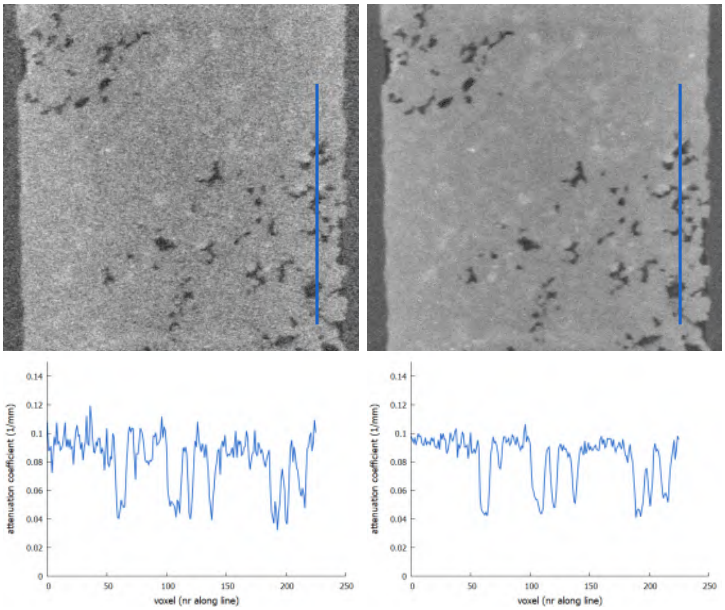


Figure 5.6: Left: xz-slice of a reconstruction of the Bentheimer sandstone. Right: slice of the same reconstruction after performing a piecewise linear fit on the dataset. The blue line indicated in the slices is plotted in the graphs below.

reduced by only saving the amount of necessary parameters for each voxel. Figures 5.5 and 5.6 show slices and plots of this reconstruction before and after piecewise linear fitting. The surrounding dark pixels contain a viton sleeve around the Bentheimer sandstone sample. The light grey round object represents a slice through the Bentheimer sandstone. The dark shapes inside the stone consist of pores filled with oil.
The reduction in spatial noise levels is clearly visible in both the horizontal and the vertical directions. This is remarkable, since the fitting only affects the t-dimension. While the noise is significantly reduced, the spatial resolution is not or barely affected, as seen in figures 5.5 and 5.6. The line profile displays slopes at the same spots and the height of the minima and maxima is the same, while they would be less pronounced in the case of blurring. This is a consequence of the fact that piecewise linear fit does not operate in the spatial dimensions, only in the temporal one.

In contrast to the spatial dimensions, the time dimension, visible in figure 5.7, experiences some smoothing. A voxel in figure 5.7 is a vertical line, with time running from top to bottom. A dynamic pore which gets filled with

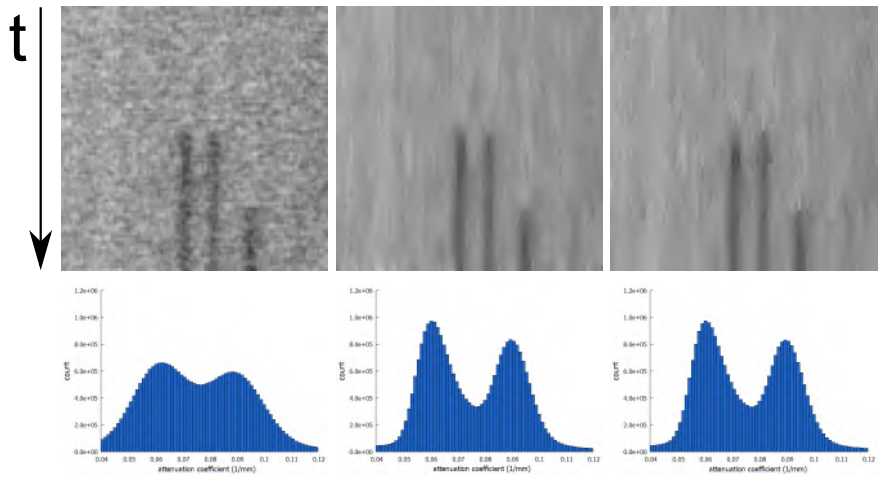


Figure 5.7: Left: part of an xt-slice of a reconstruction of the Bentheimer sandstone. Middle: the same xt-slice from a median-filtered volume. Right: slice of the same reconstruction as the left figure, after performing a piecewise linear fit on the dataset. On the bottom line histograms are shown of the grey values within a region in the rock that contained both rock and pores filled with oil.

oil during the scan will be displayed as a grey line at the top and black at the bottom. While the breakpoints are usually placed on or near the "time-edges", a lot of these edges are still smoothed, because the fitting algorithm placed a breakpoint right before and after the real moment oil entered this voxel. Further research on the optimization of the breakpoint selection algorithm is therefore still desirable. Alternatively, we could include the spatial correlations to some extent.
Figure 5.7 compares the result of a piecewise linear fit with the result of a noise filter, in this case a median filter in the time domain. The filter was only applied in the time dimension and is applied at a strength that gives a similar noise reduction as the linear fit, as can be verified in the histograms and in table 5.1. On each of the histograms, two peaks are visible: one at the grey value of the oil and one at the grey value of the rock. Table 5.1 lists the full width at half maximum of each of these peaks. It is clear that the piecewise linear fit gives a similar noise reduction as the temporal dimension median noise filter. A piecewise linear fit handles discontinuities in the time dimension (in this case, the edges of the black rectangles in figure 5.7) as a breakpoint or a very steep slope. Both possibilities should be considered when searching for these discontinuities, for example for an analysis as de-

	FWHM of left peak	FWHM of right peak
normal reconstruction	0.0303	0.0233
median filtered	0.0213	0.0174
piecewise linear fitted	0.0212	0.0173

Table 5.1: The full widths at half maximum for the peaks visible on the histograms in figure 5.7.

scribed in section 6.2.

The graphs in figure 5.8 illustrate how the fitting works on individual voxels. While the human eye would probably obtain a different fit (with less breakpoints), the result is still good, as the most important breakpoint for the human eye is present in the right graph (around $t = 10500$) and it is approximated by a very steep line in the middle graph (around $t = 40000$).

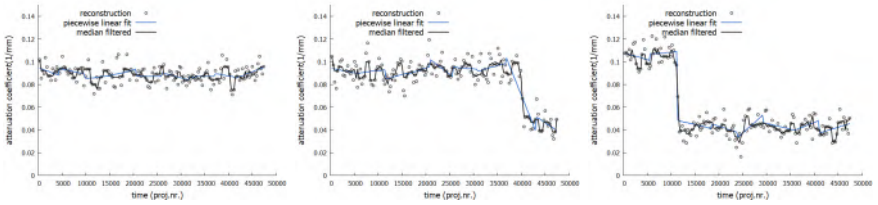


Figure 5.8: Left: plot of grey value in function of time for a static voxel (who does not undergo a dynamic process during the scan). Middle and right: a similar plot for two different dynamic voxels.

Piecewise linear fitting can be used alongside any 4D reconstruction technique, such as an initial volume or weighted back projection, as shown in figure 5.9. The chosen reconstruction technique can be tuned to the specific sample and performed completely independent from the piecewise linear fit. This combination can profit from the advantages of both piecewise linear fit and the specific reconstruction technique, i.e. a piecewise linear fit acts as an additional noise filter and as a stepping stone to further analysis, as will be described in chapter 6.

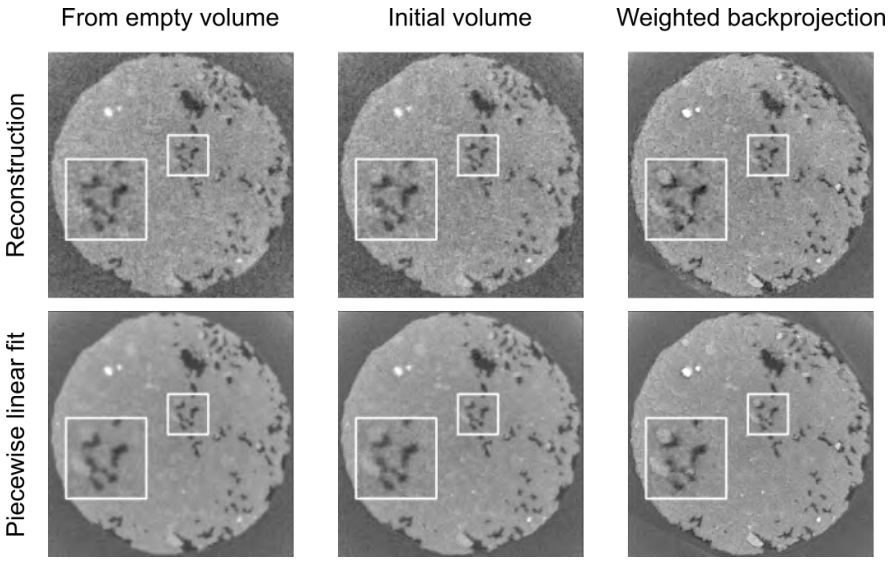


Figure 5.9: A comparison of xy-slices of reconstructions before and after performing a piecewise linear fit. The reconstructions were made with three different reconstruction techniques: normal SART, SART from an initial volume (section 3.2) and weighted back projection (chapter 4). A small part of the slice is shown zoomed in.

5.5 Conclusion

4D micro-CT scans often suffer from large amounts of noise. Piecewise linear fitting is a simple technique aiming to tackle this. Every 3D voxel in the reconstructed volume gets a piecewise linear function fitted to its grey value time evolution. This reduces the noise significantly and requires less computer memory.
Piecewise linear fitting was demonstrated on a Bentheimer sandstone and showed great noise reduction, while maintaining spatial resolution.
The individual time steps of the reconstructed Bentheimer sandstone used overlapping ranges of projections for reconstruction, in order to have enough time steps to perform a piecewise linear fit. A motion artefact originating from a small number of projections therefore propagates to multiple time steps. The individual time steps of the reconstructed Bentheimer sandstone used overlapping ranges of projections for reconstruction, in order to have enough time steps to perform a piecewise linear fit. A motion artefact originating from a small number of projections therefore propagates to multiple time steps. When using overlapping projection ranges, more time steps are

represented as an actual reconstruction. So if a motion is occurring in Δt time, it will be present in for example 1 reconstructed time step when using adjacent projection ranges for reconstructions, and in more than 1 when using overlapping projection ranges. However, the total time in which the motion artefact is present is still Δt .

Some techniques to deal with the reconstruction of dynamic scans mainly improve the spatial resolution and signal-to-noise ratio of the static regions in the sample, possibly smoothing any information present in the temporal direction ([17], [18]). Piecewise linear fitting on the other hand also improves the static regions - by fitting constant functions or lines with a low slope to these pieces - but does not smooth the dynamic regions as much in the temporal dimension, mainly due to the use of breakpoints. A dynamic scan suffering from a lot of noise can benefit greatly from the piecewise linear fitting technique.

Furthermore, if there is prior information available on the sample, it can be used to improve the reconstruction even more. Specifically, the initial reconstructions might be improved with any chosen reconstruction method without affecting the piecewise linear fitting result, including filtered back projection or iterative methods, noise filters, segmentation, etc. Combining multiple techniques like this when they suit the sample characteristics is common practice (for example by Dobson *et al.* [19]).

If it is known beforehand that the time evolution of a voxel should follow a certain function (for example piecewise constant), it is best to fit this specific function instead of a piecewise linear one. When the time evolution is not known beforehand, a piecewise linear function is a good choice since it emulates all other functions. It is also a simple function, making it easy to fit and requiring less computing power than more complex functions would need.

With this method, any time point during the scan can be shown, instead of a discrete number of reconstructed ones. In addition, less memory is needed to store these results, as you only need to store the parameters of the fitted function for each voxel, instead of an entire volume for each time step.

The current implementation could be further improved by including spatial information, so for each voxel information on its neighbours can be included in the calculations. This could improve the correct placement of the breakpoints and therefore reduce unwanted temporal smoothing. It might also remove spurious breakpoints and therefore reduce noise even more.

Another possibility to improve the result is to impose the piecewise linear fit in between iterations of the reconstruction, instead of as a post-processing step. However, this technique is currently limited by the available computer memory. Since the entire 4D reconstructed volume does not fit in RAM-memory, read- and write-operations to and from the hard disc would be required for

each iteration, since the piecewise linear fit requires the temporal information, while the reconstruction uses the 3D spatial volumes. This would vastly increase the reconstruction time.
The combination of this technique with advanced 4D reconstruction techniques, such as motion registration [16, 20] (see section 3.3), is a future prospect, hopefully able to deal with more complexity in the samples.

References

[1] M. Heyndrickx, M. Boone, T. De Schryver, T. Bultreys, W. Goethals, G. Verstraete, V. Vanhoorne, and L. Van Hoorebeke. *Piecewise linear fitting in dynamic micro-CT.* Materials Characterization, 139:259–268, 2018.

[2] A. M. Mendrik, E. Vonken, T. Witkamp, M. Prokop, B. van Ginneken, and M. A. Viergever. *Using the Fourth Dimension to Distinguish Between Structures for Anisotropic Diffusion Filtering in 4D CT Perfusion Scans.* In Spatio-temporal Image Analysis for Longitudinal and Time-Series Image Data, pages 79–87, Cham, 2015. Springer International Publishing.

[3] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, L. Van Hoorebeke, and V. Cnudde. *Fast laboratory-based micro-computed tomography for pore-scale research: Illustrative experiments and perspectives on the future.* Advances in Water Resources, 95:341–351, 9 2016.

[4] A. D. McNaught and A. Wilkinson. *IUPAC. Compendium of Chemical Terminology, 2nd ed. (the "Gold Book").* Blackwell Scientific Publications, Oxford, 1997.

[5] B. Akça and S. Z. Erzeneoğlu. *The Mass Attenuation Coefficients, Electronic, Atomic, and Molecular Cross Sections, Effective Atomic Numbers, and Electron Densities for Compounds of Some Biomedically Important Elements at 59.5 keV.* Science and Technology of Nuclear Installations, 2014.

[6] R. S. J. Sparks and H. E. Huppert. *Density changes during the fractional crystallization of basaltic magmas: fluid dynamic implications.* Contributions to Mineralogy and Petrology, 85(3):300–309, 3 1984.

[7] C. Y. Ho and R. E. Taylor. *Thermal Expansion of Solids.* CINDAS data series on material properties. ASM International, 1998.

[8] Wikipedia, 2017, *Density.*

[9] J. Marsiglio, 2015, *Piecewise linear approximation.*

[10] G. Van Eyndhoven, K. J. Batenburg, D. Kazantsev, V. Van Nieuwenhove, P. D. Lee, K. J. Dobson, and J. Sijbers. *An Iterative CT Reconstruction Algorithm for Fast Fluid Flow Imaging.* IEEE Transactions on Image Processing, 24(11):4446–4458, 11 2015.

[11] Wikipedia, 2017, *Simple linear regression.*

[12] Wikipedia, 2018, *Coefficient of determination.*

[13] F. Gao and L. Han. *Implementing the Nelder-Mead simplex algorithm with adaptive parameters.* Computational Optimization and Applications, 51(1):259–277, 1 2012.

[14] M. J. Powell. *An efficient method for finding the minimum of a function of several variables without calculating derivatives.* The computer journal, 7(2):155–162, 1964.

[15] A. H. Andersen and A. C. Kak. *Simultaneous Algebraic Reconstruction Technique (SART): A Superior Implementation of the Art Algorithm.* Ultrasonic Imaging, 6(1):81–94, 1984. PMID: 6548059.

[16] T. De Schryver. *Fast imaging in non-standard X-ray computed tomography geometries.* PhD thesis, Ghent University, 2017.

[17] D. Kazantsev, G. Van Eyndhoven, W. R. B. Lionheart, P. J. Withers, K. J. Dobson, S. A. McDonald, R. Atwood, and P. D. Lee. *Employing temporal self-similarity across the entire time domain in computed tomography reconstruction.* Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 373(2043), 2015.

[18] D. Kazantsev, W. M. Thompson, W. R. Lionheart, G. Van Eyndhoven, A. P. Kaestner, K. J. Dobson, P. J. Withers, and P. D. Lee. *4D-CT reconstruction with unified spatial-temporal patch-based regularization.* Inverse problems and imaging, 9(2):447–467, 2015.

[19] K. J. Dobson, S. B. Coban, S. A. McDonald, J. N. Walsh, R. C. Atwood, and P. J. Withers. *4D imaging of sub-second dynamics in pore-scale processes using real-time synchrotron X-ray tomography.* Solid Earth, 7(4):1059, 2016.

[20] T. De Schryver, M. Dierick, M. Heyndrickx, J. Van Stappen, M. A. Boone, L. Van Hoorebeke, and M. N. Boone. *Motion compensated micro-CT reconstruction for in-situ analysis of dynamic processes.* Scientific Reports, 8:7655, 2018.

6
Analysis

The detailed analysis of a geological sample

This chapter describes an analysis of the sandstone sample discussed in section 2.8. We will combine 3D analysis techniques (as in section 6.1) with 4D analysis techniques (based on piecewise linear fitting, see section 6.2) to get the most complete description of oil transport through the pores of a Bentheimer sandstone. We represent this sample as a volume of $512 \times 512 \times 512$ voxels.

6.1 Segmentation of the pore space

In the sandstone sample described in section 2.8, the pore space is of large interest. Therefore, we performed a segmentation to segregate this pore space from the surrounding rock. This segmentation started from the first scan described in section 2.8, a high quality static μCT scan of the sandstone where the pores were not yet saturated with an attenuation-matching solution, so they were still easily discerned from the rock. We used this scan, instead of the dynamic scan, since not all pores got filled with oil during the dynamic process (so not all pores ended up visible) and the noise level in the dynamic scan is significantly higher.

We used the software Octopus Analysis (formerly known as Morpho+ [1])

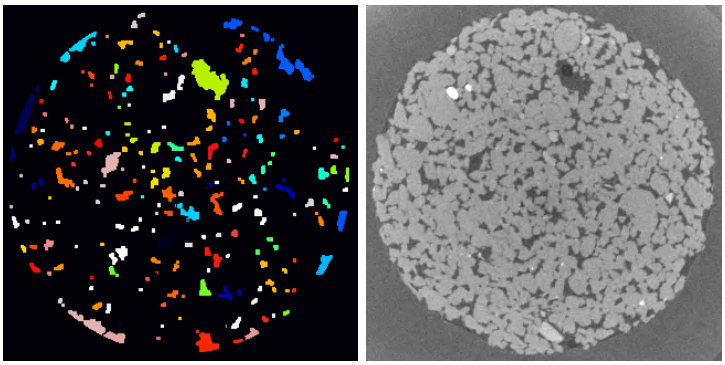


Figure 6.1: Left: slice of the segmented pore volume of the Bentheimer sandstone. Each color represents a different pore. To the right is the unsegmented pore volume for comparison.

to perform this segmentation on a volume of $512 \times 512 \times 512$ voxels, reconstructed with SART (section 2.4.2). The following steps were taken:

1. Select a cylindrical volume of interest closely encompassing the sample and discarding the top and bottom slices, where cone beam artefacts (see section 2.5.7) distort the image.

2. Select the pore space of the sample using a dual threshold filter. This filter first encloses all voxels with an attenuation coefficient lower than a threshold A into the segmented space. It then adds the voxels which are between thresholds A and B and that are neighbours of a segmented region, possibly through a region of other voxels that have an attenuation coefficient between A and B.

3. Use a shrink filter [2] to remove some of the noise, as well as (unfortunately) the smallest pores.

4. Use an expand filter [2] to restore the segmented pores to their original size.

5. Label the objects in the segmented pores. This will cause each pore to become a separate object, which is listed.

The result is a list of 9771 objects, each representing one pore having a size of 7 voxels or larger (the used size of the shrink and expand filters was 2 voxels, meaning too small pores were eliminated). 4224 of these segmented pores have a size of 100 voxels or larger, which is equivalent to a diameter of only 6 voxels in a spherical shape. The relation between the number of voxels in a pore N and the equivalent spherical diameter is the formula for the volume of a sphere $N = \frac{4}{3}\pi\left(\frac{d}{2}\right)^3$ with d the diameter in number of voxels. Since each voxel is a cube with a side of 0.02 mm, more than half of the pores have a volume smaller than $0.0008\,\text{mm}^3$ (100 voxels). Therefore, most of these segmented objects are very small pores. The largest pore consists of 20 423 voxels, which is equivalent to a sphere with a diameter of 34 voxels. This means the largest pore has a volume of $0.164\,\text{mm}^3$.
Figure 6.1 shows a visual representation of one slice of the pore space. Each segmented pore is displayed as a different colour in this slice.

6.2 Time of occurrence of steepest slope

After a piecewise linear fit reconstruction, as described in chapter 5, we have a 3 dimensional volume of $512 \times 512 \times 512$ voxels in which each voxel does not have one attenuation coefficient, but a set of parameters describing a piecewise linear function.
We would like to know, for each voxel, if oil entered during the experiment and if so, when exactly that happened. The theoretically perfectly reconstructed voxel can have two possible evolutions of its attenuation coefficient in function of time: either it is a static voxel and the function will be a constant, or it is a voxel where oil entered, in which case the function will consist of two constant functions with a breakpoint in between. The second

constant, the attenuation coefficient of oil, will be lower. This breakpoint is the time of interest t at which oil filled this voxel.
In a real reconstructed voxel, the functions will not follow this perfect behaviour. There will be more than one breakpoint and even the static parts will have a slope that is not exactly zero, although it will still be small. This is due to noise and artefacts (see section 2.5).
In these "real voxels", the time t at which specific pores got filled with oil shows up as a steep decreasing line or as a large discontinuous jump at a breakpoint. In order to find this t, we searched for the time point of each voxel where the fitted line had the steepest decreasing slope. A discontinuous jump at a breakpoint was given a 'slope' of $\Delta\mu/\Delta t$, with $\Delta\mu$ the size of the change in attenuation coefficient and Δt the distance between two consecutive time points. This procedure is just the searching of a minimum (largest negative value) and is therefore a simple analysis.
Static voxels, which keep the same attenuation coefficients during the entire scan, will have a piecewise linear function fitted to them which is a combination of multiple lines with a slope close to zero. Due to noise variations, this will not be a perfect constant function. This means the line with the steepest slope will be somewhat random for these voxels. As such, they will obfuscate the information of the interesting voxels. In order to avoid this noise, we add another criterium. A line can only qualify to have the steepest decreasing slope if it causes a change in attenuation coefficient from its start to its end which is at least $\Delta\mu'$, with $\Delta\mu'$ a value chosen to be a little less than what we estimate the real dynamic change to be.
The result of this analysis is a 3D volume with a resulting time t for each

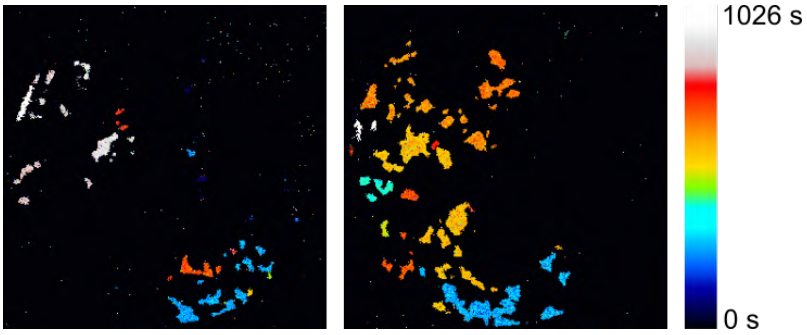


Figure 6.2: The 256$^{\text{th}}$ (left, the slice also shown in figure 6.1) and 203$^{\text{th}}$ (right) slice of the resulting 3D volume, consisting of 512 slices in total, from an analysis searching the steepest slope after a piecewise linear fit.

voxel. Figure 6.2 shows two slices of this result. We can discern complete pores, as all voxels in a pore seem to get a similar time t as a result. This

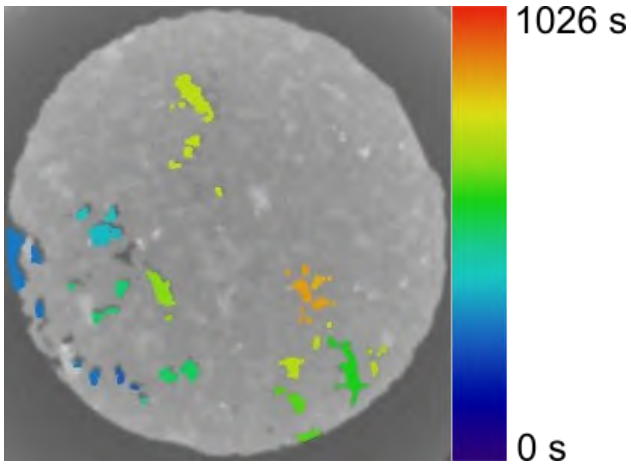


Figure 6.3: The 156$^{\text{th}}$ slice of the Bentheimer sandstone. Static voxels display in grey their average attenuation coefficient over the entire dynamic scan. Dynamic pores display in colour the result of the analysis searching the steepest slope after a piecewise linear fit, averaged over the entire segmented pore. Note: to properly view this figure, consult a colour print or the online version.

corresponds to the known behaviour of one pore being filled in a short time, shorter than our temporal resolution, making it appear 'instantaneous' in this reconstruction. This phenomenon is called Haines jumps [3] and one jump does not always fill just one pore. The fact that multiple pores get filled in one jump is also visible on figure 6.2: there are clusters of pores with a very similar t.
Figure 6.3 shows yet another slice, this time as a combination of the attenuation coefficients of static voxels with the resulting time t for the dynamic voxels. In this figure, the static voxels are averaged over all time steps, while the t for dynamic voxels is averaged over all voxels within a pore. The resulting figure therefore contains much less noise compared to figure 6.2. We choose a different look-up-table to transfer t to a color, compared to figure 6.2, because white, black and grey colors would cause confusion between the dynamic and the static part of this figure.

When plotting the distribution of the resulting time per voxel within one pore, using the pore coordinates from the segmentation described in section 6.1, a Gaussian behaviour is observed. An example for one such pore is displayed in figure 6.4, where a histogram of the filling times for each voxel within one pore is displayed along with a plotted Gaussian function. While the distribution is not exactly Gaussian, it is quite close. This is expected

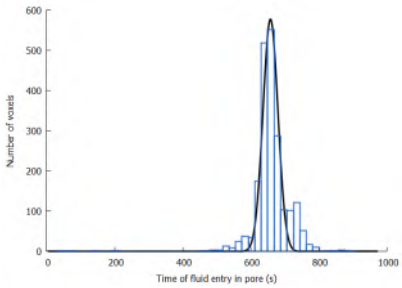


Figure 6.4: Histogram of how many voxels within one pore have a certain time of steepest slope.

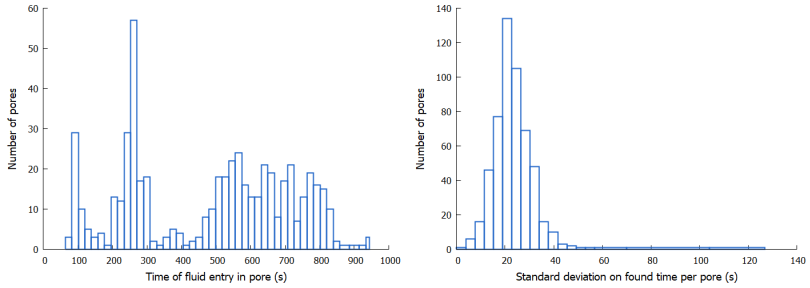


Figure 6.5: Histogram of how many pores have a certain time of steepest slope and of the standard deviation σ on this time.

behaviour, since all voxels within one pore should have the same resulting filling time. The variance on this central filling time is due to the noise levels. A small number of pores displayed bimodal filling time distributions, i.e. there were two individual peaks instead of one. A possible explanation is that pores with this behaviour were segmented incorrectly: each of these could actually consist of two individual pores, segmented as if it was only one. These pores will introduce a small error in the following analysis, since a Gaussian function can not be properly fitted to them. The very low number of such pores keeps the effect small.
In order to have a better view on this distribution of Haines jumps, we calculated the average resulting time per pore. For this, we ignored any voxel that did not have a resulting t, i.e. that had a 'flat' time evolution with no visible jump. Also not included are those pores where less than half of the voxels had a resulting t, since we assume in this case the pore did not get filled and only had a few voxels provide a resultint t due to noise fluctuations. The majority of voxels did not have a result, as they did not get filled with

fluid in the duration of the CT scan.
These resulting times of all filled pores were plotted in a histogram, visible in figure 6.5 (left). Two clear peaks are visible around 100 s and 250 s. These could indicate two big Haines jumps, influencing many pores simultaneously. The flattening towards later times is also logical: many pores are already filled at that time, so the fluid has multiple paths to continue, instead of a few bottlenecks where the pressure needs to build before the fluid breaks through, as in the first Haines jumps.
Apart from an average filling time per pore, the standard deviation on this time within a pore was also calculated and displayed in a similar histogram in figure 6.5 (right). This is a Gaussian distribution with a tail to the right: this is because σ can be arbitrarily large, but never lower than 0. Most have a standard deviation below 40 s. This is a rather large deviation (the total scanning time was 1026 s and one rotation was 12.85 s), signalling that the data contains a lot of noise. Therefore, analysis results for a single voxel have a large uncertainty. However, combining this analysis with the pore segmentation provides a much better view, since we can average over all voxels within one pore. Therefore, the results will be less influenced by noise for larger pores, in which the averaging happens over a larger amount of voxels.
We perform the same analysis on the phantom reconstructions. The phantoms are described in section 2.8. A series of projections, 600 per rotation, are created from these with added Poisson noise. Define Δt as the time required to acquire a single projection. The reconstructions, the piecewise linear fitting and the analysis (searching for the time of the steepest slope) are handled similarly to the data processing of the real scan as described above. When comparing the analysis result within individual pores to the ground truth of the phantoms, we found a minimal error of 0.5 Δt between ground truth and analysis result and an average error over all pores of 1038 Δt (i.e. 1,73 rotations) when reconstructing from low-noise projections, and a minimal error of 4.3 Δt with an average error over all pores of 1218 Δt (i.e. 2.03 rotations) when reconstructing from high-noise projections. While the analysis result is therefore close to the ground truth, it is not perfect yet and an optimal result requires further improvements to the technique.

6.3 Check: is the analysis result physically meaningful?

In the dynamic process of the Bentheimer sandstone, the sample investigated here, oil flows through pores and 'pushes away' the attenuation-matching liquid which previously filled the pores. Logically, oil can only flow in those pores that are connected to at least one other pore that is already filled with

oil, as the oil must physically have a path to reach the new location, or those pores that are connected to the border of the sample where the oil is pumped in.
We want to check if the analysis result from section 6.2 is in agreement with the previous explanation, i.e. pores are only filled at time t_1 if at least one of their neighbours was filled before t_1. Therefore, we need information not only on the pores themselves, but also on how they are connected.
Analysis of two phase fluid flow in geological materials, as the Bentheimer sandstone sample (see section 2.8) often uses a *pore network*. This is a segmentation of pores (as in section 6.1) combined with a segmentation of the throats, i.e. the connections between pores. A pore network provides not only information on the pores themselves, but also on which pores are connected to which pores, along with properties of the throats such as thickness and length.
A pore network was extracted by Tom Bultreys (PProGRess, department of geology, Ghent University) for the Bentheimer sandstone sample using open-source pore network extraction software from Imperial College London [4]. This network extraction started from the reconstruction of the first scan described in section 2.8, a high quality static μCT scan of the sandstone where the pores were easily discerned from the rock. First, voxels are separated into pore-space and background (similarly to the method described in section 6.1, i.e. using thresholds), upon which a *distance map* is performed. This means that for each voxel, the closest distance to the border of the object it belongs to, is calculated. The local maxima in this distance map are used as seeds for a watershed-based separation [5] to separate individual pores from each other, while saving the information of which pores are neighbouring which pores. The program calculates different properties of the resulting pores, such as the radius of the largest inscribed sphere.

With this extracted pore network, we investigated the analysis result from section 6.2 further. We calculated the average analysis result time of filling t_j of a pore j and checked whether this pore had a neighbouring pore with $t_{neighbour} < t_j$. If so, this pore was classified 'physical'. Otherwise, it was classified 'unphysical'. In total, 90.00% of the pores were classified physical. Both sets of pores (physical and unphysical) are plotted as t_j in function of pore radius and in function of vertical coordinate z in figure 6.6. Visual assessment indicates that most unphysical pores have a low t_j. Indeed, calculating the percentage of unphysical pores with $t_j < 95.92\,\mathrm{s}$, which is the lowest 5% of the complete time range, returns 38.43%. Another, smaller effect visible in figure 6.6 is that there are more unphysical pores at low z, i.e. in the bottom of the sample. Numerically, the lowest 5% of the complete z-range, which is all $z < 76.50$ voxels, hosts $10,98$% of unphysical pores.
Both effects can be explained by the same principle: a number of pores does

not have a neighbouring pore that was filled previously, because they are either filled from the edge of the sample (the oil was pumped in from the bottom of the sample, i.e. low z), or they do have a filled neighbouring pore, but this pore was filled before the X-rays turned on, i.e. before the first radiograph and therefore before the first reconstructed time step. As visible in figure 6.7, quite some pores were already filled at the very first time step. A lot of the pores that are filled with oil from this region will be filled in the first time steps (hence the high amount of unphysical pores at low t_j). However, some of these might still be filled later in the experiment, which will also explain some of the unphysical pores, appearing later.

The next step in this analysis would be to segment the pores that are already filled before the first time step and include pores that neighbour those as physical instead of unphysical. Lack of time prevented us thus far to perform this additional analysis. The first analysis provides a high likelihood of the results being correct, since a high amount is physical and many of the unphysical pores can be explained.

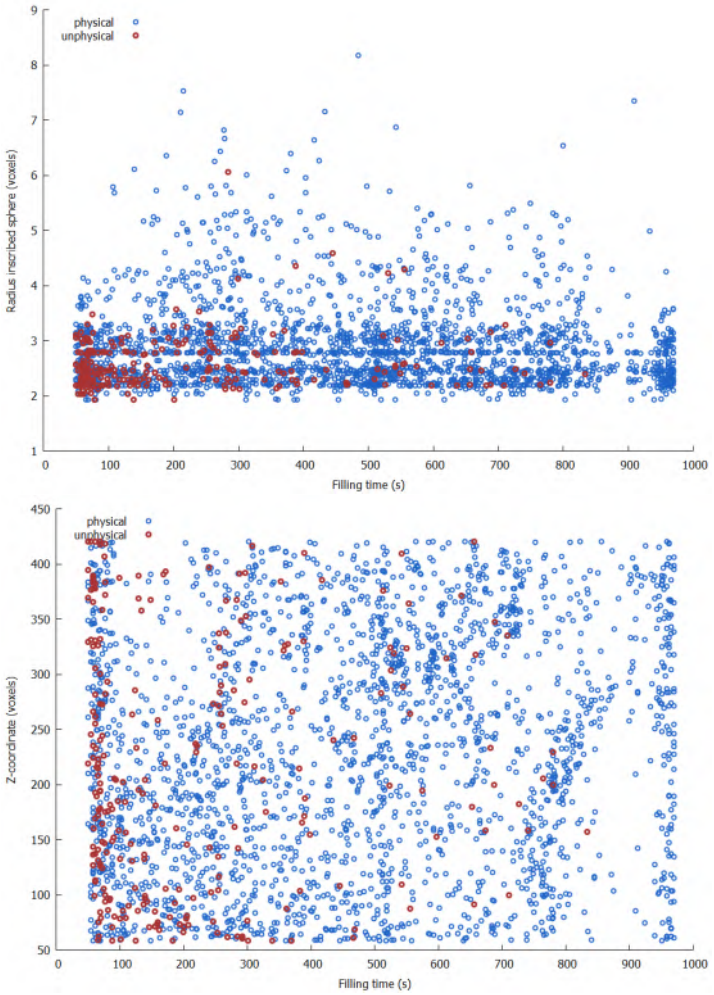


Figure 6.6: Plot showing the pores that have a physical or a non-physical analysis result according to the method used in section 6.3 in function of pore size (radius of largest inscribed sphere) and z-coordinate.

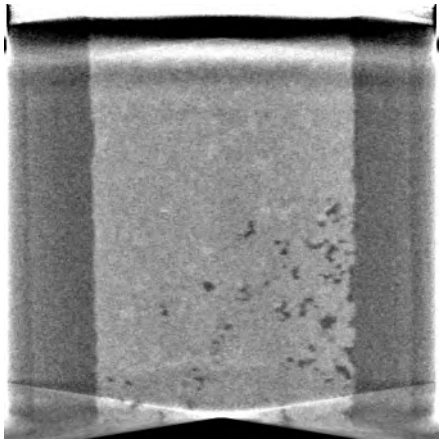


Figure 6.7: The central XZ-slice of the reconstruction of the Bentheimer sandstone at the first reconstructed time step (centered at 6.4125 s). The distortions at the top and bottom are due to cone beam artefacts (section 2.5.7).

References

[1] L. Brabant, J. Vlassenbroeck, Y. De Witte, V. Cnudde, M. N. Boone, J. Dewanckele, and L. Van Hoorebeke. *Three-Dimensional Analysis of High-Resolution X-Ray Computed Tomography Data with Morpho.* Microscopy and Microanalysis, 17(2):252–263, 2011.

[2] L. Brabant. *Latest developments in the improvement and quantification of high resolution X-ray tomography data.* PhD thesis, Ghent University, 2013.

[3] T. Bultreys, M. A. Boone, M. N. Boone, T. De Schryver, B. Masschaele, D. Van Loo, L. Van Hoorebeke, and V. Cnudde. *Real-time visualization of Haines jumps in sandstone with laboratory-based microcomputed tomography.* Water Resources Research, 51(10):8668–8676, 2015.

[4] H. Dong and M. J. Blunt. *Pore-network extraction from micro-computerized-tomography images.* Phys. Rev. E, 80:036307, 9 2009.

[5] S. Beucher and F. Meyer. *Segmentation: The Watershed Transformation. Mathematical Morphology in Image Processing.* Optical Engineering, 34:433–481, 01 1993.

7

Conclusion

What we learned from this research

4D μCT is a technique to non-destructively investigate dynamic processes inside samples. Unfortunately, one scan (one time step) requires a finite time to perform, therefore limiting the temporal resolution when using conventional algorithms dealing with reconstruction and analysis.
As explained in chapter 1, the goal of this research was threefold:

1. Developing methods to use prior knowledge in an iterative reconstruction.

2. Extend the existing 3D analysis methods to 4D, with time being the extra dimension.

3. Combine both reconstruction and analysis in one framework.

Ultimately, all three of these goals are aiming at improving the reconstruction and analysis of 4D μCT.
We tested all techniques on our demonstration dataset of fluid flow through a Bentheimer sandstone. This porous rock was first saturated with an attenuation-matching fluid, i.e. a fluid with a similar attenuation coefficient to the surrounding rock, filling all the pores. The dynamic process was the pumping of oil through this sandstone.

The first goal was achieved with the development of weighted back projection in chapter 4. This iterative reconstruction technique uses two types of prior knowledge, an initial volume and a weight volume. The initial volume is typically the reconstruction of a high-quality μCT scan before the dynamic process was initiated or after it has concluded. For our demonstration sample, it was a scan from before the oil intrusion.
The weight volume is a volume indicating which regions in the sample are likely to change in attenuation coefficient due to the dynamic process and which are static. For our sample, we based this weight volume on a high-quality μCT scan of the porespace of the Bentheimer sandstone, i.e. a scan performed before the pores were saturated with the attenuation-matching liquid.
Both the initial volume and the weight volume are used to adapt an iterative reconstruction such as SART. The values in the weight volume determine how much of the back projected correction is assigned to dynamic or static voxels. The resulting reconstruction, weighted back projection, is able to reconstruct dynamic μCT scans using fewer projections per rotation. The fast convergence provides good results when only using 10% of available projections in our demonstration dataset, while conventional reconstructions are not able to reconstruct any relevant information in this case. Using only 10% of available projections means we are simulating a faster scanner, that would rotate 10 times faster, while taking the same amount of projections per time unit. Such a scanner could use this technique to provide the same reconstruction quality. Faster processes could be imaged in this way. Unfortunately, the mechanical components of the scanners at UGCT do not allow this high speed, so we have not demonstrated weighted back projection on a real faster dataset.
A disadvantage of weighted back projection is the high susceptibility to noise in the dynamic regions. We tested several techniques to combat this, including one developed in this research, dynamic relaxation, and found that the simplest way was to decrease the relaxation factor compared to the best choice of relaxation factor for a conventional reconstruction.

Goal nr. 3 was achieved in chapter 5 by bridging reconstruction and analysis with the technique of piecewise linear fitting. In this technique, we fit a piecewise linear function to the time evolution of the attenuation coefficient in each voxel after a 4D reconstruction with any chosen reconstruction technique. This function can approximate any function, even discontinuous ones, and is therefore a suited choice when it is not known in advance what function the time evolution of the scanned dynamic process will follow. The simplicity of the function allowed us to implement the fitting in an $\mathcal{O}(N)$ (with N the number of time points) algorithm, which in turn made it suited to implement on a GPU, since these are designed to perform many small

computations in parallel, so less complex calculations for one thread are an advantage. The parallel implementation on a GPU is a necessity because of the gigantic amount of voxels (10^9), since the fitting would otherwise require hours to perform.
The result of a piecewise linear fit was demonstrated on the Bentheimer sandstone sample. The noise in the reconstructions is significantly reduced. Even though piecewise linear fitting only operates in the temporal dimensions, the noise reduction also occurs in the spatial dimensions, without affecting the spatial resolution.
As demonstrated in chapter 5, piecewise linear fitting provides a similar noise filtering effect as noise filters, which are available in many forms and for many different applications. The noise filtering of piecewise linear fitting was a very positive effect, since it did not decrease the spatial resolution. The main advantage of piecewise linear fitting however is the fact that it is a bridge towards analysis.

In chapter 6, we explored the analysis of the Bentheimer sandstone, starting from the piecewise linear fitting result from chapter 5. The different time steps resulting from a reconstruction were transferred into a function describing the time evolution of the attenuation coefficient in each voxel with piecewise linear fitting, effectively linking the discrete time steps to a continuous range. This allowed the analysis in chapter 6 to make use of the fourth dimension, instead of repeating a 3D analysis for each time step, effectively fulfilling goal nr. 2.
More specifically, we examined the time points at which specific pores were filled with fluid or whether they were never filled during the scan. We segmented the pore space with a standard 3D analysis to be able to examine the properties per pore, so the chapter demonstrates a combination of 3D and 4D analysis. The segmentation of the pore space also shows that the result of the 4D analysis is physically logical. The resulting time of filling for each voxel within a pore is similar and the distribution follows a Gaussian distribution. This is logical, since we know that one pore is filled during a very short time that can be considered instantaneous. Therefore, the distribution of filling times for all the voxels within one pore should show a peak. The width of the fitted Gaussian function indicates the uncertainty. By using a pore network with information on which pores neighbour which pores, we also verified another logical property of the analysis result: pores can only be filled with oil after one of the neighbouring pores was filled with oil, since the oil needs a physical path to reach a new pore.

4D μCT is still an open subject with room for progress. A future prospect is the combination of weighted back projection with motion registration (see

section 3.3), since weighted back projection is suited for dynamic processes occurring within a static grid, while motion registration handles deformation in samples. Presumably, a smart combination of both could handle a very wide range of applications.
The possibility of an analysis resulting from a piecewise linear fit was demonstrated on one sample in this thesis and on another in the paper describing this technique. It would be interesting to see how general the possibilities are and how many different applications could benefit from it, since the requirements for analysis are so different for each research.
We briefly discussed the many developments in the domain of 4D μCT in chapter 3, most being applicable to a certain range of applications. The future will probably bring many more possibilities and techniques to handle 4D data. Especially the use of neural networks shows a lot of potential, with many researchers exploring this promising avenue. One of the big drawbacks of neural networks, the requirement of a high amount of training data, seems to be diminishing with smart ways to increase the use of a single training data set or the use of good simulations. I also expect big improvements by using more external measurements as prior knowledge for the reconstruction of a dynamic process. For example, for fluid flow through a geomaterial, the total amount of fluid pumped into the sample at each time point could define limits on the reconstruction that could greatly improve the final image quality. Hopefully, in time, techniques will be developed having fewer limitations and more advantages, making them ever more useful so the limits of research can be moved further and further.