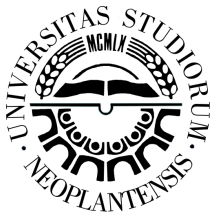


UNIVERSITY OF NOVI SAD
FACULTY OF TECHNICAL SCIENCES

A novel algorithm for high bit-depth seismic data compression

DOCTORAL THESIS

Author:
Miloš RADOSAVLJEVIĆ

Supervisors:
Dr. Dejan VUKOBRATOVIĆ
Dr. Zixiang XIONG

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Technical Sciences

in the

Department of Power, Electronics, and Communication Engineering

May 6, 2021

УНИВЕРЗИТЕТ У НОВОМ САДУ **ОБРАЗАЦ – 5а**
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА[1]

Врста рада:	Докторска дисертација
Име и презиме аутора:	Милош Радосављевић
Ментор (титула, име, презиме, звање, институција)	Проф. др Дејан Вукобратовић, Факултет техничких наука, Нови Сад Проф. др Zixiang Xiong, A и M Универзитет, Тексас, САД
Наслов рада:	„Нови алгоритам за компресију сеизмичких података велике амплитудске резолуције“, енг. „A novel algorithm for high bit-depth seismic data compression“
Језик публикације (писмо):	Енглески
Физички опис рада:	Страница 129 Поглавља 7 Референци 208 Табела 7 Слика 21 Графикона 0 Прилога 0
Научна област:	Енергетика, електроника и телекомуникације
Ужа научна област (научна дисциплина):	Обрада сигнала
Кључне речи / предметна одредница:	Компресија сеизмичких података велике амплитудске резолуције, 3Д волуметријски сеизмички подаци, ХЕВЦ.
Резиме на језику рада:	Renewable sources cannot meet energy demand of a growing global market. Therefore, it is expected that oil & gas will remain a substantial sources of energy in a coming years. To find a new oil & gas deposits that would satisfy growing global energy demands, significant efforts are constantly involved in finding ways to increase efficiency of a seismic surveys. It is commonly considered that, in an initial phase of exploration and production of a new fields, high-resolution and high-quality images of the subsurface are of the great importance. As one part in the seismic data processing chain, efficient managing and delivering of a large data sets, that are vastly produced by the industry during seismic surveys, becomes extremely important in order to facilitate further seismic data processing and interpretation. In this respect, efficiency to a large extent relies on the efficiency of the compression scheme, which is often required to enable faster transfer and access to data, as well as efficient data storage. Motivated by the superior performance of High Efficiency Video Coding

[1] Аутор докторске дисертације потписао је и приложио следеће Обрасце:
5б – Изјава о ауторству;
5в – Изјава о истоветности штампане и електронске верзије и о личним подацима;
5г – Изјава о коришћењу.
Ове Изјаве се чувају на факултету у штампаном и електронском облику и не кориче се са тезом.

	(HEVC), and driven by the rapid growth in data volume produced by seismic surveys, this work explores a 32 bits per pixel (b/p) extension of the HEVC codec for compression of seismic data. It is proposed to reassemble seismic slices in a format that corresponds to video signal and benefit from the coding gain achieved by HEVC inter mode, besides the possible advantages of the (still image) HEVC intra mode. To this end, this work modifies almost all components of the original HEVC codec to cater for high bit-depth coding of seismic data: Lagrange multiplier used in optimization of the coding parameters has been adapted to the new data statistics, core transform and quantization have been reimplemented to handle the increased bit-depth range, and modified adaptive binary arithmetic coder has been employed for efficient entropy coding. In addition, optimized block selection, reduced intra prediction modes, and flexible motion estimation are tested to adapt to the structure of seismic data. Even though the new codec after implementation of the proposed modifications goes beyond the standardized HEVC, it still maintains a generic HEVC structure, and it is developed under the general HEVC framework. There is no similar work in the field of the seismic data compression that uses the HEVC as a base codec setting. Thus, a specific codec design has been tailored which, when compared to the JPEG-XR and commercial wavelet-based codec, significantly improves the peak-signal-to-noise-ratio (PSNR) vs. compression ratio performance for 32 b/p seismic data. Depending on a proposed configurations, PSNR gain goes from 3.39 dB up to 9.48 dB. Also, relying on the specific characteristics of seismic data, an optimized encoder is proposed in this work. It reduces encoding time by 67.17% for All-I configuration on trace image dataset, and 67.39% for All-I, 97.96% for P2-configuration and 98.64% for B-configuration on 3D wavefield dataset, with negligible coding performance losses. As a side contribution of this work, HEVC is analyzed within all of its functional units, so that the presented work itself can serve as a specific overview of methods incorporated into the standard.
Датум прихватања теме од стране надлежног већа:	1.4.2021.
Датум одбране: (Попуњава одговарајућа служба)	
Чланови комисије: (титула, име, презиме, звање, институција)	Председник: Др Дубравко Ћулибрк, професор, Факултет техничких наука, Нови Сад Члан: Др Јелена Николић, ванредни професор, Универзитет у Нишу Члан: Др Татјана Лончар-Турукало, ванредни професор, Факултет техничких наука, Нови Сад
Напомена:	

UNIVERSITY OF NOVI SAD
FACULTY OF TECHNICAL SCIENCES

KEY WORD DOCUMENTATION[2]

Document type:	Doctoral dissertation
Author:	Miloš Radosavljević
Supervisor (title, first name, last name, position, institution)	dr Dejan Vukobratović, professor, Faculty of Technical Sciences, Novi Sad dr Zixiang Xiong, professor, Electrical and Computer Engineering Texas A & M University, College Station, Texas, USA.
Thesis title:	A novel algorithm for high bit-depth seismic data compression
Language of text (script):	English language
Physical description:	Pages 129 Chapters 7 References 208 Tables 7 Illustrations 21 Graphs 0 Appendices 0
Scientific field:	Power, Electronic and Telecommunication Engineering
Scientific subfield (scientific discipline):	Signal processing
Subject, Key words:	High bit-depth seismic data compression, 3D volumetric seismic data, HEVC.
Abstract in English language:	Renewable sources cannot meet energy demand of a growing global market. Therefore, it is expected that oil & gas will remain a substantial sources of energy in a coming years. To find a new oil & gas deposits that would satisfy growing global energy demands, significant efforts are constantly involved in finding ways to increase efficiency of a seismic surveys. It is commonly considered that, in an initial phase of exploration and production of a new fields, high-resolution and high-quality images of the subsurface are of the great importance. As one part in the seismic data processing chain, efficient managing and delivering of a large data sets, that are vastly produced by the industry during seismic surveys, becomes extremely important in order to facilitate further seismic data processing and interpretation. In this respect, efficiency to a large extent relies on the efficiency of the compression scheme, which is often required to enable faster transfer and access to data, as well as efficient data storage. Motivated by the superior performance of High Efficiency Video Coding (HEVC), and driven by the rapid growth in data volume produced by seismic surveys, this work explores a 32 bits per pixel (b/p) extension of the HEVC codec for compression of seismic data. It is proposed to reassemble seismic

[2] The author of doctoral dissertation has signed the following Statements:
5б – Statement on the authority,
5в – Statement that the printed and e-version of doctoral dissertation are identical and about personal data,
5г – Statement on copyright licenses.
The paper and e-versions of Statements are held at he faculty and are not included into the printed thesis.

	slices in a format that corresponds to video signal and benefit from the coding gain achieved by HEVC inter mode, besides the possible advantages of the (still image) HEVC intra mode. To this end, this work modifies almost all components of the original HEVC codec to cater for high bit-depth coding of seismic data: Lagrange multiplier used in optimization of the coding parameters has been adapted to the new data statistics, core transform and quantization have been reimplemented to handle the increased bit-depth range, and modified adaptive binary arithmetic coder has been employed for efficient entropy coding. In addition, optimized block selection, reduced intra prediction modes, and flexible motion estimation are tested to adapt to the structure of seismic data. Even though the new codec after implementation of the proposed modifications goes beyond the standardized HEVC, it still maintains a generic HEVC structure, and it is developed under the general HEVC framework. There is no similar work in the field of the seismic data compression that uses the HEVC as a base codec setting. Thus, a specific codec design has been tailored which, when compared to the JPEG-XR and commercial wavelet-based codec, significantly improves the peak-signal-to-noise-ratio (PSNR) vs. compression ratio performance for 32 b/p seismic data. Depending on a proposed configurations, PSNR gain goes from 3.39 dB up to 9.48 dB. Also, relying on the specific characteristics of seismic data, an optimized encoder is proposed in this work. It reduces encoding time by 67.17% for All-I configuration on trace image dataset, and 67.39% for All-I, 97.96% for P2-configuration and 98.64% for B-configuration on 3D wavefield dataset, with negligible coding performance losses. As a side contribution of this work, HEVC is analyzed within all of its functional units, so that the presented work itself can serve as a specific overview of methods incorporated into the standard.
Accepted on Scientific Board on:	1.4.2021.
Defended: (Filled by the faculty service)	
Thesis Defend Board: (title, first name, last name, position, institution)	President: dr Dubravko Ćulibrk, professor, Faculty of Technical Sciences, Novi Sad Member: dr Jelena Nikolić, associate professor, University of Niš Member: dr Tatjana Lončar-Turukalo, associate professor, Faculty of Technical Sciences, Novi Sad
Note:	

Contents

To my family.

Miloš RADOSAVLJEVIĆ

Doctoral Thesis Committee

Supervisors:

Dr. Dejan VUKOBRATOVIĆ, Department of Power, Electronics, and Communication Engineering, University of Novi Sad, Serbia
Dr. Zixiang XIONG, Department of Electrical and Computer Engineering, Texas A & M University, College Station, Texas, USA

Committee Members:

Dr. Dubravko ĆULIBRK, Department of Industrial Engineering and Management, University of Novi Sad, Serbia

Dr. Tatjana LONČAR TURUKALO, Department of Power, Electronics, and Communication Engineering, University of Novi Sad, Serbia

Dr. Jelena NIKOLIĆ, Department of Telecommunications, University of Niš, Serbia

Declaration of Authorship

I, Miloš RADOSAVLJEVIĆ, declare that this thesis titled "A novel algorithm for high bit-depth seismic data compression", and the work presented in it, are my own. I hereby certify that the thesis I am submitting is entirely my own original work, except where otherwise indicated. I have not used any sources other than those listed in the bibliography and identified as references. I am aware of the University's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism. Any use of the works of any other author, in any form, is properly acknowledged at their point of use. I further declare that I have not submitted this thesis at any other institution in order to obtain a degree.

Signed:

Date: May 6, 2021

UNIVERZITET U NOVOM SADU, Fakultet tehničkih nauka

Departman za energetiku, elektoniku i telekomunikacije

Novi algoritam za kompresiju seizmičkih podataka velike amplitudske rezolucije

autor Miloš RADOSAVLJEVIĆ

u cilju sticanja zvanja Doktor tehničkih nauka

Rezime

Imajući u vidu da obnovljivi izvori sve teže uspevaju da zadovolje trenutnu potražnju na rastućem globalom tržištu energenata, očekuje se da će i u narednim godinama nafta i gas ostati značajni izvori energije. Da bi se pronašla nova nalazišta nafte i gasa koja bi zadovoljila rastuće globalne energetske potrebe, neprekidno se ulažu značajni napori i sredstva kako bi se pronašli načini za povećanje efikasnosti seizmičkih istraživanja. Generalno se smatra da su u početnoj fazi istraživanja i proizvodnje novih naftnih i gasnih polja slike podzemnih površina visoke rezolucije i visokog kvaliteta od velike važnosti. Efikasno upravljanje i prenos velikih skupova podataka nastalih tokom seizmičkih istraživanja, prva su karika u lancu obrade i tumačenja seizmičkih podataka. Kao deo lanca obrade, efikasno upravljanje i isporuka velikih skupova podataka, koje industrija uglavnom proizvodi tokom seizmičkih istraživanja, postaje izuzetno važno kako bi se olakšala njihova dalja obrada i tumačenje. U tom pogledu, navedeno se u velikoj meri oslanja na efektivnu šemu kompresije, koja je neophodna da bi se omogućio brži prenos i pristup podacima, kao i njhovo efikasno skladištenje.

Motivisani superiornim performansama standarda visokoefikasnog video kodovanja, *eng. High Efficiency Video Coding (HEVC) standard*, i podstaknuti brzim rastom u količini podataka generisanih tokom seizmičkih istraživanja, ovaj rad istražuje ekstenziju HEVC kodeka za kompresiju seizmičkih podataka

velike amplitudske rezolucije od 32 bita po pikselu (b/p). Predloženo je da se seizmički isečci posmatraju u formi koja odgovara video signalu, čime se dobija na efikasnosti kodovanja upotrebom HEVC inter režima (prediktivno kodovanje pokretnih slika), pored mogućih prednosti HEVC intra režima zasnovanog na kodovanju individualnih (mirnih) slika. U tom cilju, ovaj rad modifikuje gotovo sve komponente originalnog HEVC kodeka kako bi se podržalo kodovanje seizmičkih podataka velike amplitudske rezolucije: Lagranžev multiplikator koji se koristi za optimizaciju parametara kodiranja prilagođen je novoj statistici podataka, transformacija i kvantizacija su ponovo implementirane da bi se omogućila njihova upotreba kod povećanog opsega bitske dubine, i izmenjen je adaptivni binarni aritmetički koder koji je korišćen za efikasno entropijsko kodiranje. Pored toga, testiran je optimizovan odabir veličine blokova, smanjeni broj intra modova i fleksibila procene vektora pomeraja sa ciljem prilagođavanja kodeka strukturi seizmičkih podataka. Iako novi kodek premašuje standardizovani HEVC nakon primene predloženih modifikacija, on i dalje održava generičku HEVC strukturu i razvijen je u okviru njegove opšte šeme. Kada je u pitanju oblast kompresije seizmičkih podataka, do sada nije sprovedeno istraživanje koje koristi HEVC kao osnovnu postavku kodeka. Na taj način je kreiran specifičan dizajn kodeka koji u poređenju sa JPEG-XR standardom i komercijalnim kodekom zasnovanim na vejvletima (*eng. wavelet-based codec*) značajno poboljšava performanse u pogledu vršnog odnosa signal-šum (*eng. peak-signal-to-noise ratio (PSNR)*) i stepena kompresije za seizmičke podatke od 32 b/p. U zavisnosti od predloženih konfiguracija, PSNR poboljšanje kreće se od 3.39 dB do 9.48 dB. Takođe, oslanjajući se na specifične karakteristike seizmičkih podataka, u ovom radu je predložen optimizovani enkoder. Vreme kodovanja smanjeno je za 67.17% kod All-I konfiguracije na trace skupu podataka 67.39% kod All-I, 97.96% kod P2-konfiguracije i 98.64% kod B-konfiguracije na 3D wavefield skupu podataka, sve sa neznatnim gubicima u performansama kodovanja.

Kao sporedni doprinos ovog rada, analizirane su sve funkcionalne celine (moduli) HEVC standarda, tako da predstavljeni rad može poslužiti kao specifičan pregled metoda koje su uključene u standard.

Ključne reči - Kompresija seizmičkih podataka velike amplitudske rezolucije, 3D volumetrijski seizmički podaci, HEVC.

UNIVERSITY OF NOVI SAD, Faculty of Technical Sciences
Department of Power, Electronics, and Communication Engineering

A novel algorithm for high bit-depth seismic data compression

by Miloš RADOSAVLJEVIĆ

for the degree of Doctor of Technical Sciences

Abstract

Renewable sources cannot meet energy demand of a growing global market. Therefore, it is expected that oil & gas will remain a substantial sources of energy in a coming years. To find a new oil & gas deposits that would satisfy growing global energy demands, significant efforts are constantly involved in finding ways to increase efficiency of a seismic surveys. It is commonly considered that, in an initial phase of exploration and production of a new fields, high-resolution and high-quality images of the subsurface are of the great importance. As one part in the seismic data processing chain, efficient managing and delivering of a large data sets, that are vastly produced by the industry during seismic surveys, becomes extremely important in order to facilitate further seismic data processing and interpretation. In this respect, efficiency to a large extent relies on the efficiency of the compression scheme, which is often required to enable faster transfer and access to data, as well as efficient data storage.

Motivated by the superior performance of High Efficiency Video Coding (HEVC), and driven by the rapid growth in data volume produced by seismic surveys, this work explore a 32 bits per pixel (b/p) extension of the HEVC codec for compression of seismic data. It is proposed to reassemble seismic slices in a format that corresponds to video signal and benefit from the coding gain achieved by HEVC inter mode, besides the possible advantages of the (still image) HEVC intra mode. To this end, this work modify almost all components of the original HEVC codec to cater for high bit-depth coding of seismic data: Lagrange multiplier used in optimization of the coding parameters has been

adapted to the new data statistics, core transform and quantization have been reimplemented to handle the increased bit-depth range, and modified adaptive binary arithmetic coder has been employed for efficient entropy coding. In addition, optimized block selection, reduced intra prediction modes, and flexible motion estimation are tested to adapt to the structure of seismic data. Even though the new codec after implementation of the proposed modifications goes beyond the standardized HEVC, it still maintains a generic HEVC structure, and it is developed under the general HEVC framework. There is no similar work in the field of the seismic data compression that uses the HEVC as a base codec setting. Thus, a specific codec design has been tailored which, when compared to the JPEG-XR and commercial wavelet-based codec, significantly improves the peak-signal-to-noise-ratio (PSNR) vs. compression ratio performance for 32 b/p seismic data. Depending on a proposed configuration, PSNR gain goes from 3.39 dB up to 9.48 dB. Also, relying on the specific characteristics of seismic data, an optimized encoder is proposed in this work. It reduces encoding time by 67.17% for All-I configuration on trace image dataset, and 67.39% for All-I, 97.96% for P2-configuration and 98.64% for B-configuration on 3D wavefield dataset, with negligible coding performance losses.

As a side contribution of this work, HEVC is analyzed within all of its functional units, so that the presented work itself can serve as a specific overview of methods incorporated into the standard.

Index Terms - High bit-depth seismic data compression, 3D volumetric seismic data, HEVC.

Acknowledgements

All the presented work in this thesis is carried out under the supervision of Prof. Dr. Dejan Vukobratović, University of Novi Sad, Serbia, and Prof. Dr. Zixiang Xiong, Texas A & M University, TX, USA. The joint work on the topic has been initiated as a part of the project QoSTREAM - Video Quality Driven Multimedia Streaming in Mobile Wireless Networks, funded by the European Commission, Grant number PIRSES-GA-2011-295220. The project supported my stay at Texas A & M University for 12 months in total. Prof. Xiong and I continued working remotely over the time, and as a result of our collaboration this thesis emerged. I would also like to thank Dr. Ligang Lu (Shell Exploration and Production Inc., TX, USA) for his supervision and concept validation. Dr. Lu provided to me all necessary information about seismic data. He has been helpful in providing technical background in the domain of the seismic data processing and seismic data compression. He also provided seismic datasets used in this thesis, and also provided additional benchmark results of the two codecs that are used to compare the proposed solution. I would like to thank once again to Prof. Dr. Zixiang Xiong and Prof. Dr. Dejan Vukobratović for believing in me. Many thanks to Prof. Xiong for his concise and clear advices. In conversations with him, I learned how to approach the problem pragmatically and how to think out of the box. During conversations about the project, he often almost unknowingly gave me significant life lessons. Special thanks for Prof. Dr. Vukobratović for encouragement and unconditional support during all these years. His leadership spirit taught me how the right things come at the right time, he taught me patience, and taught me to solve things gradually one by one.

I am infinitely grateful to my family, especially to my parents for their support during my schooling. Without them none of this would be possible. Finally, I thank my wife a lot for her patience and support, and because she has always been my inspiration and driving force during our life together.

List of Figures

List of Tables

Published Work

Miloš Radosavljević, Marko Panić, Vladan Minić and Predrag Lugonja, "Local Binary Patterns and Its Applications in Analysis of 3D Scene", *Conference of Digital Speech and Image Processing, Kovačica, Serbia, October 2012., (paper is written in Serbian)*

Miloš Radosavljević, Srdjan Sladojević, Dubravko Ćulibrk, and Dejan Vukobratović, "QoE-aware Rate-Conservative Dynamic HTTP Streaming Over Mobile Cellular Networks", *IEEE/IFIP Network Operations and Management Symposium (NOMS), 5-9 May 2014, Krakow, Poland*

Miloš Radosavljević, Georgios Georgakarakos, Sebastien Lafond and Dejan Vukobratović, "Fast coding unit selection based on local texture characteristics for HEVC intra frame", *IEEE Global Conference on Signal and Information Processing (GlobalSIP), Orlando, FL, 14-16 Dec. 2015*

Miloš Radosavljević, Zixiang Xiong, Ligang Lu and Dejan Vukobratović, "High bit-depth image compression with application to seismic data", *IEEE International Conference Visual Communications and Image Processing (VCIP), 27-30 Nov. 2016, Chengdu, China.*

Miloš Radosavljević, Zixiang Xiong, Ligang Lu, Detlef Hohl, and Dejan Vukobratović, "HEVC-based compression of high bit-depth 3D seismic data", *IEEE International Conference on Image Processing (ICIP), 17-20 Sept. 2017, Beijing, China.*

Miloš Radosavljević, Marko Adamović, Branko Brkljač, Željen Trpovski, Zixiang Xiong, and Dejan Vukobratović, "Satellite Image Compression Based on High Efficiency Video Coding Standard - An Experimental Comparison With JPEG 2000", *Conf. on Big Data from Space (BiDS) Turning Data into Insights, 19-21 Feb. 2019, Munich, Germany*

Miloš Radosavljević, Branko Brkljač, Predrag Lugonja, Vladimir Crnojević, Zixiang Xiong, and Dejan Vukobratović, "Lossy Compression of Multispectral Satellite Images with Application to Crop Thematic Mapping: A HEVC Comparative Study", *MDPI Remote Sensing (Special Issue Remote Sensing Data Compression), 12, no. 10: 1590.*

Miloš Radosavljević, Zixiang Xiong, Ligang Lu, Detlef Hohl, and Dejan Vukobratović, "High Bit-Depth Seismic Data Compression: A Novel Codec Under the Framework of HEVC", *IEEE Access, vol. 8, pp. 114443-114459, 2020.*

Chapter 1

Introduction and Motivation

In the age of the fourth industrial revolution and the apparent increase in human population [3], there is a global need for additional energy in order to support sustainable economic and social development. With continued population growth, the expectation is that by the middle of the century global energy demand will constantly rise compared with now-days. Of course, it is difficult to recognize the exact needs, especially given the many local and global factors that are often intersected by economic and political influences. In that sense, there is always interest in analyzing the geo-political, social, economic, and technological aspects that drive energy sector [4, 5, 6, 7, 8]. Due to many possible outcomes and the great uncertainty of what might happen in the future, variety of scenarios emerged, taking into account the various variables and dependencies between them. The previously referenced reports present many interesting analyses and different scenarios, which represent different foreseeable pathways for energy transition.

Renewable energy, even with its rapid rise and energy efficiency awareness, cannot meet demand, and conventional fossil fuels are needful. It is obvious, from today's point of view, that switching to carbon-free sources on a scale comparable to the oil & gas industry would not be easy and quick. For example, today's share of oil & gas in global primary energy consumption is 57.3% according to [9]. In addition, it is difficult to find any substitutes, given the current technologies, for the energy-rich hydrocarbon fuels in some industrial sectors, such as in chemical industry, iron, and steel industry where high temperatures are required, or in long-distance heavy-duty road transportation or seaborne freight. Fossil fuels will remain a certain factor of the global energy system, also because of its long-term reliability, cost-effective production, convenient storage and transportation capabilities. Regardless of the many factors that may

influence the direction in which the world will develop, each report [4,5,6,7,8], and the scenarios presented therein, predict that oil & gas will remain part of the future drive of the society and economy to a greater or lesser extent.

However, what we know for sure is that, to align with Paris Agreement (more can be found at [10, 11, 12]), and in order to meet the climate goals presented in it, the energy sector will definitely be changed. Gas, as the cleanest fossil fuel, will play a vital role in reducing carbon-dioxide and other pollutants. It will become an extremely important source of energy, whose growing demand is mainly driven by industrial and power generation use. Gas is expected to replace solid fuels in the power sector, which is currently heavily dependent on much dirtier coal. Coal as one of the dominant fuels for heavy industry and electricity production today uses 27% of the total energy sources [6, 9]. In order to reduce carbon-dioxide emission heavily produce by the coal, in some sectors it can be replaced by the much cleaner gas. In the developing countries many people still use solid fuels, e.g., firewood, for domestic purpose which can also be replaced by gas. Consequently, natural gas accounted for the largest increment in energy consumption in recent few years, and it will continue to grow. For example, in 2019, the share of gas in primary energy rose to a record high of 24.2%, with a tendency to surpass coal in a total share (which is falling at its lowest level of production since 2003) [9].

Even with the large breakthrough of renewable energy sources in electricity generation, electrification is not totally aimed towards green sources (only 10.4% of electricity production today comes from renewables according to [9]). According to [7], in 2040, fossil-based generation remains the dominant source of electricity, albeit at a lower share compared to today, since gas penetration accelerates in the power sector. Wind and solar are deployed rapidly thanks to the falling costs of renewable electricity generation. Hence, as technology advances, renewable energy overtakes fossil fuels such as oil, gas and coal as the primary source of electricity in the late 2050s [13]. In addition, even that electricity mix increasingly shifts towards renewable sources, regionally there is significant variation in share of global electricity generation by fuel. For example, due to the geographical location of the renewables resource base, which is often a long way from centers of energy demand, in some regions higher penetrations of renewable are challenging to accomplish [4]. In that light, many emerging economies, such as China and India, continue to pay more attention to the stability of the power sector, and stick to, and invest in, conventional sources, such as gas. As already mentioned, gas is the most reliable way to suppress

coal (Asia's dominant fuel in electricity generation) in the most economical and fastest way, and without greater exposure to highly alarming environmental risks caused by carbon-dioxide emission. According to [7], by 2040 renewable power rise to 33-43% depending on a scenario. The rest of the needs is mostly met by coal, oil, and gas.

Car electrification, development of on-demand mobility services, smarter integration of public transport systems, or ride sharing, are evolving, which brings down use of oil for passenger vehicles. Although the global car fleet is more than doubling, to reach two billion by 2040, oil consumption in public transport is declining [5]. In part, this also comes from rapid gains in energy efficiency, e.g., efficiency of cars with internal combustion engine (ICE), but also from fuel switching - from hydrocarbon fuels to electricity. One uncertainty, however, comes with the current popularity of sport utility vehicles (SUVs), which are more difficult to electrify fully, and conventional SUVs consume more fuel than medium-sized cars [6]. If the popularity of SUVs continues to grow in line with recent trends, this could add not so small burden on oil needs. But even as the share of electric vehicles in the global car fleet grows rapidly, some scenarios, for example [7], predict that in 2040 ICE technology will retain almost half of the global market share in passenger cars and light commercial vehicles, e.g. city buses. However, by 2050, it is impossible to buy a new passenger vehicle powered by an ICE anywhere in the world according to [5]. Although the fuel combination for passenger vehicles is rapidly switching to electricity (where liquid fuels are almost halved from now until 2050, and where electricity is expected to dominate by 2070), oil & gas are still deeply present in the transport sector. Other forms of transport, however, are sufficient to maintain the still high demand for oil, mostly due to current lack of low-carbon substitutes for aviation, shipping/marine, or heavy-duty trucks. Thus, overall energy demand in transport continues to grow [7], and globally transport sector continues to be dominated by oil [7, 8], whose projected use is around 85% by 2040 [8]. Oil remains the dominant source of energy because of the rising demand for transport, mainly long-distance [7].

It is clear that the increase in energy demand will affect every sector of social and economic life, from industry, transport, buildings, and electricity generation. Nevertheless, demand for oil & gas will continue to grow steadily over the next few decades (especially gas is growing strongly). According to all that has been said, it is not feasible to expect that fossil fuels will be set to margins or totally thrown from use, at least not in recent years to come. The range of scenarios

presented in [4, 5, 6, 7, 8] predict that in primary energy consumption oil & gas (without coal) will be nearly at 50% (today's level is at 57%). In total, fossil fuels continue to meet more than two-thirds of primary energy demand across many scenarios, e.g., see [7]. In one of the most optimistic scenario towards low-carbon energy presented by Shell [5], it is estimated that by 2050 nearly 45% of primary energy demand will still come from fossil fuels. Other reports foreseen similar or even higher share, for example see [7] or [8]. At the end, please note that even the share in the overall energy system falls (due to rapid penetration of renewable sources in some sectors), net production of oil and gas certainly will be higher by the mid of the century than today. Therefore, oil & gas will remain a substantial source of energy for decades to come in order to meet the rapid growth.

In addition, it is estimated that current reserves are sufficient for the next fifty years (given today's production and consumption) [9]. The estimate includes only oil & gas reserves that can be recovered in the future from known reservoirs under existing economic and operating conditions. However, there are still undiscovered sources of oil & gas waiting to be found, which is more than enough to power society and economic growth in a long run. Driven by the obvious need for additional fuel, it is up to oil & gas industry to produce advanced techniques to find and capture those complex, jet undiscovered, reservoirs, which often requires constant investments in the research and development of advanced technologies. To illustrate, these investments are expected to reach trillions of dollars [8]. Therefore, investment in new production will continue to be crucial to meeting society's ongoing oil & gas needs. Based on the projection in [8], although the exact outlook is uncertain, it looks like the world will consume significant amounts of oil & gas for decades, requiring substantial investments.

In order to adopt novel techniques, today's research in geoscience is technically complex and involve many complicated coupled processes, and it often requires cross-disciplinary knowledge. However, it is natural to expect that research and development of novel methods used in the acquisition and processing of seismic data will continue to attract the attention of the scientific community. In the past, the industry could only utilize a portion of the data recorded in 3D seismic surveys due to the high complexity and huge volumes. However, with constant improvements in compression techniques, and apparent increase in available computational resources, today industry is able to perform complicated tasks in order to extract as much information as possible from the obtained

data. Even so, further development of compression techniques within this area is expected, and highly appreciated. The research potential is huge, as evidenced by recent publications in this very niche field, e.g., see next Chapter 2.

To find a reservoirs they have not been able to see and reach, companies are developing more sophisticated geophysical methods. New seismic imaging, machine learning, and visualization methods have been adopted to enable the discovery of as yet undiscovered resources and to enable the provision of high-quality images of the subsurface that are necessary for geoscience and petroleum engineering experts to make the right production decisions. Therefore, the main purpose of seismic data is to provide detailed knowledge of subsurface parameters in order to localize a given target and in order to increase productivity. For example, sharper seismic images mean that oil companies can drill new development wells in deep reservoirs with greater reliability and accuracy.

With recent advances in oil & gas exploration, sophisticated high-density imaging methods have been used to create high-definition images of subsurface geology, allowing more accurate mapping of geological structures. As new technologies emerge, the size of the seismic data sets also becomes larger, such as 3D seismic data, deep subsurface seismic survey, and high resolution seismic imaging. Therefore, the industry creates vast amounts of data (both 2D traces and 3D wavefields). Some seismic images can have as many as 16 million pixels in one dimension, and most of them are high bit-depth images with 32 b/p resolution to cover a wide dynamic seismic range. For example, the amount of data processed in just one seismic survey usually exceeds the order of several tens of terabytes of raw data per day. It is important that seismic data compression technology is able to reduce a terabyte-sized dataset to a manageable volumes, in order to further facilitate rapid processing and interpretation.

In addition, in the oil & gas industry, the development and deployment of new technologies to improve productivity and reduce costs are becoming increasingly important. In seismic data processing, cost and productivity are highly related to the data volumes and data transfer speed between a computer system and storage disks, which relies on an efficient coding scheme. More precisely, a typical performance issue in modern computer systems is usually related with an input/output (I/O) bottleneck, which occurs in the data transfer rate between the storage disks (main memory) and the node memory due to the mechanical nature of storage disks, transfer bus speed limits, and memory bandwidths. In high processing computing, many hardware optimizations

and memory access scheduling have been introduced to provide high-speed interprocess communication links. The goal is to eliminate as much as possible disk I/O (which is inherently slow, since it is a mechanical process) and to optimize read/write processes. In addition, available size of on-chip data memory (fast, but expensive module) is significantly lower than the size of a datasets, requiring a lot of transferring operations between the on-chip memory and the node memory through the limited bus interface. Also, it is commonly know that, bandwidth from the disk to the end-user did not advance as did the computational speeds, e.g. computing power of central processing unit, making the read/write, i.e. input/output, of data more and more of a bottleneck. This is especially evident when it comes to large data sets, such as seismic data. I/O data flow in such a system can be tremendous, where the disk access speed is much slower than processing speed of the central processing units. Even with the use of fast memory modules that have emerged in a recent years as affordable solutions, e.g., solid state drives - faster alternatives to standard hard disk drives, the computational throughput of central processing units is still much higher than that of data access and data transfer speeds. Therefore, in such setting, the direct raw data access time greatly outweighs the time when data to be processed is coupled with an efficient compression/decompression algorithm. A strategy to harness the computational power of a modern multithreaded system, and to reduce data workloads through the utilization of an efficient compression scheme, can lead to a significant improvement in disk access time, e.g., to accelerate I/O processes. Thus, to reduce the impact of the I/O bottleneck, compression is typically deployed to overcome the I/O limitations, e.g., to reduce the data transfer rate within nodes in a computer system, sacrificing additional processing time to perform data decompression, but on overall reducing the total data manipulation time when compared to raw data access. In that way, we can overcome the limitations of disk and memory bandwidth, which occurs due to intensive I/O transfer operations that could consume above 80-90% of the total processing time [14, 15], and thus improve the performance of I/O intensive workloads in seismic data applications. Of course, in such case, an efficient and fast decoder is of great importance.

Clearly, as the data burden is expected to become even greater in the future with plans of new seismic surveys, there is an apparent need for an efficient coding technique that will meet high quality and high compression ratio requirements for diverse seismic data applications. As a matter of fact, the benefit of using an efficient compression scheme within seismic surveys is twofold:

1) significant reduction of overall storage size required for seismic data, while targeting the highest possible reconstruction quality, and 2) significant cost and time savings across the workflow where seismic data is generated, transferred, saved, copied and used. Thus, improving the compression scheme also leads to better image quality, which in turn leads to better interpretation and precise localization of undiscovered resources. Based on the above, in short, compression is often required to enable and facilitate the deployment of new technologies, to reduce required storage, power consumption, data access time, and processing time (for example in computer-aided subsurface detection using machine learning techniques). In addition, a turnaround time, from acquisition to processing and interpretation of acquired data, is also crucial in order to increase productivity and to provide cost savings. In the background, this is related to the implementation of an efficient compression scheme within the seismic data processing workflow. In such case, a new efficient compression scheme aims to enable timely evaluation and timely decision-making possible, and is expected to significantly reduce cycle time in exploration phase of a field by reducing the required data transfer time, e.g., via a limited satellite link. For example, seismic data transmission by satellite from vessel to an onshore data processing center rely on a good and effective coding scheme. Efficiency is important to facilitate rapid processing and interpretation in real-time while the vessel is still on prospect in the event questionable or interesting areas need reshooting. As compression scheme is efficient, it enables faster transmission via satellite without compromising the integrity or quality of the data. In terms of data access speed, it can enhance interactive application performance, allowing petroleum experts to focus their time on delivering insights and results.

One option for the industry is to license commercial compression software libraries for their compression needs in seismic processing. However, as seismic processing technology advances, the requirement for computing hardware and compression from new applications often change. Without ownership of the source code, the compression library cannot be customized unless paying more in addition to the licensing fee to meet the requirements arising from new seismic processing applications. Therefore, a more viable and flexible solution is to develop seismic data compression library by adapting state-of-the-art image or video compression standards' test model codes to meet one's compression requirements arising from emerging new technologies and their applications. This work contributes to the goal of finding such an efficient compression solution by proposing a novel scheme for seismic data compression under the framework of

HEVC [16]. Driven by the need of compression of seismic data and by the high compression efficiency of HEVC, this work studies the application of a 32 b/p extension of the HEVC codec. The decision to give preference to HEVC in this study comes from the fact that 3D seismic data can have highly correlated individual slices, resembling video signal, which makes them particularly suitable for HEVC inter mode application [17], besides the possible advantages of the (still image) HEVC intra mode application [18]. Such approach, to treat 3D seismic data as a sequence of frames in order to obtain higher compression gains by utilizing motion-compensated predictive codec, is opposed to still image coding approaches that are frequently used and advised to be utilized by other competing lossy compression techniques in related literature. However, it is important to note that, by using the intra coding mode, the resulting codec also can handle still image compression of 2D trace images, along with aforementioned 3D seismic data compression. In addition, HEVC is currently the latest widely accepted industry standard for video coding. It is designed to support high reconstruction quality with low bandwidth requirements, where complex features have been added to achieve better performance - but still it uses highly parallel design approach and it follows hardware friendly implementation [19, 20, 21, 22].

Moreover, the standardized version of HEVC accepts input data up to 16 b/p [23], and it is mainly developed to maintain a high compression ratio for the most common consumer video applications. Since seismic data use up to 32 b/p, the HEVC cannot be directly applied. It is also very important not to threshold seismic data prior to compression since some sensitive information may be lost, and this is the main reason why the use of standardized 16 b/p version of HEVC is not directly applicable. Thus, in this work almost all core components of the original codec are modified to propose a novel coding scheme for high bit-depth seismic data compression under the HEVC framework. HEVC uses a hybrid block-based approach, where the input image is divided into the non-overlapping square blocks, that may be additionally divided into the smaller blocks of variable sizes [24]. Furthermore, it utilizes intra and inter-predictive coding [17, 18], two-dimensional (2D) discrete cosine transform (DCT) and uniform quantization, [25] and context adaptive entropy coding in order to achieve efficient compression gains [26]. While the block division and block structure, as well as the prediction part of the proposed codec mainly remain the same, other parts were subjected to major changes in order to cater the targeted 32 b/p input. Standardized HEVC's transform has been replaced with the new lifting-based transform of flexible block sizes ranging from 4×4 to 32×32 pixels.

Quantization has been replaced with a uniform quantization scheme with an increased quantization parameter range. At the end, a modified context adaptive binary arithmetic coder (CABAC) with additionally improved throughput has been utilized for efficient entropy coding. Also, a new model for the Lagrange multiplier has been used in the Rate-Distortion (RD) optimization loop in order to accommodate the extended bit-depth range and to empower the extended quantization parameter range (as necessitated by 32 bit-depth).

After the proposed modifications are introduced, the standardized HEVC solution has been compromised. Nevertheless, this does not affect the functionality of the new codec. The goal was not to stay within the limits of HEVC, since within the HEVC design it was not possible to implement a version that would support 32 b/p extension, and at the same time does not violate the standardized codec. Even though the new codec, after implementation of the proposed modifications, goes beyond the standardized HEVC, it still maintains a generic HEVC structure, and it is developed under the general HEVC framework. Leaving behind the standardized version of HEVC allows to experiment with a wide range of new approaches and modifications of the original HEVC in order to propose specific codec design that improves coding gain over the existing solution by adapting it to the specific seismic data attributes. Also, by adapting the proposed codec to specific seismic data attributes, an optimized encoder is proposed that significantly reduces encoding time with negligible compression performance losses.

Each of the aforementioned coding steps will be described in the following, with a more detailed analysis of the modified or newly implemented parts. Obtained results using the proposed codec (in both inter and intra coding mode) significantly outperform the performance of codecs that are widely used in industry for seismic data, such as JPEG eXtended Range (JPEG-XR) [2], or licensed commercial wavelet-based codec [1]. Also, the subjective quality of the proposed codec, which was evaluated by Shell's geologists, confirmed highly satisfactory results.

Lastly, note that without the proposed changes to the original codec, direct application of HEVC in its standardized form is not possible. Due to its specific design adapted to standard consumer video applications, where certain losses are tolerable, some of the original components render a huge error when applied to extended bit-depth data (even without quantization). In this regard, the aforementioned modifications have been introduced to replace these critical codec parts. The proposed codec provides a suitable basis for further development that

could improve the coding gain by proposing new coding features, or optimizing existing ones in addition. Accordingly, the proposed codec is built from HEVC, whose components have been modified for a targeted 32 b/p input, in order to make it practically useful in the first place. To the best of author's knowledge, there is no similar work in the field of seismic data compression that uses the HEVC as a base codec setting. Also, known to author, there is no codec on the market for 32 b/p seismic data that exploits redundancy in all three dimensions for improved performance. Therefore, this work represents an initial effort to provide valuable insights of using one well established coding scheme, such as that given with HEVC, for the purpose of seismic data compression.

The rest of this work is organized as follows. A review of related literature, with an emphasis on lossy techniques as the dominant approach to seismic data coding, is given in Chapter 2. The general design of the HEVC is briefly given in Chapter 3. More detailed analyses of the coding methods that are used within HEVC, and the proposed solutions resulting from this work, are introduced in Chapter 4. This chapter represents the main contributions of this thesis. The experimental setup is highlighted in Chapter 5. In the same Chapter 5, the underlying data that has been used as a test set in this research has been briefly described, along with a description of the code configuration setup and performance metrics. In Chapter 6 the analyses of the obtained results are given. This chapter summarizes the final achievement of the proposed codec. The work is concluded in Chapter 7.

Chapter 2

Seismic Data Compression: An Overview

This chapter gives detailed overview of the existing coding schemes which appear in the related literature. Recent advances and exciting progresses in the field of seismic data compression are given below. The goal is to put at one place different approaches that have been published recently at one place, and hence help researchers to have better overview on the current progress in the field. In that sense, this chapter may serve as a tutorial or specialized survey. The focus is on lossy compression as it is the dominant compression method in seismic data flow.

Lossless compression is able to provide flawless reproduction of the originally sampled data, but unfortunately true lossless reproduction is mostly inefficient for a compact representation. Often there is interest in expanding such approaches to lossy compression, by which utilization higher compression gains are possible, however for the price of the inevitable and irreversible loss of information. In those cases, the utility of lossy compression scheme is mostly affected by the underlying data type and planed application, and it is also very dependent upon the system characteristics itself. Since seismic exploration commonly produce tens of terabytes of data (sometimes it could go up to the order of petabytes), the need for efficient lossy compression of high bit-depth data, which will significantly reduce data rates beyond the capabilities of lossless technology, is apparent. Thus, within a given (seismic data) scope, lossy compression techniques are preferable since data rates would put extreme burdens on recording capacity and communication link. Therefore, instead to yield perfect reconstructions, in certain circumstances it is much more optimal to make a compromise between loss of information and desired compression ratio, but

at the same time to allow reproductions that are satisfactory for a targeted applications. In that sense, many research has been done in order to find adequate lossy compression solution within the seismic data workflow that would be able to satisfy rigid requirements for a variety of seismic data applications. In fact, oil & gas industry has always been struggled to reduce collected data volumes from seismic surveys, and compression of seismic images has long been a subject of study when it comes to efficient storage and transmission [27].

Compression performance of traditional transform based methods is mainly affected by the transform's ability to decorrelate acquired data. In most cases, after decorrelation has been applied, in order to represent data in the most compact way, the usual subsequent steps that follow transform are adaptation of a properly designed quantizer, e.g., uniform, weighted, or frequency-adjusted, and efficient entropy coding, e.g., arithmetic, Huffman, or run-length coding. Several approaches using those principles have been compared in [1] and [28]. The key idea behind these traditional compression methods is to extract and retain only a small number of low-frequency coefficients and encode these while discarding the remaining ones.

Among many transforms, wavelet based approaches have played a dominant role in performing decorrelation of seismic data [29, 30, 31], recently appearing in its efficient form which use lifting based implementation [32, 33]. The popularity of the wavelet based coding scheme could be found in its efficient data representation in the transformed domain which easily allows compressed image manipulation, e.g., by utilizing straightforward quality control scheme or progressive image decompression. One such effective coding scheme, based on set partitioning in hierarchical trees (SPIHT) [34], was recently adopted to seismic data in [35], and also partially adopted by methods in [1]. Hereof, embedded coding structure of SPIHT has found application in many seismic compression schemes that possess hierarchical (multiresolution) decomposition structure, e.g., as with methods in [1] among others.

However, traditional wavelet-based approaches may not be well suited to the highly oscillatory nature of seismic data according to [1]. To alleviate this problem, and to get closer to higher compression ratios than those achieved with multiresolution decomposition using traditional wavelet basis, one can utilize wavelet packets or adaptive local cosines as proposed in [1,36,37] or in [38], respectively. Since collection of wavelet packets is overcomplete (there are many more basis functions than the dimension of the input space) one can construct a basis that is fitted for a target image (or for a class of images such as seismic

images) in order to gain the benefit of such a data-driven approach. Similarly, method in [38] is based on the approach of finding ways to optimally segment an image in terms of local cosine bases. Both aforementioned approaches depends to a large extent on the proper design of the cost function, which has the crucial role in the selection on an optimal coding decisions, e.g. how to select, among a large collection of bases, the basis which is best adapted to encode a given image. Gilded by similar modality to the references mentioned above, new methods were also presented in [1]. Also, hybrid of wavelet with either wavelet packet or local cosine are presented in [39]. Hybrid approach applies a wavelet transform in the horizontal direction and the local cosines in the vertical direction to better catch characteristic of seismic images, and to better retain all seismic events in reconstructed data. Principally similar, but methodologically different approach has been presented in [40], where cascade of compressions is applied successively to the image itself and to the residuals that resulted from the previous compressions. At each step a different transform, or basis, at different bit-rate, is applied (e.g. wavelet basis at first step and local cosine basis at second step).

Besides aforementioned approaches, other related literature considered cosine basis as a prolific decorrelation approach for seismic data, that proved to be suitable transform for oscillatory patterns [41]. Although sparsification of a data in such case is highly satisfactory, the lack of an efficient coding scheme, such as those used with wavelets, has put these approaches slightly aside. In part, this disadvantage has been overcome by method in [42], which is based on generalization of local cosine basis named generalized lapped orthogonal transforms (GenLOT) [43], by the fact that the proposed transformed coefficient can be reorganized to fit the dyadic wavelet-like structure. Thus, the quality control feature and progressive coding can be easily implemented using embedded coding scheme, similarly as in [39]. In addition, GenLOT reported improved performance over the existing coding approaches at that time. Furthermore, work in [44] use generalized unequal length lapped orthogonal transforms (GULLOT) [45] for seismic data compression in order to reduce ringing artefacts that may occur with GenLOT, by which they achieve improved performance on less smooth signals such as seismic shot gathers. It was achieved by reducing the length of the high-pass filters, while keeping the same length for the low-pass filters. Conceptually similar work based on sub-band decomposition using different filter-banks, and other related transform-based approaches, can also be found in [46, 47, 48].

As datasets get larger, and their efficient managing more heavier, compression of seismic data has continued to draw attention since the seismic surveys began. As a consequence, novel approaches that increasingly take into account the nature of the seismic data (e.g. seismic image geometry) and high demands of new forms of seismic analysis (in order to preserve all significant seismic phenomena and structural information in the reconstructed images), and that are able to sparsely represent seismic data at the same time, have been widely studied in related literature. Among many, we point to covariance based decorrelation method such as principal component analysis (which is also known as Karhunen-Loève transform) [49,50], generalized Radon transform [51], or work in [52] based on regularized nonstationary autoregression, or locally adaptive wavelets [53], and a broad family of wavelet-like functions named x-lets, such as bandlets [54], seislets [55, 56], brushlets [57], dreamlets [58], etc. Curvelets, [59,60,61], due to their redundant representation in transformed domain (up to 7 times more coefficients than pixels) have not been used widely in seismic image compression (although redundant information may be beneficial in some other fields such as denoising or amplitude recovery). However, second generation curvelets, or so called symmetric curvelets [62,63], have overcome this problem and have found its application in seismic image compression. The same stands for contourlets [64,65], that possess up to 33% redundancy which is undesirable for compression. Thus, a critically sampled directional multiresolution image representation (CRISP-contourlets) has been proposed to overcome redundant expansion, and to be more applicable to the compression tasks [66]. Such a wide range of new class of bases result in various new signal decomposition (and compression) designs, although they have been based on similar principles of sparse representation and coding of seismic data. As a matter of fact, what distinguishes one approach from another is how discriminating it is, and how signal coding in sparse domain affects seismic signal's geophysical characteristics. The more it takes into account the characteristics of the particular signal it is compressing, the more efficient sparse representation could be. Thus, previously mentioned seismic data oriented decorrelators have tendency to suppress traditional transforms due to their efficient (and sparse) representation of seismic events in transformed domain.

Dictionary learning is becoming an attractive method for still image compression [67, 68], which is a significant change in approach compared to previously mentioned conventional decorrelation techniques. Its promising results gives further motivation to apply it to the seismic data [69, 70, 71, 72, 73, 74, 75].

In such approaches, sparse representation is obtained by learned (data-driven) dictionaries, which shows improved compression results, and where performance depends on how well dictionary can represent the data. Dictionaries are firstly learned on-line or off-line from a (usually but not necessarily larger) set of images with similar characteristics, and hence compression gain could be improved compared of that of the predefined over-complete dictionaries given with a clear mathematical formulations, such as those aforementioned generic dictionaries (e.g. wavelet, cosine basis, curvlets, etc.). Even though, after sparsity has been achieved with a trained dictionary, those methods do not differ in essence, meaning that proper quantization of transformed coefficients followed by the entropy coding should be applied, e.g., by using arithmetic coding [71] or [73]. Also, it is to emphasize that besides sparsity of the transformed data, rate-constrained criterion should be included in the learning process, since not all sparse representations are well compressible in terms of entropy coding (optimize bit-rate and sparsity jointly, instead to only optimize sparsity of seismic signal representation), see works in [72, 73, 74]. However, as reported in [68], these methods are still inferior to the HEVC which is baseline method of our study, and further development of dictionary learning based methods is expected in the future.

As stated in [76], conventional surveys have been designed to acquire data with uniform temporal and spatial sampling that honored Nyquist requirements. With a recent proliferation and remarkable progress in compressed sensing (CS) theory [77,78,79], we can liberate of this theoretical limitation, go below Nyquist sampling rate, and take a turn towards a new way of acquiring a signal with less measurements and reduced computational cost. Following new paradigm, saying that sparse signal (sparse with respect to some apriory known transform domain) can be fully described by small number of compressed sampling measurements, compression can be obtained during the acquisition process itself. In that way, a turn has been made from standard signal acquisition (where ever-growing dense sampling requirement can become prohibitively expensive) to signal recovery by using under-sampled data. However, even CS principles guarantee that signal can be fully recovered from its non-uniform (and random) sparse measurements (incomplete data) under certain conditions, it does not say exactly how. Therefore, research based on CS theory has attracted a lot of attention, where sparsity (and compressibility) has been promoted by using the properly

designed basis, e.g. curvelets as in [80] or dreamlet as in [81], and where an efficient large-scale optimization techniques have been developed in order to efficiently recover signal from its incomplete or sparse measurements, e.g., [80, 82] or other developments in solver technology such as [83, 84, 85]. Also, the benefit of applying CS principles in seismic explorations has been recently advocated in [76], and the references therein. Nowdays, in seismic imaging, CS concepts have been extensively studied in order to address challenges exposed in modern seismic surveys, not only to reduce data volumes, but also to improve image quality and reduce operational costs [80,86,87,88,89,90,91,92,93,94]. Notwithstanding, the authors of two separate studies, [94] and more recently [95], have concluded that the opportunities of CS in seismic exploration is still to be enhanced, which is mainly related to its practical realization in the field, but with an emphasis on great initial potential (see also [91]).

Deep neural network aided compression methods have been emerged recently as deep learning gained on popularity [96, 97, 98, 99, 100, 101, 102]. For example, one can utilize deep auto-encoders [96, 97, 98], or generative adversarial network in conjunction with compressed sampling theory as proposed in [99, 100], or it can be used to construct prediction signal after which difference of the original and predicted signal (residuals with lower dynamic range) are quantized and entropy coded like in in [101] and similarly as in [102] that utilize recurrent neural network. However, these approaches still need to mature in order to be widely accepted by the industry as a compression scheme choices.

In addition, some recent advances that demonstrated significant improvements in compression gain have been presented in [69], whereby the various schemes are mainly adapted to address different challenges arising in wireless seismic acquisition systems [103], which is also addressed in [104]. Also, another alternative techniques are reported, e.g., [105] that uses specific seismic data modeling, or [106] based on functional (vector) quantizaton.

Lastly, a compression technology that has been in use for many years by the industry is the licensed commercial wavelet based scheme proposed in [1]. It allows easy quality and rate control, and its computational simplicity and compression performance efficiency outperforms many other seismic compression schemes available at the market. Its compression performance is competitive with that of JPEG-2000 [107], which is also wavelet based codec and which uses similar lifting-based implementation as proposed in [33]. Another recent standard image coding scheme JPEG-XR [2], whose aim is low-complexity compression of high dynamic range images, has been utilized by the industry

to compress 1-D and 2-D seismic data [108]. Although JPEG-XR cannot compress all 32 bits of an input image (up to 24 b/p lossless and up to 32 b/p lossy), low complexity makes it a good candidate for commercial use, since it is friendly to heavy workloads and hardware with low configuration. However, it is only limited to still image compression. JP3D, an extension of the JPEG-2000, has been proposed to add support for 3D volumetric datasets [109]. However, it has not been considered as a compression scheme for seismic data applications so far.

One option for the industry is to license commercial compression software libraries for their compression needs in seismic processing. However, as the seismic processing technology advances, the requirement for the computing hardware and compression from new applications often change. Without the ownership of the source code, the compression library cannot be customized unless paying more in addition to the licensing fee to meet the requirements arising from new seismic processing applications. Thus a more viable and flexible solution is to develop seismic data compression library by adapting the state-of-the-art image or video compression standards' test model codes to meet one's compression requirements stemming from emerging new technologies and their applications. Following the same idea, this work emerged. Thus, this work reports application of one well known compression scheme, such as HEVC, to compression of high bit-depth seismic data by using both intra (still image) and inter (predictive) coding. HEVC is hybrid codec, meaning that it utilizes both prediction scheme to reduce dynamic range of the input signal, and transform to sparsely and compactly represent residual data obtained after prediction, followed by the uniform quantization and arithmetic coding. An additional comprehensive feature set has been developed as a part of the standard for the purpose of representing the final bit-stream as compactly as possible.

As we can see, there are many approaches (sometimes fundamentally different) to seismic data compression in the related literature. In common to all of them is however the fact that they strive to enable more efficient management and storage of this type of data. Some of them are already integrated in the seismic processing workflows, and some of them form a basis for future exploration in the domain of seismic data compression that have tendency to be integrated in the future seismic data processing and interpretation systems. Anyhow, due to rapid changes in the hardware capabilities and compression techniques itself, it is crucial to provide constant improvements and extensive research in this domain, in order to meet the ever-growing needs of the industry for efficient data

handling. Thus, seismic data compression is active research topic today, and we can expect many new attempts to improve current techniques in the coming years in order to push the boundaries of today's approaches.

Chapter 3

High Efficiency Video Coding in a Nutshell

Demand for the high quality and high resolution videos is growing rapidly. At the same time, this triggers a great demand for continuous improvement of coding algorithms in order to deliver the same content to consumers using fewer resources and lower data consumption. Content providers such as Netflix, Amazon, and others create a huge amount of data volumes, where most of the data has been streamed over the network, thus consuming the majority of the internet bandwidth. This was recently confirmed in a white paper by the Cisco Visual Networking Index [110]. The forecast estimates 4-fold increase in the IP video traffic by 2020, where video will account of 82% of all IP traffic globally. For example, live video will increase 15 times from 2017 to 2022, consumer video-on-demand will almost double in the same time period, internet gaming traffic will increase 9 times, and internet video surveillance will increase 7 times. An efficient coding scheme is also needed to deal with the massive data transmission demands resulting from 5G and ultra-high-bandwidth-low-latency streaming. Therefore, the need to constantly develop new coding algorithms to support existing and/or future infrastructure is obvious, necessary, and unavoidable. The HEVC, also known as H.265, is the latest and most advanced video standard at the time being, jointly developed by Video Coding Experts Group (VCEG) and Moving Picture Experts Group (MPEG), gathered around one group - Joint Collaborative Team on Video Coding (JCT-VC) [16]. It utilizes video compression features needed to meet the immediate and rapidly growing needs of the market.

The first version of the HEVC standard was released in early 2013, primarily focusing on typical consumer videos. It was later followed by the second, third, and fourth versions, which became available in 2014, 2015, and 2016,

respectively. They provided some additional functionality regarding range extension (16-bit support, various chroma sub-sampling formats, etc.), scalability extension profiles, multi-view extension profiles, 3D video formats, screen content coding mode, and they applied other new features and enhancements [23, 111, 112, 113, 114]. In general, the overall goal of HEVC was to achieve significant bit-rate reductions compared to existing codecs on the market across the wide range of use-case scenarios. In that sense, HEVC represented a major leap forward in video compression technology as it achieves higher compression ratios than its predecessors, while maintaining the same video quality, e.g., compared to H.264/AVC [115]. In fact, it can be said that HEVC has emerged from H.264/AVC, following the same general coding approach as H.264/AVC. Also, it can be said that HEVC is an extension of the concepts introduced in H.264/AVC. It was reported in [116] that HEVC intra coding (still image compression) reduced the average bit rate by 17% compared to H.264/AVC, 23% compared to JPEG-2000, 30% compared to JPEG-XR, and 44% compared to classic JPEG. Taking into account predictive coding, e.g., motion-estimation (ME) driven inter prediction, the compression gain goes up to 50% compared with its predecessor (H.264/AVC). It also supports much higher video resolutions, including Ultra-high-definition (UHD) 4K and 8K. Also, note that the HEVC codec accepts 8-16 b/p input data.

Although initially HEVC was designed to support the high demand for the typical consumer video streaming and storage, later versions provided the ability to use HEVC in many other applications. Some applications that require intensive compression, and can fit within a typical scope of HEVC use-cases, are screen content coding (e.g., remote desktop applications, remote gaming, wireless displays), medical image coding (e.g., ultrasound, radiography, medical X-rays), remote sensing image coding (e.g., multispectral, hyperspectral, or synthetic aperture radar (SAR) data compression), light field image coding, light detection and ranging (LiDAR) data coding, etc. [114, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133]. Also, for example, initial work where HEVC is used as a base for seismic image compression is given in [134, 135, 136]. These three publications essentially represent the works from which this thesis emerged.

HEVC uses hybrid block-based approach, where the input image is divided into the non-overlapping square blocks, and which are further recursively divided into the variable size blocks that forms the quadtree structure [24]. Furthermore, it utilizes intra and inter prediction [17, 18], 2D transform and uniform

quantization [25], and context adaptive entropy coding (context adaptive binary arithmetic coding) in order to achieve efficient compression gains [26]. In general, block-based video coders with a hybrid architecture use a combination of multiple approaches which have previously been shown to be the most commonly used and most reliable methods for decorrelation. Hence the name *hybrid coder*. The general and most commonly used approach is the one that combines prediction, e.g., differential pulse code modulation, and transform coding, all in order to remove redundancy as much as possible from the input data. Input samples are spatially or temporally decorrelated by generating a difference between the input signal and the prediction signal obtained from the previously coded samples. Transformation of the resulting difference signal (residual signal) into a more suitable form, for more efficient compression, additionally decorrelate the data by projecting the residual samples onto orthogonal base vectors. In addition to redundant data removed in this way, the quantization of the transformed coefficients eliminates less significant signal information. Finally, entropy coding based on conditional symbol probabilities, which are given in a specific context, is used as an encoding scheme to further remove coding redundancy. In this way, total data used to represent the input image is significantly reduced.

Figure 3.1 shows a general diagram of HEVC's hybrid block-based encoder and decoder[1]. The input frame is divided into blocks b. In HEVC, block b is represented with the largest block called Coding Tree Unit (CTU). It is possible to recursively divide each CTU into the smaller Coding Units (CU), that in its structure can incorporate one or more Prediction Units (PU) and Transform Units (TU). More about block structure in HEVC will be said in the next Chapter, and at this point more detailed overview of the HEVC's block structure is omitted. The residual signal r is obtained as the difference between the input b and the prediction signal p (intra or inter predicted). Transformation of the block r into more suitable representation for compression, and quantization of transformed coefficients (TR+Q module), determine the values of q (quantized levels), according to which block samples will be presented in the output bitstream after entropy coding. It should be noted that HEVC supports discrete cosine and sine transform, where in its standardized form HEVC applies integer approximations of those two well known decorrelation approaches. Since the encoder includes all modules that are also included in the decoder (shaded

[1] Image is slightly modified image from Mathias Whien's book - *High Efficiency Video Coding: Coding Tools and Specification*

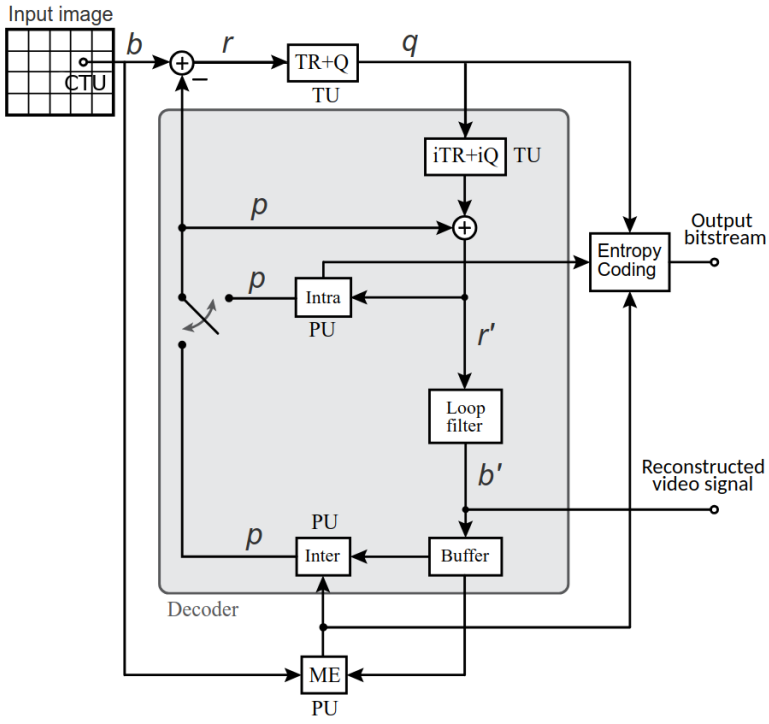


FIGURE 3.1: A typical block-based hybrid video coder scheme.

by the gray part in Figure 3.1), an accurate reconstructed image is available on both sides (if we assume error-free channel between encoder and decoder). This means that the decoder is capable of calculating the same prediction signal p as it was calculated at the encoder. Also, since the scheme in the Fig. 3.1 represents the general lossy coding scheme (although HEVC supports lossless mode), the reconstructed blocks b' and the reconstructed residual signal r' are not identical to b and r. The reconstructed block b' is obtained by transferring the $r' + p$ (after inverse transform and dequantization iTR+iQ was performed to obtain r') through a loop filter, which serves to correct the compression distortions. In addition, coder retains reconstructed data in a buffer in which reference frames are stored to be available for further predictive coding (inter prediction).

Many of the high-level concepts mentioned above are complex when we look down into their implementation. They include many advanced algorithms and specially designed solutions, all with the aim of improving compression

gain. Of course, these algorithms are also well balanced in terms of computational complexity to allow practical applications of the standards. Also, many different levels of details in the design of these concepts are included in the related literature, as well as in this work. Some of these details, which are closely related to this work, and which are relevant for understanding the proposed changes and modifications (to cater 32 b/p extension), will be presented shortly in the next Chapter. An overview of the basic concepts will be given below, and details will be slightly introduced. After the details are introduced and the proposed changes are described, they will be supported with the experimental results and comparison with other methods. However, it should be clear that due to the high complexity of the standard, some specific details have been omitted from the analysis. In any case, appropriate references are given for further reading.

Chapter 4

Methods and Proposed Solutions

In this work, in order to meet the ever-growing requirements arising from new high-density seismic surveys, as a starting point, the standardized and ubiquitous, industry accepted HEVC solution is used. This work has been guided by the idea that a more viable and flexible approach for industry is to develop a seismic data compression library by adapting state-of-the-art image or video compression standards' test model codes to meet one's compression requirements stemming from emerging new seismic technologies and their applications. It is proposed to reassemble the seismic slices in a format that corresponds to the video signal and benefit from the coding gain achieved by HEVC inter mode, as opposed to the still image coding approach that is frequently used and advised to be utilized by other competing lossy compression techniques in the related literature. Inspired by HEVC's high performance, a new codec for 32 b/p data compression under the framework of HEVC is proposed in this chapter. The general coding design and modifications that are introduced in the new codec, in order to accommodate the extended bit-depth range, are given in the following subsections, including the motivation that leads to selected design. This chapter also points to the parts that are believed to have space for further development and improvement.

At this point, it is worth mentioning that the high-level syntax has remained almost unchanged. The emphasize is on video coding techniques (only core algorithms of the Video Coding Layer are adopted and modified in order to meet seismic data compression requirements), although some overhead in high-level syntax can be removed. However, that overhead and the bits it produces into the final bit-stream, are negligible compared to the total number of bits. As it is going to be shown, some of the HEVC's features are never utilized it this work. For example, one such feature is transform skip, among several others.

Therefore, the syntax elements related to that feature have no specific purpose but still are located in the final bit-stream. However, as already pointed out, these elements belong in a small percentage to the final bit-stream, and their impact on overall performance is totally negligible. As mentioned in the introductory part of this thesis, however, the proposed codec after all is not HEVC compatible. Therefore, it is indeed possible to further tighten the bit-stream syntax in order to save additional bits. However, emphasis of this work was on algorithmic changes and improvements, and not on standard's syntax redefinition.

In this work, we look at the standard from the encoder side, even the standard itself defines the syntax of the bit-stream. This means that the standard specifies only how to decode the content without specifying how to encode it. Thus, the standard defines decoder side, and encoder side has left to the implementer as long as the bit-stream is standard compatible. The limitation on the bit-stream syntax has one advantage – it gives freedom to optimize the implementation of the codec in a manner suitable for specific application, taking into account required encoder complexity and available resources vs. desired compression gain and reconstructed video quality. Therefore, the given analysis is observed from the encoder side, however in this study a conforming decoder has also been developed in order to obtain the reconstructed data after compression. Just to note that this approach where coding algorithms are analyzed on the encoder side is not uncommon. Even if the standard defines a decoder side, a decoder without an equivalent encoder cannot be operative, so these two sides are always closely related.

In addition, it should be noted that this work examines the lossy coding scheme and the use of lossy techniques for seismic data compression. Seismic data applications can tolerate some loss of information, and therefore lossless techniques will not be covered by this work. Even though HEVC supports lossless compression, this type of coding will not be considered.

Before going into more details, it is to note that some codec's components such as those given in Chapter 4.1 and Chapter 4.2 mainly remain unchanged and less modified than other parts. However, those components are greatly utilized during encoder complexity optimization which is presented in Chapter 5 and Chapter 6. In that sense, for clarity purpose, they are also described in more detail.

4.1 Block Structure

HEVC uses a block-based hybrid approach, where the input image is divided into non-overlapping square blocks that are further recursively divided into variable size blocks to form a *quadtree structure* [24]. The quadtree concept is not new, and it has been considered in academic work in the past [137, 138, 139]. However, with the HEVC's call for proposal, it was used for the first time in the practical implementation of video coding algorithms [140, 141].

At a high level, HEVC divides the image into a grid of the non-overlapping Coding Tree Units (CTU). CTU usually consists of three coding blocks, namely the luma (Y) block and two chroma (Cb and Cr) blocks, and associated syntax elements. Each block is called a CTB (Coding Tree Block). In such way we have: $CTU = CTB_Y + CTB_{C_b} + CTB_{C_r} + Syntax\_Elements$. Luma CTB has the same size as the CTU, while the size of the chroma blocks depends on the subsampling scheme (usually $2\times$ smaller by each dimension for the most common format 4:2:0). Since in this work focus is on seismic data that are represented in the form of monochromatic images (consisting only of luma channel), we are going to consider only the luma block CTB_Y, and in the sequel we will ignore the chroma components/blocks. This is equivalent to a 4:0:0 chroma subsampling format, where there is only luma component. This format is a part of the range extension version of HEVC, and it is supported in HEVC reference software. In such way, we can reformulate notation as: $CTU = CTB_Y + Syntax\_Elements$. Thus, the units in HEVC are associated with local regions of the image, e.g., in our case luma samples, plus additional syntax elements, together. Contrary, a block refers only to pixel samples without syntax elements. For example, for seismic images that have only one image component (luma component), the block will only be associated with a luma component consisting of a 2-D array of image samples (pixels), and the unit will be equivalent to a block plus additional syntax elements. In other words, a block, along with associated syntax, form the units. Thus, in order to simplify notation, instead of units, in this work we always refer to blocks, since they are directly associated with the image information (pixels) on which calculations are performed. At this point, it was important to clarify notation to avoid possible confusion, as the related literature uses both, units and blocks, without proper explanation.

Furthermore, one of the main contributions of HEVC is the expansion of the block size, e.g., from 16×16 pixels in H.264/AVC [115], up to 64×64 pixels

in HEVC. On the other hand, small details are still important, and sometimes it is useful to perform coding at a lower granularity, for example, using smaller blocks of 4×4 pixels. When the block covers part of the image with a flat texture, a homogeneous region, the use of a larger block size can be beneficial since it can achieve higher compression gains. This could be particularly appreciated in high resolution images, such as seismic images. For a region of complex texture, smaller blocks may have more accurate prediction, but they require more side signaling bits. In this case, the variable block size provides ability to the encoder to adapt to the image characteristics, and consequently to improve the coding gain. Therefore, the improved and highly flexible quadtree structure is used to get the optimal block partitioning.

To support such an improved and highly flexible quadtree structure, HEVC additionally introduces different block types. These are Coding Blocks (CB), Prediction Blocks (PB), and Transform Blocks (TB) [24]. As already mentioned, at a high level, HEVC divides the input image into the grid of non-overlapping Coding Tree Blocks (CTB). The CTB can be of size of 64×64, 32×32, or 16×16 pixels. The size of the CTB must be defined before encoding starts. Once selected, the CTB size remains the same, and it is signaled as a side information within an additional HEVC syntax element. The size is chosen based on the planned application type and available resources, and it is usually a matter of compromise between complexity and desired performance.

Furthermore, the CTB can be recursively divided into multiple coding blocks (CBs), see Fig. 4.1 and Fig. 4.2. Each CB becomes a decision point for the type of prediction, where two types of prediction are defined in the HEVC – intra and inter prediction. Supported CB sizes start from the same size as CTB to as small as 8×8. One possible quadtree partitioning is given in Fig. 4.1 and in Fig. 4.2, where both figures are equivalent and represent the same splitting decisions that are illustrated in a different way. Starting from the size of the CTB, which is equivalent to $depth = 0$, the encoder can recursively divide each block into 4 smaller square CBs with half the horizontal and vertical size of the parent block by descending further down on multilevel depth hierarchy. The split can be continued for each newly derived CB until the maximum possible depth, or equivalently the smallest CB size, is reached. Partitioning is content dependent and it aims to capture the characteristics of the image region covered by the CTB, while relying on RD optimization (more about RD optimization can be found in Sec. 4.6). The resulting tree is called *coding quadtree*, and for example, in Fig. 4.2 it is shown in solid line. The leaf nodes represented as dots

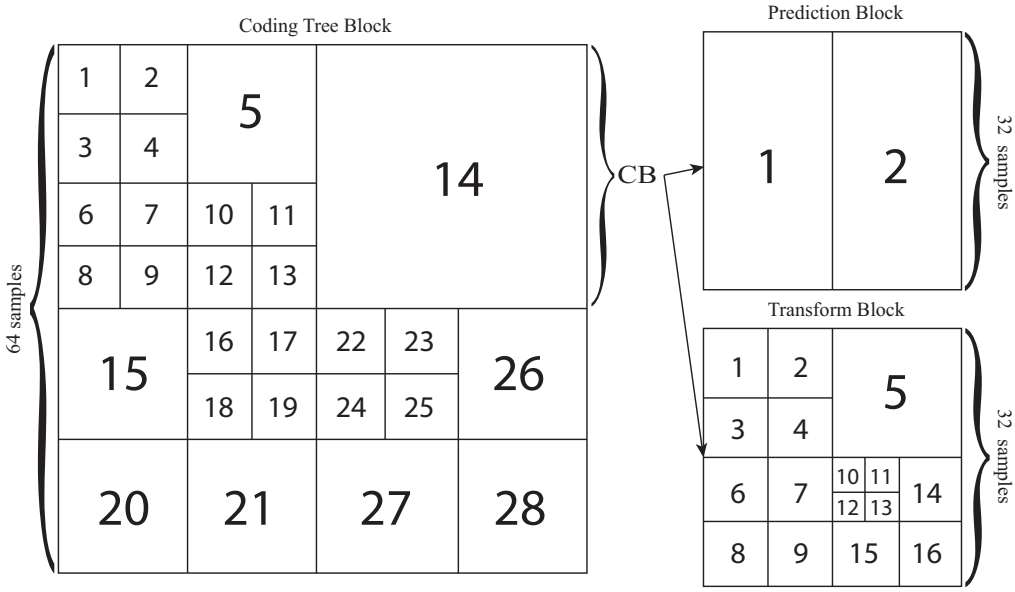


FIGURE 4.1: An example of a variable block size used in HEVC. The illustration shows how 64 × 64 CTB can be divided into CBs using a quadtree structure (left side), and how each CB can additionally include PBs and TBs in its structure (right side).

are the final decision points for the CB size.

To carry out various prediction parameters, a prediction block (PB) has been introduced. Although an intra or inter prediction type decision has been performed at the CB level, more precise prediction parameters are carried out within the prediction blocks. In the standard, the PB partitioning is related to the proper selection of the partition mode. As it can be seen from the Table 4.1, the partition mode is associated with whether prediction mode is identified as intra or inter within the CB. Square, rectangular, and asymmetric rectangular partition modes (PB shapes) are supported in HEVC. Hence, a CB may consist of either one PB with the same edge as the CB, two rectangular PBs, or four square PBs, and the size of the PB cannot be less than 4 × 4 pixels. Thus, each CB may include one or more PBs in its structure. Starting from the CB as the root point, the PB can be the same size as the CB, or it can be divided further, considering limitation on a PB size as defined in standard with respect to its parent CB. In this sense, the CB partitioning on PBs is restricted, and may consist

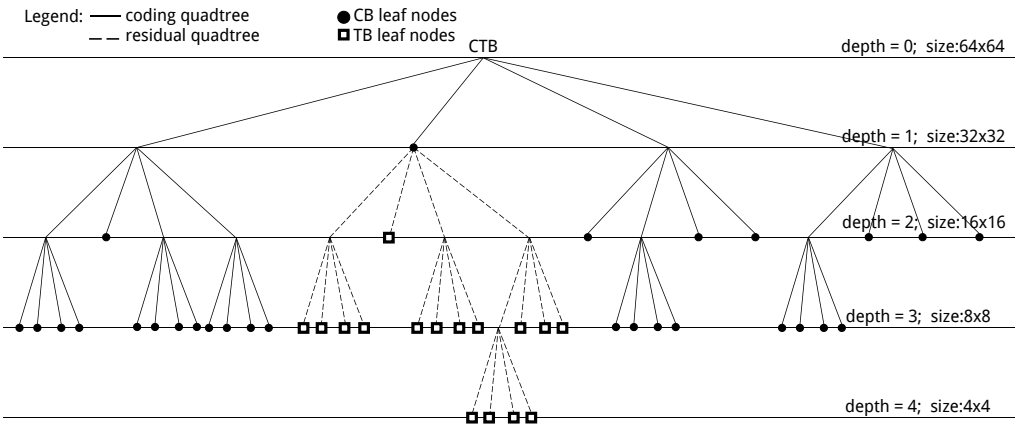
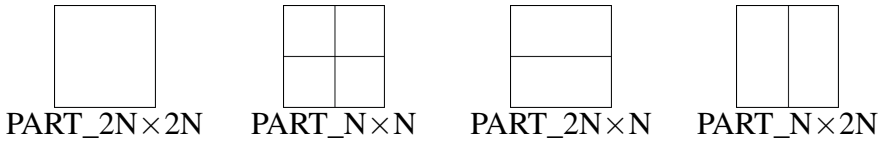


FIGURE 4.2: Another illustration of a hierarchical multilevel depth structure. The solid lines represent the coding quadtree, and dashed lines represent the residual quadtree.

of either one PB with the same edge size as the CB, two rectangular PBs, or four square PBs, see Table 4.1 for illustration. Also, PB size cannot be smaller than 4×4. One example of PB partitioning is given in Fig. 4.1. In intra mode, the size of the PB relative to the parent CB can be the same as the CB, which is given with the PART_2N×2N partition mode, except when the CB is the smallest allowed size (8×8 pixels). In that case, the CB may contain 4 smaller square PBs of size 4×4 that can have their own intra prediction mode, which is equivalent to the PART_N×N partition mode. In inter prediction mode, eight partition modes are defined (see Table 4.1), including non-square modes (PART_2N×N and PART_N×2N), and asymmetric non-square partition modes (PART_2N×nU, PART_2N×nD, PART_nL×2N, and PART_nR×2N). In case when the CB consists of two or more smaller PBs, each of them may have its own specific prediction parameters. However, a combination of intra and inter prediction is not possible in a single CB. It is important to note that the partition mode defines the shape and size of the PB based on the size of the parent CB. Once selected, the partition mode is final, meaning that the PB can be only split once, and it does not allow further hierarchical depth division. Note that N in the Table 4.1 takes values from $\{32, 16, 8, 4\}$. Also, the use of asymmetric modes is optional, and it can be disabled by using additional syntax within the bit-stream. This part will be examined in the proposed optimized encoder for seismic data compression.

TABLE 4.1: Prediction Block (PB) partition in HEVC's intra and inter prediction.

Prediction Mode	Partition Mode
Intra Prediction	PART_2N×2N PART_N×N
Inter Prediction	PART_2N×2N PART_N×N PART_2N×N PART_N×2N PART_2N×nU PART_2N×nD PART_nL×2N PART_nR×2N



In order to better utilize the transform in HEVC, a transform block (TB) has been used, since the CB may be too large to be represented as a structure for transform, as it may contain both a detailed parts (high frequencies) and a flat parts (low frequencies). The root node of a TB partition is a leaf node of a coding quadtree. Hence, TB is still dependent on CB, but starting from CB level as a root, it can be recursively divided into 4 smaller square blocks. For example, look at the right-bottom of the Fig. 4.1. Thus, TB has its own quadtree partitioning, where allowed TB sizes go from 4×4 to 32×32, supporting only a square transform, and where TB size does not have to match PB size. The resulting tree is called *residual quadtree*. In the Fig. 4.2 it is given by a dashed line, and the leaf nodes represented as squares are the final decisions on the TB size. Therefore, each CB has been further partitioned into the associated PBs and residual quadtree of TBs. However, more about prediction and transform will be said in the following sections.

In the Fig. 4.1 and Fig. 4.2, a CTB that has 4 levels of hierarchical depth division has been illustrated. The root node represents the CTB, and the leaf nodes illustrate the final CB and TB partitions. Furthermore, each CB leaf node

is the starting point for the PB partitioning and the TB residual quadtree. The root of the CTB in the given example corresponds to $depth = 0$, and has the size of 64 $\times$ 64. At $depth = 1$, the CB size is half of the CTB size (e.g., CB number 14). At $depth = 2$, CB size is one quarter of the CTB size (e.g., CB number 5 or number 20 among others), and at $depth = 3$, the CB size is 8 $\times$ 8 which is the lowest size defined by the standard for CBs (e.g., CB number 1 or 25 among others). The maximum hierarchical depth level relative to the root CTB is a configurable parameter and it is defined prior to encoding and signaled as the side information. Depending on the application, different combinations of maximum CTU size and maximum quadtree depth can be specified, e.g., 64 $\times$ 64 and depth 3, or 32 $\times$ 32 and depth 1. A similar approach can be applied to the *residual quadtree* hierarchical depth level relative to its parent CB. The relative depth of the residual quadtree, starting from the leaf node of the coding quadtree, is also a configurable parameter. Note that the maximum total depth for TB partitions is $depth = 4$, which is equivalent to 4x4 size, the minimum size allowed for the transform in HEVC. Also note that CBs within the nested quadtree are processed in the raster scan order, that can be seen in Fig. 4.1.

It is important to note that in this research the complete block structure and associated syntax of the HEVC standard has been kept. The motivation lies in the assumption that a given block structure is well suited for seismic images as it is for natural video, as they both can have large uniform regions that can be compressed by using larger block sizes, and yet contain regions with lots of detail that can be compressed by using smaller block sizes (which in turn maintains the ability to retain important seismic features). Since block partitioning is not directly associated with the extended bit-depth range, or to the fact that seismic data is used instead of natural video, there is no need to modify this part. However, we have underwent additional experiments that aimed to show performance when some block size restrictions are introduced, e.g., see Chapter 6. These experiments are related with a limited maximum block size below the standardized maximum of 64 $\times$ 64 pixels, or related with a increased minimum block size of 4 $\times$ 4 pixels. Since those block sizes are chosen to be the default values in the standard to get the best performance for the widest possible range of applications, the question is whether these default values are also the best solution when compressing seismic data. For example, it would be useful to see if we can choose to limit the minimum CB size to 8 $\times$ 8 instead to 4 $\times$ 4 pixels. Also, for example, we can experiment with a limited hierarchical depth division

of the residual quadtree, or to use limited partition modes, e.g., exclude asymmetric partition modes from consideration. In addition, it is possible to reduce the available transform sizes. In this way, the usefulness of a standardized block partitioning scheme can be tested, or on the other side, if experiments show negligible changes in performance, we can choose to adjust the encoder according to these findings and to unburden the encoder of unnecessary calculations. These experiments are analyzed in detail in Chapter 6. As a subject of further research, block partitioning can be considered in the context of reducing algorithmic complexity, similarly as it was applied for natural video [142, 143, 144]. The underlying idea is that the collocated blocks at the same position in consecutive seismic slices are highly correlated and most likely have a very similar content, structurally and statistically, mainly due to the low motion characteristics of seismic data. Thus, some optimization scheme that will speed-up block partitioning based on previous decisions can be utilized in this part of the codec. However, advanced complexity reduction is not the part of this work.

4.2 Intra and Inter Prediction

Intra and inter predictions are two types of predictions that are used in HEVC [17, 18, 145]. Block based predictors are used, where in intra prediction the focus is on the reduction of spatial redundancy, and in inter prediction on the reduction of temporal redundancy. The underlying concept is not new and it is inherited from the previous standard (H.264/AVC). However, several improvements have been proposed in HEVC.

4.2.1 Intra Prediction [18]

Intra prediction of HEVC relies on spatial redundancy that is inherited in the data. Pixels in the same area of the image that are close to each other are usually highly correlated. In intra coding mode, only information from the surrounding CBs can be used to form a prediction signal. Therefore, by using intra prediction, each seismic image can be compressed without referencing to other images in the sequence. This is equivalent to the still-image compression approaches. This kind of spatial redundancy reduction is also referred as intra-slice prediction.

Only spatially adjacent samples that are located on the upper and left sides of the PB have been exploited to calculate the intra predicted signal. We refer to those surrounding samples as the reference samples $R_{i,j}$, see Fig. 4.3a. Note that a total $4N + 1$ reference samples can be used for PB of size N, and in any intra mode (see below about different intra prediction modes), only these samples are available when calculating the prediction block. Thus, each prediction mode uses the same set of reference samples. In certain cases, e.g., at image boundaries, some reference samples are not available, and they must be compensated in some way. Missing samples are created by repeating previously available samples by using the reference sample substitution algorithm that is defined in the standard. If, however, the reference samples cannot be reproduced in any way (when they cannot be reconstructed by repeating previously available samples), the nominal average value for a given bit-depth is used, e.g., 2^{31} for 32 b/p seismic data. When constrained intra prediction is enabled, reference samples belonging to neighboring *inter-predicted* PBs are omitted (to avoid possible error propagation in case of missing frames/seismic slices).

Also, when residual quadtree partitioning leads to situations where one CB is split into multiple TBs, intra prediction is applied for each TB separately, instead of applying intra prediction once at the PB level. However, for each TB it is applied using the same intra prediction mode over all TBs. This enables the possibility to always exploit the nearest neighboring reference samples from an already reconstructed TB (because the correlation between pixels is higher when they are spatially closer). As for the size of the PB relative to its parent CB, in intra coding, the PB may be the same size as the CB, except when the CB is the minimum allowed size. In the latter case, the CB may contain 4 smaller square PBs that have their own specific intra prediction mode, independent of each other, see for example Sec. 4.1 and Table 4.1.

In addition, HEVC provides 35 different intra prediction modes, see Fig. 4.3b. In order to more accurately model different textural structures, it utilizes 33 angular modes, along with DC and planar mode for more precise modeling of smooth regions. Angular predictions are computed with 1/32 sample accuracy, and the predicted value $P_{x,y}$ is the weighted function of two neighboring reference samples, e.g., $R_{i,0}$ and $R_{i+1,0}$, which are determined to be located at the direction of the given angular mode. Bi-linear interpolation has been used, where the predicted value $P_{x,y}$ can be considered as a linear combination of the two closest reference samples, e.g., by choosing two samples in the selected prediction direction after projecting the mode direction to the reference sample

locations with 1/32 sample accuracy. Thus, the prediction is given as

$$P_{x,y} = ((32 - f) * R_{i,0} + f * R_{i+1,0} + 16) \gg 5 \tag{4.1}$$

where $\gg$ is bit-shift operation. In this case, the predicted value is the weighting function of two neighboring reference samples which are accessed using the index i, and weighted by the fractional displacement factor f. Also, i and f are based on the angular parameter d which can have a value from -32 to +32, and which is mode dependent. The indices i and f are given with

$$\begin{aligned} i &= ((y + 1) * d) \gg 5 \\ f &= ((y + 1) * d)\&31 \end{aligned} \tag{4.2}$$

where & is bit-wise "and" operation. It is important to note that the given equations (4.1) and (4.2) apply to vertical modes, while for horizontal modes (e.g., those with indexes 2-18, see Fig. 4.3b) the same equations may apply, with the addition of swapping the coordinates x and y.

For smooth regions, DC and planar mode may be a better choice than directional modes. In the case where the DC mode is selected, the average value of the available reference samples is used to predict the entire block. In some cases, the boundary samples can be treated in a slightly different way (e.g., predicted values of the first row and column – on Fig. 4.3b represented by the values at $P_{0,0}$, $P_{0,j}$, and $P_{i,0}$ locations). In this case, the predicted DC values at the previously mentioned locations are post-processed by using two- and three-tap smoothing filters. Similar boundary smoothing operations can be applied to the horizontal and vertical modes [146]. In contrast, planar mode is designed to predict smooth gradient surfaces with gradual changes, by averaging two linear predictors, horizontal and vertical [147]. Planar mode is essentially defined as the weighted average value of only four reference samples, e.g., to predict $P_{x,y}$ it uses $R_{0,N+1}$, $R_{0,y}$, $R_{x,0}$, and $R_{N+1,0}$ as reference samples. It is given with

$$\begin{aligned} P_{x,y} = &\frac{y+1}{2N} * R_{0,N+1} + \frac{N-1-x}{2N} * R_{0,y} + \\ &\frac{N-1-y}{2N} * R_{x,0} + \frac{x+1}{2N} * R_{N+1,0} \end{aligned} \tag{4.3}$$

As an advanced feature, HEVC proposes to use a reference sample smoothing, which is reported to reduce artifacts in reconstructed images, based on a tree-tap

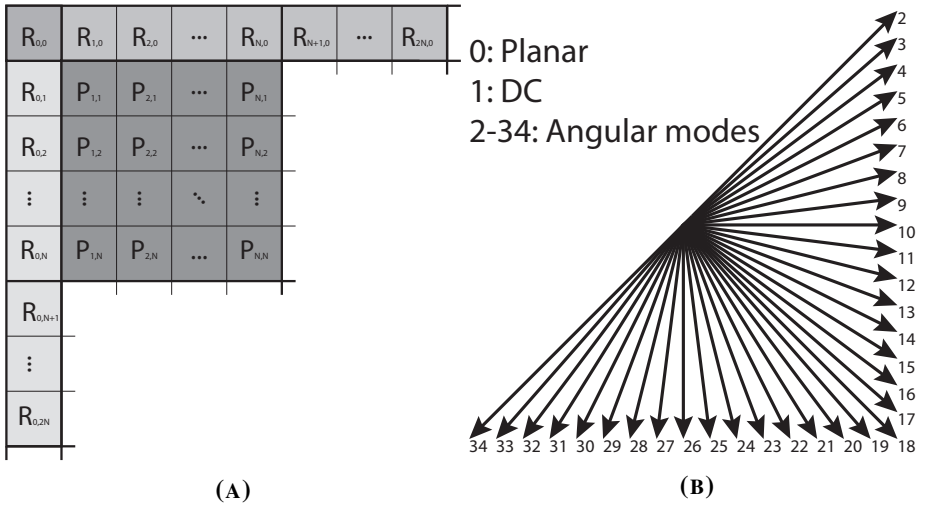


FIGURE 4.3: Intra prediction. (A) Reference samples $R_{i,j}$ used to form intra prediction signal $P_{i,j}$, and (B) Intra modes available in HEVC.

smoothing filter. This feature showed negligible performance on seismic data compression.

Also, HEVC allows efficient intra mode coding that increases the coding efficiency by signaling intra mode with minimal overhead. This technique is known as Most Probable Mode coding (MPM), which increases the symbol coding efficiency [148]. It uses intra mode decisions of the previous, surrounding blocks, and instead of transmitting the mode number itself, it transmits only the index from MPM list if the current chosen mode is in it. Intra mode signaling, MPM derivation, reference sample manipulation, intra prediction mode design, and mode derivation remains the same as in HEVC [18].

As we can see, for example, by looking at equations (4.1), (4.2), and (4.3), the intra prediction derivation is universal, and it is not limited to the dynamic range of input values. As long as we take care that the software implementation with its internal variables satisfies a wide range of dynamic values that can occur when using high-bit depth data, intra prediction in a given form perfectly matches seismic data. The given formulas and the standardized way of calculating the predicted samples can be used directly without any modifications. Therefore, in order to benefit from a standardized way of removing the spatial correlation, it was not necessary to introduce any modifications to the intra prediction calculations. Of course, besides already mentioned adaptations

of the reference software to support larger internal buffers/variables, which is not only related to this specific part of the codec, but follows the entire codec implementation (reference code) in each of its parts.

To avoid full RD optimization calculations, HEVC's reference software (HM) [149] applies a fast algorithm for intra mode selection [150]. In the first stage, it calculates the N the best intra prediction mode candidates using the sum of absolute Hadamard transform coefficients as the distortion measure D_{Had}, which is a simplified version of the brute force approach. For the rate R_{mode}, it uses the number of bits necessary to code only a specific, at the time considered, prediction mode. The Lagrange cost is then calculated as $J = D_{Had} + \lambda R_{mode}$. More about Lagrange cost and RD optimization can be found in Sec. 4.6. After that, in the simplified J encoder, HEVC considers N the best intra prediction modes for full RD optimization, where N is [9 9 4 4 5] for the block size [4×4 8×8 16×16 32×32 64×64], respectively. However, the optimal numbers for N candidates are provided based on experimental observations by using natural video. In addition, these simplifications negatively affect performance, which has been shown to be negligible for natural video. For seismic images the main focus is the high quality of reconstructed data, but with at the same time the highest possible compression gain. Since the implications of the simplified J encoder have not been analyzed yet for seismic data, the effect of the proposed simplified RD optimization is explored in this work during development of the seismic codec. Further investigation is conducted during development and the results are summarized in Chapter 6. It would be the most confident path to utilize all modes in full RD optimization, and gradually reduce its number as long as there is no significant negative effect on performance. However, this would place a significant computational burden on the experiments, since full RD optimization is very demanding in terms of computing resources. Thus, at first, we can also start with a limited number of modes for full RD optimization, and than gradually increase and decrease this number until some meaningful effect on performance is observed (see Chapter 6). In that way, after conducting experiments on optimal N candidates, we were able to pick the suitable number of modes for full RD optimization that does not influence the overall performance on seismic data.

In addition, the intra coding complexity still remains high and further computational savings are possible. It is assumed that 33 angular modes are more than enough since seismic data do not possess as many textural structures as natural images do. This motivates this work to focus on reducing the number of

angular modes for optimized encoding without significant performance losses. In order to experiment with a reduced number of intra modes, one simple option is to adopt the 9 intra prediction modes of H.264/AVC, which is essentially a uniformly decimated version of the 35 modes in HEVC. Hence, by using a uniformly decimated version of the 35 modes in HEVC, e.g., by taking every second or every third angular mode, the solution to unburden the encoder of unnecessary mode calculations can be reached. Another option is to utilize its the most frequent subset, as it is analyzed in Chapter 6. The assumption is that a better way is to choose a subset of modes that is better adapted to the structure of seismic data. After collecting statistics of mode selection of a large set of trace and 2D wavefield images, we can choose to use a (sub)optimal subset of modes for seismic data compression. This is analyzed in detail in Chapter 6 of this work.

However, similar to previous notes on complexity reduction, advanced complexity reduction was not examined in this work. An efficient and adaptive intra mode selection is planed to be the part of future research directions. An attempt can be made to reduce the number of modes based on previous statistics or structural information in the data such as edges, and to create a subset of intra modes for particular PB, e.g., as it is done in [151, 152, 153].

4.2.2 Inter Prediction [17]

Successive frames, or equivalently seismic slices, usually share structurally and statistically highly correlated regions. In that sense, some parts of the image may be repeated in time with little or no changes. This is particularly true for seismic data, where consecutive slices share structurally and statistically highly correlated regions. Therefore, in order to improve compression performance over still-image (intra coding) approaches, in this work 3D seismic data is treated as a sequence of frames, and predictive coding such as HEVC based inter prediction is applied. This kind of temporal redundancy reduction is also referred as inter-slice prediction.

However, compared with natural video, one of the characteristics of seismic data is low-motion activity. This means that when two consecutive frames, or as we refer to it seismic slices, are compared, there is high overlapping of the image regions with its close neighborhood in the previous slice. Hence, some of the previous findings that are used during the development of the standard, but which are based on natural video, must be reconsidered in this work.

To eliminate temporal redundancy between slices (commonly referred as inter-frame correlation), the encoder incorporates a prediction mechanism that is based on motion estimation (ME). It is based on block matching, and it consists of finding a displaced block within a reference frame, where the displacement is defined by a motion vector (MV). At the same time, while searching for a displaced block, an appropriate distortion measure is minimized to find the most similar match. The displaced block is searched in a reference image identified by the reference image index, which is part of the encoder configuration setting. More about different reference image management configurations in the proposed seismic codec can be found in Chapter 5.

In addition, in inter mode each PB is associated with one (uni-directional prediction) or two (bi-directional prediction) motion vectors, depending on the frame type (P or B frames). In the latter case, average values or weighted prediction of predicted signals are used. Bi-directional prediction has a significant influence on compression performance when natural video has been used. However, the effectiveness of such an approach for low-motion seismic data has not yet analyzed, as well as whether it is worth the additional complexity (additional calculations are required to compress B frames compared to P frames). This is also experimentally covered in this work. Also, the use of multiple reference frames/slices has been analyzed.

The motion compensated prediction process begins with the advance motion vector prediction (AMVP) scheme [24, 154], where the encoder has to choose a motion vector predictor among multiple predictor candidates [155]. Based on the standardized competition scheme, the AMVP forms a list of candidates, see Fig. 4.4, which competes for the motion vector predictor [155]. Only the first two available spatial candidates are chosen among the adjacent blocks, A_0, A_1, B_0, B_1 and B_2, where the candidates are processed in the given order. In case of unavailability of spatial candidates, one temporally collocated neighbour can be selected, C_0 or C_1. Thus, spatial candidates, temporally collocated candidate, or in addition zero-MV candidate, can be added to the list of competitors. If the final number of candidates in the list is less than two, zero motion vector may be added to the competition list so that at least two candidates compete for the motion vector predictor. Only two competitors from the list are further passed to the RD optimization module, and based on the minimum cost, one MV candidate is selected as a predictor for a particular PB. Later, this predictor is used as a starting point in ME. Although temporal candidates may be useful to improve performance, they require a significant amount of memory to

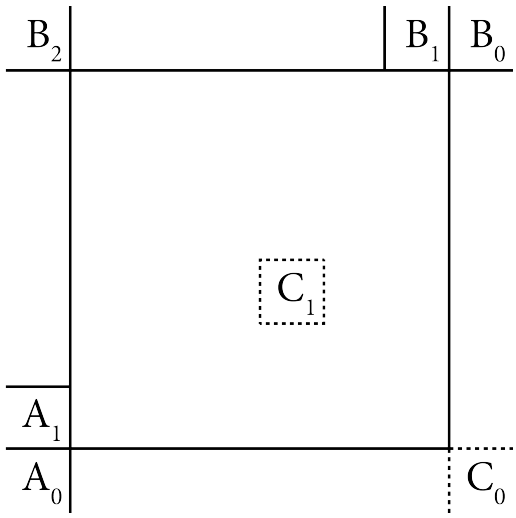


FIGURE 4.4: Motion vector predictor candidates.

store motion-related data for each PB from all reference frames. However, the encoder can control the use of the temporal predictor by using a flag within the configuration parameter set. Thus, the usefulness of the temporal predictor on the seismic data compression performance can be tested. If the results show negligible performance changes, it is better not to use temporal predictor, and by doing so we can slightly relax memory requirements.

Thereafter, the ME continues by applying integer-sample precision motion search. In the initial setup presented in this work, the full search is performed in a predefined search window range, meaning that each integer point displacement within the window has been checked in the RD sense. One point, which has proved to be the best in terms of coding price, is selected for further search. Fractional-sample ME is performed afterwards within the neighborhood of the selected integer point. Eight surrounding points are first evaluated using half-sample precision, followed by eight quarter-sample search to fine tune previously selected point. Quarter-sample precision additionally increases the complexity but at the same time it improves the performance [156]. Finally, efficient coding of the prediction information is utilized, where the difference between the MV predictor and the actual MV, and index of the AMVP candidate, are encoded into the bit-stream. It is important to note that fractional sample estimation uses seven-tap and eight-tap interpolation, which additionally increases

computational complexity [17]. In this work, we present the potential effectiveness of the half- and quarter-sample prediction when applied to seismic data. Although fractional sample precision can improve coding performance for natural video, interpolation introduce additional calculations. More importantly, in order to finely tune the MV, RD optimization must be performed for each half-sample and quarter-sample precision point. Thus, in this work, it is also analyzed how fractional MV estimation influence the performance of the proposed codec, in terms of coding performance as well as in terms of computational complexity.

Also, taking into account the low-motion characteristics of seismic data, this work attempts to limit the search window range while maintaining the same coding performance. The search range is narrowed to a neighborhood of 64 pixels, thus checking 64 integer displacements in each direction. In addition, results are provided for a reduced search range that goes down to 1-pixel neighborhood. Also, instead of performing RD optimization on each integer point within the window, test zone search (TZSearch) can be used [157]. It combines diamond search, raster search, and star refinement methods to speed-up encoder with negligible performance losses by omitting some integer-precision points in RD optimization. To additionally experiment with the number of integer-precision searches, raster search and star refinement within TZSearch are disabled. Since TZSearch can lead to suboptimal MV, its impact on the proposed custom codec performance is explored. The results of the examination and analysis of the experiments are given somewhat later in this thesis. It should be noted that in this work TZSearch is not used in the initial setup. Later-on, it was tested to see how it affects potential computation speed-up and compression performance. Also, a more detailed flow-chart of TZSearch algorithm can be found in [157].

A similar MV prediction scheme as AMVP is also used in MERGE and SKIP mode (a special case of MERGE mode when no residual data is present in the bit-stream). See [158] for more details on MERGE and SKIP mode in HEVC. Based on the set of the possible candidates, the encoder may select inter-coded PBs that share the same prediction information, and therefore does not have to explicitly transmit any motion related information. The merge index is used at the decoder side, along with the same competition scheme to efficiently reconstruct MV. The same candidates are used as in AMVP, except that a maximum of five candidates are selected for full RD optimization. The number of competitors in MERGE mode is controlled by a configurable parameter, and it is varied between 1 and 5 to show its influence on the codec's performance. It

is important to note that by using MERGE mode, the encoder is able to form a merged region that shares the same prediction information. In that way, by using block merging approach, the encoder's limitation that structurally the same regions of the image cannot be represented jointly if the region is divided into two (or more) CBs, is relaxed. In that sense, we can theoretically form a block of any shape, and that block is consistent in terms of image content.

As for the constraints for partition types, in inter prediction, the CB may consist of 1 PB of the same size, or 2 rectangular smaller PBs, or 4 square PBs, see Tab. 4.1. The division into 4 PBs is allowed only when the size of the CB is equal to the minimum allowable, the same as for intra prediction. When there are 2 PBs, an asymmetric PB partitioning can be enabled, and there are a total of six different type of non-square partitions. Asymmetric partitioning is only allowed when the size of CB edge is 16 or larger. In addition, to test the impact of asymmetric block partitioning on coding performance, the codec's performance is investigated when SIZE_2NxnU, SIZE_2NxnD, SIZE_nLx2N, SIZE_nRx2N modes are disabled.

To summarize, in this work, by taking into account the low-motion characteristics of seismic data, certain research has been conducted to limit the search window range to the lowest possible while maintaining approximately the same coding performance. In addition, instead of performing RD optimization on each integer point within the search window, tests are performed to see how the TZSearch affects the performance of seismic data compression. Also, this work examines the potential effectiveness of the half- and quarter-sample ME precision when applied to seismic data. A tests are also conducted when only zero-motion prediction is adopted in inter prediction. We refer to it as 1-1 prediction. A different number of AMVP candidates and MERGE candidates are also being tested.

4.3 Core Transform

Transform coding of the residual signal has been used to further make the most out of the spatial dependencies and to further improve compression gain. The basic purpose of the mathematical transform is to take the input signal, in our case block of residual samples resulting from intra or inter prediction, and represent it (transform it) from one type of representation to another, in a way more suitable for compression. It is used to translate the signal from spatial domain,

into another numerical quantitative form that we refer to as frequency domain or spectral domain. It can be shown that any finite set of the data can be expressed as sum of a weighted basis functions, for example see [159]. Hence, by manipulating the data, decorrelating transform separates the information based on its principal components (frequencies), and concentrates the data in a few significant transformed coefficients. Thus, data can be represented in the frequency domain using less information than in the spatial domain, with no or little harm on the reconstructed data quality. However, how many coefficients can be neglected during quantization process (that usally follows transform step, see Sec. 4.4) is a matter of compromise between the quality of the reconstructed samples and the compression ratio. Less coefficients are kept, the higher compression ratio will be, but the distortion of the reconstructed signal will be also higher.

In HEVC, transform block partitioning, which is given with residual quadtree hierarchical block division (see Sec. 4.1), has been utilized in order to carry out transform process of the residual signal. To obtain transformed coefficients, for each residual block, two-dimensional (2D) transform with the same size as the TB has been used. Possible transform block sizes are ranging from 4×4 to 32×32 pixels, and only square transform has been adapted in HEVC. Although Discrete Sine Transform (DST) is supported by the standard, only the Discrete Cosine Transform (DCT) [25, 159, 160] has been examined in our research. In total there are eight types of DCT (DCT-I to DCT-VIII), where in HEVC variant DCT-II and its inverse variant DCT-III are implemented.

Computationally vise, 2D DCT transform can be implemented by a column-row decomposition, using two separate 1D transforms, each one in horizontal and vertical direction. At the first stage, one dimensional N-point transform has been applied on each column of the input TB block (of size N $\times$ N). Next, N-point transform has been computed for the each row of the intermediate result from the previous stage, generating final 2D transformed coefficients of size $N \times N$. Therefore, 2D transform is equivalent to a one-dimensional transform performed along a single dimension followed by a one-dimensional transform in the other dimension. The approach when using two separable 1D transforms is more computationally effective than implementation of a direct 2D transform. The resulting transformed coefficients (*coeff*) are later-on subjected to the quantization and entropy coding. Mathematically, the simplest form of transformed coefficients calculation is in a matrix multiplication form, and it can be

expressed as

$$COEFF = (X \cdot B) \cdot X^T, \tag{4.4}$$

where COEFF is resulting transformed block composed of the transformed coefficients (*coeff*), B is a residual block after prediction, and X is DCT transform matrix. Inverse transform is simply represented as

$$B = (X^T \cdot COEFF) \cdot X. \tag{4.5}$$

Additionally, the DCT used in HEVC is an integer approximation of the orthonormal (original floating-point) DCT. Transform matrix coefficients are derived by approximating scaled DCT basis functions. The scaling factor used in HEVC is $2^{6+\frac{M}{2}}$ compared to an orthonormal DCT, where $M = log_2(N)$, and N is TB size. Also, to preserve the norm compared with orthonormal DCT and to keep dynamic range within 15 bits without sign bit (as defined by standard), additional scaling has been defined. A total scaling at the forward pass of the standardized DCT is $2^{(15-B-M)}$, and total scaling at the inverse transform is $2^{-(15-B-M)}$. This scaling is forwarded to the HEVC's quantizer (see the following section), to simplify and reduce the calculations by jointly observing the transform scaled factors with the quantization scaling. A given maximum transform range limitation has been considered as a reasonable trade-off between accuracy and implementation cost for standard video. For more details about standardized DCT see [25].

Contrary to the orthogonal basis vectors, standardized HEVC's integer transform introduces losses, even when observed without the intermediate scaling, and quantization and dequantization steps. For example, a scaled version of the original floating-point precision 4x4 DCT matrix is

$$X = \begin{bmatrix} 64.0000 & 64.0000 & 64.0000 & 64.0000 \\ 83.6200 & 34.6366 & -34.6366 & -83.6200 \\ 64.0000 & -64.0000 & -64.0000 & 64.0000 \\ 34.6366 & -83.6200 & 83.6200 & -34.6366 \end{bmatrix}$$

with $X \cdot X^T = X^T \cdot X = 16384 I_{4\times4}$, and the corresponding DCT in HEVC is

$$\tilde{X} = \begin{bmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & -64 & -64 & 64 \\ 36 & -83 & 83 & -36 \end{bmatrix}$$

with

$$\tilde{X} \cdot \tilde{X}^T = \begin{bmatrix} 16384 & 0 & 0 & 0 \\ 0 & 16370 & 0 & 0 \\ 0 & 0 & 16384 & 0 \\ 0 & 0 & 0 & 16370 \end{bmatrix}$$

and

$$\tilde{X}^T \cdot \tilde{X} = \begin{bmatrix} 16377 & 0 & 0 & 7 \\ 0 & 16377 & 7 & 0 \\ 0 & 7 & 16377 & 0 \\ 7 & 0 & 0 & 16377 \end{bmatrix}.$$

Thus, standardized transform given with $\tilde{X}$ is not unitary, and it does not support perfect reconstruction either. The same applies for the transform matrices of size 8×8, 16×16, and 32×32. Since the HEVC mostly focus on the applications that support high compression ratios, the loss introduced by the non-unitary transform does not degrade overall compression-quality performance. However, this property shall play an important role in the design of our custom codec. Although $\tilde{X}$ can be used for standard video data, the approximate error will be magnified several orders of magnitude for 32 b/p data, rendering it useless even without quantization. Also, HEVC adapts a better approximation of the DCT for 16 b/p data, with a scaling factor of $2^{14+\frac{M}{2}}$ for forward transform (inverse transform kept the same scaling as before), and the maximum transform range that is limited to 22 bits (without sign bit). However, the precision is still not good enough for 32 b/p data. Even it uses more precisely designed coefficients for higher bit depth extensions, with more bits required to represent transform matrix coefficients, based on experimental results with seismic data we can show that even extended precision introduce huge and unacceptable error. Thus, the core transform has to be carefully designed for 32 bit-depth data.

Furthermore, all DCT transform matrix coefficients can be stored in a single matrix of the length 32 (32×32 matrix). Any other lower-order transform

matrix may be derived from 32-point matrix by using its sub-sampled version. It is illustrated in Table 4.2. For example, 8-point matrix (matrix of the size 8×8) may be derived by using first 8 matrix coefficients and every 4^{th} column of the 32-point matrix, as it can be seen in Table 4.2. At last, preserving the symmetry and anti-symmetry of the original floating-point DCT transform matrix coefficients, fast algorithms may be applied to reduce the number of arithmetic operations. Although it does not support full factorization, DCT defined in HEVC may be computed by partial butterfly approach – combination of matrix multiplication and butterfly structures by using a $N/2$-point DCT and a matrix-vector product of $N/2 \times N/2$ matrix with an $N/2$-point vector, see [25, 160] for more details. An integer approximation of the HEVC's DCT transform matrix has been given in Tab. 4.2. It represent 32-point DCT, the highest possible transform size defined in HEVC. A smaller transform sizes are embedded within the 32-point transform matrix, and they are painted using different colors. Pink shaded parts form a 16-point DCT, yellow shaded coefficients are part of 8-point DCT, and blue colored coefficients represent 4-point DCT. Note, in total there are 29 different transform coefficients, and they can be represented using 8 bits [161]. This fact is an important factor in hardware implementation, when those coefficients are stored in a memory with a limited capacity, e.g., cache memory. Those coefficients are contained in the first row of the 32-point matrix. In addition, 16-point matrix use only 15 unique DCT coefficients, 8-point use 7, and 4-point DCT contains only 3 unique matrix coefficients. Also, we can see that DCT uses real computations since all DCT coefficients are real numbers (integers), and it is signal independent, which makes a DCT as good candidate for implementation in a compression algorithms. Performance in terms of the energy compaction is close to what can be accomplished by using more efficient decorrelators, e.g. transforms that are based on the underlying signal statistics such as KLT [159]. In general, standardized DCT preserves closeness to the original floating-point DCT. Almost equal norm of all basis vectors has been achieved, hence no additional scaling matrices are required, and the same quantization and dequantization scheme can be used for all transform sizes.

As we can conclude from the above, DCT in HEVC is very carefully designed taking into account to achieve good compression results, but also the complexity of the implementation of the transform for practical purposes was

TABLE 4.2: Integer approximation of the HEVC's 32-point DCT. Smaller sizes of 16-point, 8-point, and 4-point DCT are embedded in the 32-point transform matrix. For better visibility, they are painted with different colors.

64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64
90	90	88	85	82	78	73	67	61	54	46	38	31	22	13	4	−4	−13	−22	−31	−38	−46	−54	−61	−67	−73	−78	−82	−85	88	−90	−90
90	87	80	70	57	43	25	9	−9	−25	−43	−57	−70	−80	−87	−90	−90	−87	−80	−70	−57	−43	−25	−9	9	25	43	57	70	80	87	90
90	82	67	46	22	−4	−31	−54	−73	−85	−90	−88	−78	−61	−38	−13	13	38	61	78	88	90	85	73	54	31	4	−22	−46	−67	−82	−90
89	75	50	18	−18	−50	−75	−89	−89	−75	−50	−18	18	50	75	89	89	75	50	18	−18	−50	−75	−89	−89	−75	−50	−18	18	50	75	89
88	67	31	−13	−54	−82	−90	−78	−46	−4	38	73	90	85	61	22	−22	−61	−85	−90	−73	−38	4	46	78	90	82	54	13	−31	−67	−88
87	57	9	−43	−80	−90	−70	−25	25	70	90	80	43	−9	−57	−87	−87	−57	−9	43	80	90	70	25	−25	−70	−90	−80	−43	9	57	87
85	46	−13	−67	−90	−73	−22	38	82	88	54	−4	−61	−90	−78	−31	31	78	90	61	4	−54	−88	−82	−38	22	73	90	67	13	−46	−85
83	36	−36	−83	−83	−36	36	83	83	36	−36	−83	−83	−36	36	83	83	36	−36	−83	−83	−36	36	83	83	36	−36	−83	−83	−36	36	83
82	22	−54	−90	−61	13	78	85	31	−46	−90	−67	4	73	88	38	−38	−88	−73	−4	67	90	46	−31	−85	−78	−13	61	90	54	−22	−82
80	9	−70	−87	−25	57	90	43	−43	−90	−57	25	87	70	−9	−80	−80	−9	70	87	25	−57	−90	−43	43	90	57	−25	−87	−70	9	80
78	−4	−82	−73	13	85	67	−22	−88	−61	31	90	54	−38	−90	−46	46	90	38	−54	−90	−31	61	88	22	−67	−85	−13	73	82	4	−78
75	18	−89	−50	50	89	18	−75	−75	18	89	50	−50	−89	−18	75	75	−18	−89	−50	50	89	18	−75	−75	18	89	50	−50	−89	−18	75
73	−31	−90	−22	78	67	−38	−90	−13	82	61	−46	−88	−4	85	54	−54	−85	4	88	46	−61	−82	13	90	38	−67	−78	22	90	31	−73
70	−43	−87	9	90	25	−80	−57	57	80	−25	−90	−9	87	43	−70	−70	43	87	−9	−90	−25	80	57	−57	−80	25	90	9	−87	−43	70
67	−54	−78	38	85	−22	−90	4	90	13	−88	−31	82	46	−73	−61	61	73	−46	−82	31	88	−13	−90	−4	90	22	−85	−38	78	54	−67
64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64	64
61	−73	−46	82	31	−88	−13	90	−4	−90	22	85	−38	−78	54	67	−67	−54	78	38	−85	−22	90	4	−90	13	88	−31	−82	46	73	−61
57	−80	−25	90	−9	−87	43	70	−70	−43	87	9	−90	25	80	−57	−57	80	25	−90	9	87	−43	−70	70	43	−87	−9	90	−25	−80	57
54	−85	−4	88	−46	−61	82	13	−90	38	67	−78	−22	90	−31	−73	73	31	−90	22	78	−67	−38	90	−13	−82	61	46	−88	4	85	−54
50	−89	18	75	−75	−18	89	−50	−50	89	−18	−75	75	18	−89	50	50	−89	18	75	−75	−18	89	−50	−50	89	−18	−75	75	18	−89	50
46	−90	38	54	−90	31	61	−88	22	67	−85	13	73	−82	4	78	−78	−4	82	−73	−13	85	−67	−22	88	−61	−31	90	−54	−38	90	−46
43	−90	57	25	−87	70	9	−80	80	−9	−70	87	−25	−57	90	−43	−43	90	−57	−25	87	−70	−9	80	−80	9	70	−87	25	57	−90	43
38	−88	73	−4	−67	90	−46	−31	85	−78	13	61	−90	54	22	−82	82	−22	−54	90	−61	−13	78	−85	31	46	−90	67	4	−73	88	−38
36	−83	83	−36	−36	83	−83	36	36	−83	83	−36	−36	83	−83	36	36	−83	83	−36	−36	83	−83	36	36	−83	83	−36	−36	83	−83	36
31	−78	90	−61	4	54	−88	82	−38	−22	73	−90	67	−13	−46	85	−85	46	13	−67	90	−73	22	38	−82	88	−54	−4	61	−90	78	−31
25	−70	90	−80	43	9	−57	87	−87	57	−9	−43	80	−90	70	−25	−25	70	−90	80	−43	−9	57	−87	87	−57	9	43	−80	90	−70	25
22	−61	85	−90	73	−38	−4	46	−78	90	−82	54	−13	−31	67	−88	88	67	31	−13	−54	82	−90	78	−46	4	38	−73	90	−85	61	−22
18	−50	75	−89	89	−75	50	−18	−18	50	−75	89	−89	75	−50	18	18	−50	75	−89	89	−75	50	−18	−18	50	−75	89	−89	75	−50	18
13	−38	61	−78	88	−90	85	−73	54	−31	4	22	−46	67	−82	90	−90	82	−67	46	−22	−4	31	−54	73	−85	90	−88	78	−61	38	−13
9	−25	43	−57	70	−80	87	−90	90	−87	80	−70	57	−43	25	−9	−9	25	−43	57	−70	80	−87	90	−90	87	−80	70	−57	43	−25	9
4	−13	22	−31	38	−46	54	−61	67	−73	78	−82	85	−88	90	−90	90	−90	88	−85	82	−78	73	−67	61	−54	46	−38	31	−22	13	−4

taken into account. However, it is designed for a completely different application and a different purpose from the one that this work is dealing with. Therefore, it cannot be directly applied in this work. Starting from exploring the possibilities of integer-based approximated versions of the original floating-point DCT, a completely different solution for the core transform has emerged in this work.

Nevertheless, since many different implementations of the DCT have been shown to be practically effective in terms of the implementation cost and computational performance, we choose to keep using the DCT and re-implement it in order to empower extended bit-depth. In addition, work in [1] showed that adaptive local cosine transform with different sizes performs very well on seismic data.

4.3.1 Original Floating-Point DCT

The original floating-point DCT has been considered as a straightforward solution to the previous problem when using integer approximation of DCT. The definition of a one-dimensional original floating-point DCT for a discrete signal $f(n), n = 0, 1, ..., N-1$ is given by

$$F(i) = c(i)\sqrt{\frac{2}{N}}\sum_{n=0}^{N-1} f(n) * cos\left[\frac{(2n+1)i\pi}{2N}\right]$$
$$i = 0, 1, ..., N-1, \tag{4.6}$$

and the inverse transform is defined as

$$f(n) = \sqrt{\frac{2}{N}}\sum_{i=0}^{N-1} c(i)F(i) * cos\left[\frac{(2n+1)i\pi}{2N}\right]$$
$$n = 0, 1, ..., N-1, \tag{4.7}$$

where $c(i) = 1/\sqrt{2}$ for $i = 0$, $c(i) = 1$ for $i = 1, 2, ..., N-1$. The vector F represents transformed coefficients, while f could be any discrete signal but in our case it is represented as the residual signal. F has the same dimension as input vector f, but in the new domain – frequency domain. As we said before, in practice 2D DCT is implemented using column-row decomposition, where each time 1D DCT has been performed. For example, $F(i)$ on the given location i is a result of the 1D forward transform within one line (column or row) of the residual block (data block after prediction).

It is important to note that the cosine terms given with (4.6) and (4.7), together with scaling factors $c(i)$ and $\sqrt{2/N}$, can be pre-computed and stored as a DCT matrix for further use (since both the forward and inverse transforms use the same scale factor, which makes the DCT orthogonal). Indeed, starting from the expression above, we can derive the original floating-point transform matrix x (see below). It can be shown that x can lead to the HEVC's standardized DCT matrix $\tilde{X}$.

The transform matrix x is given with

$$x(i,j) = c(i) * \sqrt{\frac{2}{N}} * cos\left[\frac{(2j+1)i\pi}{2N}\right]$$
$$i = 0,1,...,N-1; j = 0,1,...,N-1, \tag{4.8}$$

where $c(i) = 1/\sqrt{2}$ for $i = 0$, $c(i) = 1$ for $i = 1,2,...,N-1$. For example, the original floating-point DCT transform matrix of size $N = 4$ is

$$x = \begin{bmatrix} 0.5000 & 0.5000 & 0.5000 & 0.5000 \\ 0.6532 & 0.2705 & -0.2705 & -0.6532 \\ 0.5000 & 0.5000 & 0.5000 & 0.5000 \\ 0.2705 & -0.6532 & 0.6532 & -0.2705 \end{bmatrix}.$$

In the same manner, using (4.8), we can calculate original floating-point DCT matrix for any other transform size N. In that way, matrix X, which was shown earlier in the text, is the scaled version of the x, and $\tilde{X}$ is its integer approximation. It is important to note that in our codec the transform matrix coefficients are stored in a double format, where a coefficients are rounded to 8 decimal places after decimal point. Also note that the forward and inverse DCT matrix are transposes of one another.

However, floating-point multiplications are computationally very intense in both software and hardware, and hence not suitable for practical usage. This particular approach is extremely inefficient due to a double nested for-loop, and floating-point multiplication. For example, one pass (1D transform) requires N^2 floating point multiplications and $N(N-1)$ additions. The number of operation doubles for 2D transform. For that reason, even though the best energy compaction can be achieved when compared to other more practical solutions, floating-point DCT has been used mainly for comparison purpose in order to show the performance in terms of achievable coding gain and computational speed, e.g., to estimate binDCT's performance.

4.3.2 BinDCT

By utilizing the symmetry (and anti-symmetry) properties of the basis vectors, the number of arithmetic operations can be significantly reduced. For example, it could be achieved by using Chen's factorization of the DCT matrices [162],

which consists of alternating cosine/sine butterfly matrices (also known as plane rotations) with a binary butterfly matrices (also known as ± 1 butterfly). An illustration of cosine/sine butterfly and binary butterfly is given on Fig. 4.5. The approach to decompose a DCT matrix in a way as given in [162] is known as even-odd decomposition, where even part is a transform of lower order, e.g. N/2 $\times$ N/2, and odd part is factorized into the series of matrices of different distinct types. A total 4 distinct types exist in Chen's approach and $2 * log_2 N - 3$ matrices of those different types are used to decompose odd part of a N-point transform. Thus, by using even-odd decomposition of the original floating-point DCT matrices, a recursive form of implementation has been utilized, where lower order transform matrices are part of a larger transform. It can be shown that extension to the next higher order transform only requires adding one set of binary butterfly to accommodate a new input samples, and new set of alternating cosine/sine butterfly matrices with a binary butterfly matrices at the odd part of the scheme. In general, Chen's algorithm may be used for any value of $N = 2^n$, where $n \geq 2$, in contrast to some other methods that are more efficient in terms of number or computational steps, but cannot be generalized to the arbitrary transform sizes. It turns out to be very important property due to the HEVC's variable block size. That is the main reason why we have decided

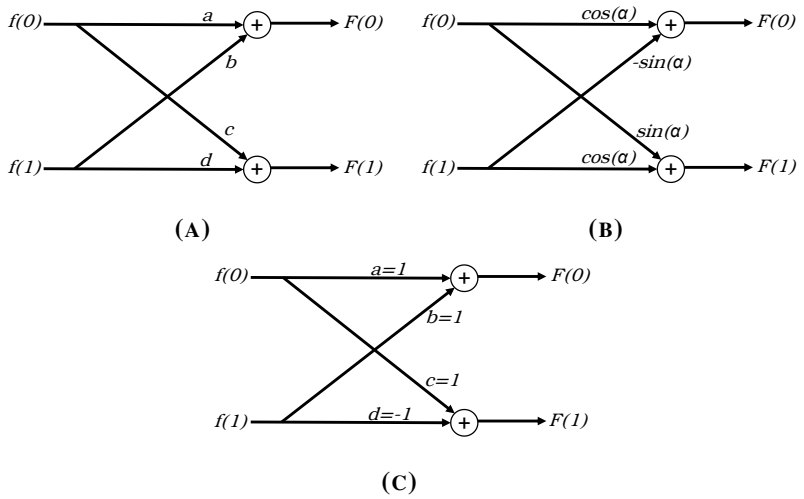


FIGURE 4.5: An example of butterfly structure. (A) General butterfly, (B) cosine/sine butterfly, and (C) binary butterfly

to use the Chen's approach for a fast DCT implementation. For more detailed and systematic way of how to decompose a DCT matrix of an arbitrary size, readers are referred to the original paper [162]. The complete DCT flow graph after factorization for $N = 4, N = 8, N = 16$, and $N = 32$ is given in Fig. 2 in the original paper. Lastly, a given matrix decomposition approach requires $(3N/2)(log_2N - 1) + 2$ additions, and $Nlog_2N - 3N/2 + 4$ multiplications.

Regardless, even though the number of the computations may be significantly reduced when compared to a direct matrix multiplication approach, both aforementioned approaches still use floating-point multiplications, no matter if decomposition is done or not. In order to propose more practical solution compared to DCT with floating point multiplications, we have examined a lifting-based scheme called the binDCT [163]. We extend the idea of [163] and implement binDCT of sizes 4×4, 8×8, 16×16, and 32×32, with the lifting steps derived by using already mentioned Chen's factorization. Chen's relationships with the binDCT scheme comes from the point that each (non-binary) butterfly step after Chen's decomposition can be additionally decomposed into three lifting steps as

$$\begin{bmatrix} cos\alpha & -sin\alpha \\ sin\alpha & cos\alpha \end{bmatrix} = \begin{bmatrix} 1 & p \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ u & 1 \end{bmatrix} \begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix},$$

where $p = b = (cos\alpha - 1)/sin\alpha$, and $u = sin\alpha$. In addition, each butterfly step is a biorthogonal transform (to invert a forward transform, we have to subtract out what was added at the forward transform.), and its inverse can be represented with a similar lifting structure, i.e.,

$$\begin{bmatrix} cos\alpha & -sin\alpha \\ sin\alpha & cos\alpha \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -p \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -u & 1 \end{bmatrix} \begin{bmatrix} 1 & -b \\ 0 & 1 \end{bmatrix}.$$

However, lifting coefficients are still floating points. Further approximation of p, u, and b lifting coefficients is applied in order to permit fast multiplierless approximations. Thus, binDCT is implemented by using lifting steps with only binary shifts and additions, and floating-point lifting coefficients approximated by dyadic values in $k/2^m$ form. Even with coefficients approximation with its dyadic values, binDCT is lossless as long as forward and inverse transform use the same shifting procedure. It still stands that to invert a forward transform, we have to subtract out what was added at the forward transform. It also maps

integers to integers in every step of the transform with perfect reconstruction.

In order to enable different complexity vs. coding performance trade-off, we have examined dyadic approximations with different accuracy. Depending on how we choose $k/2^m$ for different lifting coefficients, we obtain different performances. Better approximation give us binDCT coefficients closer to the floating-point lifting coefficients, but requires more shifts and additions. However, due to large possible choices of approximations, it is important to determine dyadic lifting values prior to implementation in order to avoid encoder-decoder mismatch. After extensive experiments, in order to limit complexity, and at the same time to closely approximate floating-point values, we choose $2^m = 32$ as the highest allowed divisor in our setup (thus the highest shift is $m = 5$). Note that after Chen's factorization, in total, 38 unique floating-point lifting coefficients have to be calculated, distributed among 10 lifting stages (for the largest 32-point binDCT). However, taking into account previous constraint ($2^m = 32$) some of unique floating-point coefficients can be fitted with the same dyadic value, leading us to only 18 unique lifting coefficient approximations, which also relax memory requirements. It will be shown that selected design closely approximate original floating-point DCT, with significantly lower computational complexity (see Sec. 6). Thus, binDCT is also able to closely approximate the original DCT while providing perfect (lossless) reconstruction (without quantization). For more details on a selected dyadic values, see the proposed binDCT implementation at [164]. The given implementation emerged as a part of this work. It provides c++ code that is compliant to HEVC's reference code. It also provides code generator written in Python. It means that for different coefficient approximations, which are given as an input parameter in the textual file, it generated c++ code using those coefficients. In that way we were able to easily test different configurations (different coefficient approximations), and to pick the one that is the best in terms of compromise between coding gain and computational complexity. It will be shown that configuration C2 given in [164] is the most appropriate choice given the required computational power and given the resulting compression performance.

In this way we have implemented the fast lossless version of DCT transform that can be used in a seismic image compression, which turns out to provide good energy compaction, and which is optimal in terms of simplicity. It can be shown that performance in terms of coding gain is very close to what can be achieved by original floating-point DCT, however with significantly less computational effort, which also can be controlled by using a different configuration,

and memory requirements may be limited by choosing the appropriate number of unique lifting coefficients.

4.4 Quantization

The coefficients obtained after the transform are subjected to a quantization process in order to further reduce the dynamic range and to get quantized *level*. The quantizer is used to discard transformed coefficients with relatively small amplitudes without introducing visual distortions into the reconstructed data. In some cases, when higher compression ratios are needed, the quantizer can discard a significant number of transformed coefficients, but at the cost of deteriorating the quality of the reconstructed data. The more coefficients we truncate within the quantization process, we can achieve higher compression ratios, but the quality of the reconstructed data will be lower. A scalar quantizer has been used in HEVC, where each input symbol is treated separately in producing the output. Each quantization interval has the same length, so uniform quantization is used. The advantage of using such a quantizer is its low computational complexity.

HEVC initially supports the quantization parameter Q_p in the range from 0 to 51. For high bit-depth data, the given Q_p range is too narrow. Thus, an additional offset is introduced to allow bit-depths greater than 8 b/p, leading to

$$Q' = Q_p + 6(B - 8), \tag{4.9}$$

where B is the bit-depth, Q_p is the input parameter, and Q' is the new internal quantization parameter used by the encoder/decoder. Taking into account the offset, the Q_p value can be a negative number, allowing a lower compression ratio for extended bit-depths. For example, it can be useful in many applications, such as remote sensing [117, 118]. However, it is important that the internal Q' is not less than 0. This leads to a new range of input Q_p values, which in the case of 16 b/p input data (maximum bit-depth supported by the standard) goes from -48, while the value of 51 remains the upper limit for input Q_p. Using (4.9) we can easily find upper limit for internal Q' which is defined for 16 b/p, and it is equal to 99 (when $Q_p = 51$). To summarize, for $B = 16$ the initial Q_p range can go from -48 to 51, and the corresponding internal Q' from 0 to 99. For $B = 8$ we can see that $Q' = Q_p$ and ranges from 0 to 51, as defined in the standard. In the proposed seismic encoder, it could be straightforward to extend the range

of the quantization parameter by simply setting $B = 32$ in (4.9). In such a case, Q_p is in the range from -144 to 51, and consequently the internal quantization parameter Q' goes from 0 to 195. However, there are other problems with the use of the HEVC's quantizer in the proposed codec for seismic data, which will be analyzed after clarifying the standardized implementation in a little more detail.

In HEVC, Q' is not directly used for quantization, rather it is mapped to Q_{step} [25, 160]. Q_{step} is the actual uniform quantization step size, which is directly utilized in the quantization (and dequantization), and it is given with

$$Q_{step}(Q') = 2^{(\frac{Q'-4}{6})}. \tag{4.10}$$

The quantizer given in (4.10) can be also expressed as

$$Q_{step}(Q') = G_{Q'\%6} << round(\frac{Q'}{6}), \tag{4.11}$$

and thereafter its integer approximation is used by shifting and rounding the values in $G = [2^{-4/6}\ 2^{-3/6}\ 2^{-2/6}\ 2^{-1/6}\ 2^{0/6}\ 2^{1/6}]$ to certain level to avoid direct division/multiplication at the quantizer/dequantizer. The operator '$\ll$' represents bitshift, and '$\%$' is modulo operator. Thus, it can be implemented with integer multipliers and shift operations only, where the multipliers depend only on the value of QP (or equivalently Q') and the shifts depend only on the transform size. More details on integer implementation can be found in [25], and some of its the most important aspects will be given bellow. The dequantization process and scaling factors are defined in HEVC (since standard defines decoder side), although recommendation for quantization has been integrated within the HEVC reference software.

Therefore, the quantizer is indirectly defined in HM and its integer approximation is given with

$$\begin{aligned} f_{Q'\%6} &\approx \frac{2^{14}}{G_{Q'\%6}} \\ &= [26214, 23302, 20560, 18396, 16384, 14564] \end{aligned} \tag{4.12}$$

$$level = ((coeff * f_{Q'\%6} + offset1) \gg \frac{Q'}{6}) \gg shift1, \tag{4.13}$$

where *coeff* is the input to the quantizer (transformed coefficient), f is a scaled version of G (see (4.12)), the offset has been implied to secure rounding ($offset1 = 1 \ll (shift1 + \frac{Q_p}{6} - 1)$), and *shift1* has been introduced to preserve the norm of the original DCT. The output of the quantizer (*level*) is the quantized value of the transformed coefficient. *shift1* has been obtained in such a way that the up-scaling introduced by forward transform and quantization jointly (in order to have an integer representation) has to be annulled and the norm of the original DCT has to be preserved. It leads to $2^{-shift1} * f_4 * 2^{(15-B-M)} = 1$, where f_4 represents the quantizer when $Q_p = 4$ (or equivalently $Q_{step} = 1$, hence no division). The total scaling from HEVC's forward transform (total scaling of the coefficients before quantization) is equal to $(15 - B - M)$, where B is the bit-depth and M is $log2$ of the transform size N. Taking all into account we can get $shift1 = (29 - M - B)$.

The dequantizer is defined in the standard. For the quantized level, dequantization is defined with

$$g_{Q'\%6} = round(2^6 * G_{Q'\%6}) = [40, 45, 51, 57, 64, 72] \tag{4.14}$$

$$coeff = ((level * (g_{Q'\%6} \ll \frac{Q'}{6})) + offset2) \gg shift2. \tag{4.15}$$

Similarly as in the quantizer, *offset2* is used for rounding ($offset2 = shift2 - 1$), g is a scaled version of G (see (4.14)), and *shift2* has been calculated in a similar way as *shift1* where the total scaling when multiplied together should result in product of 1. In order to preserve the norm, and cancel the scaling from inverse integer transform and dequantization, *shift2* should be equal to $M - 9 + B$ (given from $2^{-shift2} * g_4 * 2^{-(15-B-M)} = 1$). It is important to point out that the factor $\frac{Q'}{6}$ in both (4.13) and (4.15) represents an integer division. At the end, also note that the size of the quantization/dequantization matrices is equal to 6, one entry for each value of $G_{Q'\%6}$. This way of definition of the quantizer/dequantizer is efficient in terms of memory requirements as well.

It is obvious that the quantization process is well suited for HEVC when jointly observed with an integer approximation of standardized DCT. However, when incorporated in the proposed codec with adjusted DCT (see Sec. 4.3), the original HEVC's integer approach cannot be anymore exploited, and it should be changed accordingly. Thus, in this work the following has been examined. Instead of using an integer approximation of HEVC's quantizer/dequantizer, one

solution could be to directly utilize Q_{step} given with (4.10) (or equivalently with (4.11)). In this case, quantization is nothing more than division of the transform coefficient (*coeff*) with Q_{step} on the encoder side, and dequantization is equivalent to multiplication of the decoded *level* by Q_{step} at the decoder side, followed by shifts to preserve the norm of transformed coefficients. Therefore, starting from (4.13), direct quantization after omitting the HEVC's integer approximation after redeployment can be written as

$$\begin{aligned} level &= \frac{coeff * \frac{\frac{2^{14}}{G_{Q'\%6}}}{2^{\frac{Q'}{6}}} + \frac{2^{shift1+\frac{Q'}{6}-1}}{2^{\frac{Q'}{6}}}}{2^{shift1}} \\ &= coeff * \frac{\frac{2^{14}}{G_{Q'\%6}}}{2^{\frac{Q'}{6}} * 2^{shift1}} + \frac{2^{shift1+\frac{Q'}{6}-1}}{2^{\frac{Q'}{6}} * 2^{shift1}} \\ &= coeff * \frac{1}{G_{Q'\%6} * 2^{\frac{Q'}{6}}} * \frac{2^{14}}{2^{shift1}} + 0.5 \\ &= \frac{coeff}{Q_{step}(Q')} * 2^{14-shift1} + 0.5 \\ &= \frac{coeff}{Q_{step}(Q')} * 2^{14-(29-M-B)} + 0.5 \\ &= \frac{coeff}{Q_{step}(Q')} * 2^{-(15-M-B)} + 0.5. \end{aligned} \tag{4.16}$$

Hence, quantization is simply a division by the quantization step size that is a function of Q'. Downscaling by $2^{-(15-M-B)}$ is exactly the same as the total upscale factor that has been introduced in the forward transform (necessary to preserve the norm of the original floating-point DCT and at the same time to have integer representation). Clearly, transform and quantization in HEVC should be observed jointly due to the scaling factors that are transferred between the two modules. Offset has been implied to secure rounding. Similarly, after redeployment of (4.15), for dequantization we can get

$$coeff = level * Q_{step}(Q') * 2^{(15-M-B)} + 0.5, \tag{4.17}$$

which is equivalent to multiplication with Q_{step}, followed by scaling that is

transferred from the inverse transform and adding an offset to secure rounding. It is important to note that the scaling factor of the standardized HEVC's integer transform has been incorporated in the quantizer/dequantizer, in order to preserve the norm of the transformed coefficients and to correctly reconstruct the quantized levels. The reason for the joint observation of transformation and quantization (or inverse transformation and dequantization) is justified in the simplified implementation if the scaling is shifted to quantization/dequantization, in which case it can be jointly observed with quantizer/dequantizer shifting factors (when integer approximation of quantizer/dequantizer is used).

Obviously, to extend the Q_p range of the HEVC's quantizer is straightforward, after which the quantizer in (4.10) is wide enough to facilitate the extended bit-depth range, e.g., maximum $Q_{step} \sim 2^{32}$. It is clear that in HEVC transform and quantization should be observed jointly due to the scaling factors that are transferred between the two modules. Since the HEVC's integer transform is replaced, we cannot utilize the quantization given with Eq. 4.13 and Eq. 4.15 directly. Thus, one solution is to use the wrapped out formulas given in (4.16) and in (4.17), however without any scaling factors, and in such a way to use direct division/multiplication with the quantization step size given with (4.10) (or equivalently with (4.11)). While this is the most logical solution, keep in mind that the quantizer in that way uses a floating-point multiplication/division, which can be expensive and time consuming in both hardware and software.

In order to propose the more efficient solution, the HEVC's quantizer is completely omitted. Thus, we can make the most out of the uniform scalar quantizer proposed in [2]. The following quantizer mapping is adapted, in which quantization parameter Q_p is mapped to the scaling factors according to

$$Q_{step} = \begin{cases} 1 & Q_p = 0, \\ 2 * Q_p & Q_p < 16, \\ man * 2^{exp} & Q_P \geq 16, \end{cases} \tag{4.18}$$

where $man = 16 + (Q_p \% 16)$ and $exp = round(Q_p/16)$. In our codec, Q_p ranges from 0 to 400 ($MAX\_QP = 400$) with $Q_p = 0$ leading to lossless compression (assuming DCT transforms with perfect reconstruction). Also, limiting the maximum Q_p to 400 is enough to facilitate the high-dynamic range, since the maximum quantization step size is approximately 2^{29}, which is effectively

more than enough for any practical purpose. Note that the new quantizer has a *harmonic* scale because Q_{step} increases linearly with respect to Q_p initially, but later grows exponentially to allow high compression ratios for high bit-depth data. This leads to fast implementation by bit-shifts (together with a few multiplications due to limited number of "mantissas"). Additionally, when compared with the quantizer mapping in (4.10) that uses 195 different values to approach the maximum quantization step ($\sim 2^{32}$), the proposed quantizer mapping uses a much finer scale to approach even the lower upper limit for the quantization step ($\sim 2^{29}$). Thus, it leads to the quantizer step spacing that is sufficient enough to offer a wide range of different qualities and rates. Also note, that the uniform quantization is a well-established and proven approach to coding seismic data [1].

4.5 Entropy Coding

To further reduce statistical dependencies and efficiently compress data, HEVC utilizes Context Adaptive Binary Arithmetic Coder (CABAC) as the main entropy coding mechanism [26, 165, 166]. It has been proposed in H264/AVC, and with some modifications, mainly to improve throughput, it has been used in HEVC. The CABAC scheme is itself a lossless compression technique, however only if it is isolated from other loss-making modules of HEVC, e.g., transform and quantization. It consists of three stages: syntax element binarization, context modeling or bypass mode, and binary arithmetic coding.

Binarization is used to rewrite integer syntax elements, e.g., quantized levels, into binary codewords, where each element of a binary code word (also called a bin string) is called a bin. Thus, HEVC uses a more efficient alphabet, in this case binary, by assigning codewords using different binarization schemes. The binarization part of CABAC has high impact on the final bit-rate, therefore several binarization approaches are adopted for different syntax elements. The binarization approaches used in HEVC are fixed length, unary, truncated unary, Golomb-Rice code, and k-th order Exponential-Golomb code. In addition, to code some syntax elements HEVC uses concatenation of those schemes. Note, when syntax element is already in a binary form, with a value of 0 or 1, the binarization step is omitted.

In order to form the final bit-stream and to approach the information theoretic minimum bit-rate, binary code is further compressed by the arithmetic

coder (AC) [167, 168]. More precisely, the moduo coder (M-coder), a fast table-driven approximation of AC, is utilized in HEVC [169]. Reduction to the binary alphabet allows to utilize the fast and multiplication free versions of (AC). It is easier to use the binary form of the AC, given that it uses a simpler form of interval subdivision since only two symbols can appear (0 or 1). Another important aspect of using binary codes is that it enables encoding on the sub-element level by utilizing the binary arithmetic coder and context modeling on a bin level, which in addition enables reduction of inter-symbol redundancy. In addition, AC can operate in two modes – context adaptive mode (regular mode) or bypass mode. In the regular mode, different probability models (context models) have been assigned. Probability estimation is adaptive, which means that underlying probability distribution is based on image statistics and probability distribution is not considered static. Hence, rather than to use fixed probabilities for bin coding, an adaptive probability estimation has been used based on the correlation between bins, which is essentially referred as adaptive context modeling. Different probability models have been used given already coded bins in the neighborhood of the currently observed bin. Context modeling ensures that the probability distributions over the bins are not the same at each step, and they are updated after each bin has been encoded, which models the statistics of the underlying data. After each bin has been encoded, the probability models are updated. Therefore, conditional probabilities for coding bins are estimated, and later-on used in AC. Thus, the binary AC engine compress the bins with the given context models (estimated probabilities) to the final bit-stream.

In HEVC, context models have been initialized with a given pre-defined probability distribution prior to coding. The initialization is Q_p dependant, and it is designed based on an extensive offline analysis of data statistics after quantization. Note that, the most commonly used data source in these analyzes was natural video. However, in this work, the context initialization is configured such that the same initialization has been used for any Q_p. The initialization of context models as HEVC's initialization when $Q_p = 0$ has been used in this work, since it most closely matches the data statistics that is expected to appear in a given use-case scenario (the larger absolute levels of quantized coefficients are most likely to appear in 32 b/p extension). A similar approach has been used in the original HEVC design for 16 b/p extension, where all negative Q_p values are clipped to 0 prior to context initialization. Clearly, initialization of the context models might be sub-optimal for seismic data statistics. Therefore, there is space for further research on a more efficient probability distribution for the

context model initialization for different Q_p values.

Also, the regular mode in CABAC is highly sequential, given the nature of context modeling and the dependencies within it. Contrary to the regular mode, the bypass mode is used in order to speed up encoding process and to improve throughput. No specific probability model is assigned for bypass-encoded bins, and it is assumed that the probability for both bin values (0 or 1) is equal. In such a case, arithmetic coding can be implemented only with shifts (due to simple range division), and bypassed bins may be grouped together in order to be processed in a single cycle since no data dependency exists.

Since the CABAC (including M-coder) has been designed and optimized with the aim to provide efficient coding, and at the same time to provide a low computational cost and increased throughput, and with a reduced data dependencies in order to enable better parallelization, in this study it was chosen to make the most out of the HEVC's coding mechanism. Compared to H264/AVC, many improvements have been proposed in CABAC. For example, compared to the 298 context models in H264/AVC, HEVC utilizes only 154 context models. The reduced number of contexts significantly improved throughput and reduced complexity, which in addition has been shown to have a high influence of the practical implementation of the codec. At the same time, with profound context design, coding performance has not been degraded. Thus, during the design of the entropy coding part of HEVC, a lot of effort was put in to reduce the total number of contexts, to reduce the number of context coded bins that are shifted to be used in bypass coding mode, to reduce the number of bins on average and in worst case scenario (by using the variety of binarization approaches that were mentioned earlier), and to reduce dependencies in context selection. Hence, it has been designed to achieve a good trade-off between coding efficiency and coding complexity. In addition, it is obvious that the simplifications proposed with the M-coder reduce the coding precision and negatively affect the performance of the probability estimation. However, the use of a standardized HEVC's M-coder significantly reduces the coding complexity given the extensive use of multiplication operations during the encoding process in the original form of the AC. Hence, the reduction to the binary alphabet and the use of M-coder is justified in this regard, especially considering the cost of multiplication operations and the number of their use during encoding and decoding. Since multiplications are totally omitted in the M-coder by using a pre-computed table for range division, a compromise made in terms of slightly poorer coding performance is more than welcome because otherwise practical implementation would be much

more complicated and use of AC in practical applications would not be feasible. Therefore, the use of a given entropy coder has a justification in the efficient implementation, and this work has kept this part of the HEVC as it is.

Despite being very effective, some parts of the CABAC still have to be adapted. Since the increased bit-depth range changes no other syntax elements than the *quantized levels*, this work choose to exploit the most out of the HEVC's coding engine, and to keep the most parts related with syntax modeling, binarization, and syntax element coding. At the same time, the encoding of quantized levels involves several key elements that are completely inherited from the original HEVC, such as multi-level significance map coding, last significant level coding, sign coding, etc. Those parts are briefly discussed in the following in order to have general impression on how HEVC implements coding of each block after transform and quantization are performed. However, more details can be found in [26, 170, 171] and in the references provided in the following text.

4.5.1 Golomb-Rice and Exponential-Golomb code

Before going into more detailed explanation of the binarization and coding scheme of absolute quantized levels, the codes that are used to represent absolute quantized levels in binary format are briefly explained. The codes used in HEVC for this purpose are Golomb-Rice (GR) and Exponential-Golomb (EGk) code [172,173,174,175,176]. In addition, binarization consists of the concatenation of the two binarization approaches. Readers familiar with these binarization approaches can skip reading this section. Also, fixed code, unary, and truncated unary codes are included for more detailed explanation. To illustrate a construction of those codes, an example is given in Table 4.3 for a maximum absolute level of $L_{max} = 15$.

- Fixed-length code (FL). Each symbol is mapped to the binary string (binary representation) of length $\lceil \log_2(L_{Max} + 1) \rceil$, where L_{Max} is the highest absolute level for a given dynamic range.

- Unary code (U). The symbol of a value L is mapped to a series of $L + 1$ binary values of which the first L elements are equal to 1, while the last element has a value of 0 and it is known as terminating bit.

- Truncated Unary code (TU). The TU code truncates out the least significant bit if L is equal to the maximum allowed value, $L = L_{Max}$. In HEVC, L_{Max} is predefined for each syntax element that uses TU codes. In the case when $L < L_{Max}$ the code is the same as unary code, else if $L = L_{Max}$ the code consists of series of L elements that are equal to 1, where terminating bit 0 is left out.

- Golomb-Rice (GR) codes represent a symbol as a two part, where the first part is quotient (prefix) and the second part is reminder (suffix). Hence, symbol L can be represented as a $L = q * m + p = \lfloor L/m \rfloor * m + p$, where q is quotient, p is reminder, $m = 2^r$ and r is Rice parameter. Also, in HEVC, Rice parameter r is not fixed. It uses adaptive process of Rice parameter initialization, and later-on it can be adapted based on the statistics of underlying data. In addition, q is coded as truncated unary code with $q + 1$ bits, and p is represented by the fixed length code of the length r. When $r = 0$, this scheme is equivalent to unary code.

- Eksponential-Golomb codes of k-th order (EGk) code is the derivation of the GR code. In EGk code, the code word size increases exponentially. For each bit in the prefix part, the number of the codes in the suffix part doubles. However, it still consists of two parts – prefix q and suffix p (similar to the quotient and reminder in GR codes). Every code word consists from an unary code to represent the prefix part, followed by fixed length code for suffix part. Prefix value to be coded is given with $p(L) = \lfloor \log_2(L * 2^{-k} + 1) \rfloor$, and it is coded using $l(p) = p(L) + 1$ bits with unary code. A suffix value to be coded is $s(L) = L + 2^k(1 - 2^{p(L)})$, and it is coded as fixed length representation of a length $l(s) = p(L) + k$, where k is the order of the Exponential-Golomb code. A fixed length part can be observed as a binary representation using the $l(s)$ least significant bits. Note that, in HEVC specifically, k is equivalent to the $r + 1$, where r is previously mentioned Rice parameter.

The following two schemes are specifically designed for HEVC, and they are a simple derivation of GR and EGk codes. It is based on the fact that, in image compression, finite length codes have been used due to the limited bit-depth range. Thus, in HEVC, the possible codeword lengths can be bounded by the dynamic range given by the bit depth of the input data.

- Truncated-Rice (TR) code is derivation of the GR code. In TR code prefix part can be truncated based on its length. Since the scheme limits the length of the prefix part of the GR codes, it can be truncated in order to remove the redundancy in the binary representation. In HEVC only, the maximum value that can be coded with the truncated Golomb-Rice code is given with $L_{Max} = 4 * 2^r$, where r is the Rice parameter. This correspond to the maximum length of the prefix bins that is set to 4 in HEVC. If the prefix value is less than $\lfloor L_{Max}/2^r \rfloor$, it is coded as before using unary codes. Otherwise, one bin can be truncated, with all bins being equal to 1. Suffix is only presented when the value L is smaller than L_{Max}, and when r is greater than 0. Also, when prefix is truncated (e.g. equal to 1111), the remaining level $L_{Rem} = L - L_{Max}$ is coded using EGk codes, and no suffix part of the TR code is presented.

- The prefix bins of Limited Exponential-Golomb (LEGk) code are limited based on the maximum dynamic range. In the worst case scenario, when Q_{step} is equal to 1, the range of quantized levels is $-2^{(B+6)}$ to $2^{(B+6)} - 1$, inclusive, where B is the bit-depth (e.g., for 16 b/p video the dynamic range is $-2^{(22)}$ to $2^{(22)} - 1$). In the case of 16 b/p, GR+EGk may produce the code words of more than 40 bins (47 when r is 0, and $k = r + 1$). Hence, to limit code length to the 32 bins, which is desirable for hardware and software implementation, the next modification has been used in the original HEVC. First, maximum transform dynamic range (dynamic range after the DCT) is given with $maxTrRange = max(15, B + 6)$. Then, the prefix length is limited in the way that $maxPrefixExtensionLength = 28 - maxTrRange$, and it is coded using truncated unary code. When maximum prefix length has been reached, suffix is modified to accommodate the rest of the possible codes, maximizing its size. In that way suffix length is given with exactly $maxTrRange$ bins, and it is coded as fixed length representation.

4.5.2 Processing and Coding

A coding of the CB is conducted only if non-zero elements exists after quantization. Otherwise, a flag that indicates a non-significant CB is signaled into the bit-stream. Also, the encoding process of the quantized levels involves several

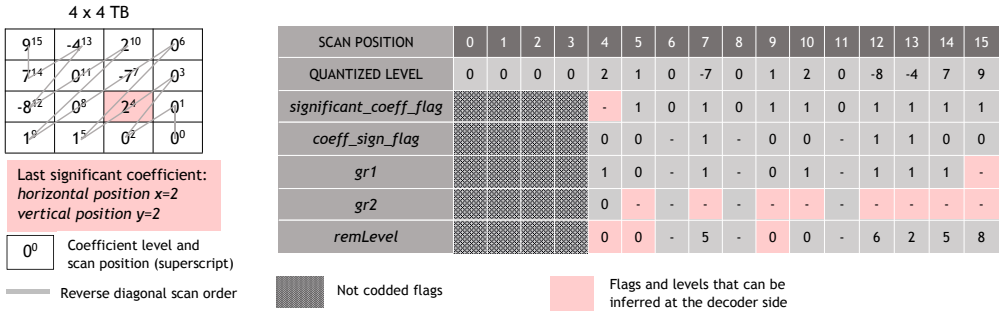
TABLE 4.3: An example of the binarization approaches for $L_{Max} = 15$. Shaded part for GR and EGk code represent prefix, and the rest represents suffix part of the code. Fixed Length (FL), Unary (U), and Truncated Unary (TU) codes are also included.

L	FL	U	TU	GR r=1	GR r=2	EGk k=0	EGk k=1
0	0000	0	0	0 0	0 00	0	0 0
1	0001	10	10	0 1	0 01	10 0	0 1
2	0010	110	110	10 0	0 10	10 1	10 00
3	0011	1110	1110	10 1	0 11	110 00	10 01
4	0100	11110	11110	110 0	10 00	110 01	10 10
5	0101	111110	111110	110 1	10 01	110 10	10 11
6	0110	1111110	1111110	1110 0	10 10	110 11	110 000
7	0111	11111110	11111110	1110 1	10 11	1110 000	110 001
8	1000	111111110	111111110	11110 0	110 00	1110 001	110 010
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
14	1110	111111...10 (14)	111111...10 (14)	11111110 0	1110 10	1110 111	1110 0000
15	1111	1111111...10 (15)	1111111...1 (15)	11111110 1	1110 11	11110 0000	1110 0001

key elements: 1) multi-level significance map coding, 2) scanning order of TBs within CB and scanning order of the quanitzed levels within the TB, 3) last significant quantized coefficient level coding, 4) quantized level sign coding, and 5) absolute quantized level coding.

The coding process starts by specifying weather the entire area of a particular CB consist of zero coefficients, or there exists at least one significant (non-zero) coefficient. If CB must be coded, which means that there is at least one significant quantized level, a similar inquiry should be applied for each TB in a given transform quadtree. If the TB is not significant, the coding process moves forward to the next TB in the quadtree. Otherwise, the procedure that will be described in the following text has to be deployed. To better understand coding process of a single TB, the Fig. 4.6 is included. It illustrates the general process on how HEVC encodes a single TB and all syntax elements associated with it, so that quantized levels are encoded in the most efficient way. The figure will be referenced through the following text on several occasions.

The process continues by encoding the coordinates/position of the last significant quantized level. In the Fig. 4.6 it is given with the coefficient on the fourth scan position (note reverse scan order). By this, some parts of the area



SCAN POSITION	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
QUANTIZED LEVEL	0	0	0	0	2	1	0	-7	0	1	2	0	-8	-4	7	9
significant_coeff_flag					-	1	0	1	0	1	1	0	1	1	1	1
coeff_sign_flag					0	0	-	1	-	0	0	-	1	1	0	0
gr1					1	0	-	1	-	0	1	-	1	1	1	-
gr2					0	-	-	-	-	-	-	-	-	-	-	-
remLevel					0	0	-	5	-	0	0	-	6	2	5	8

FIGURE 4.6: Coding of the quantized levels and associated syntax elements

within the TB, e.g., trailing zero coefficients at higher frequencies, have been skipped from the coding process. For example, in Fig. 4.6, these coefficients are associated with scan positions from 0 to 3, as they are positioned before the last significant coefficient in the scan order. In addition, for TB larger than 4×4, the subdivision has to be done into the smaller non-overlapping 4×4 processing units, called Sub-Blocks (SB). Note that 4×4 TB consists of only one SB. The encoding process is further completely transmitted to the SB level. The scanning order of the SBs is diagonal, but starts from the bottom-right and goes to the top-left SB. Hence, the reverse diagonal scan has been used [177]. For 4×4 and 8×8 intra prediction TBs, two additional scan mods may apply – vertical (column by column) and horizontal (row by row) scan, which depends on the selected intra prediction mode (also known as mode dependent coefficient scan) [178]. Again, it uses a reverse scan pattern, starting from the bottom-right to the top-left. The coefficient scan pattern within the SB (a total of 16 coefficients) has the same pattern as the SBs scan in TB. In this way, a vectorized form of the coefficients within one SB has been formed, explicitly defining the processing order of the coefficients within the block. It can be seen on the left part of the Fig. 4.6. After that, the significance flag has to be transmitted for each SB, indicating another level of significance – now weather SB has been covered by the non-zero quantized levels [179]. Finally, after the scanning process is done, and if all of the different levels of significance indicate that particular block – CB, TB and SB – have non-zero elements, each quantized level within each SB has to be encoded. Note once again that coding process of the TB starts with the SB that contains the last significant coefficient, and the SBs that are placed before a particular block in the reverse scanning order are skipped, see Fig. 4.6. A

more detailed overview of the coding techniques can be found in [26, 170, 171]. Note that this part has not undergone any changes since it is well modeled and no part of it has been negatively affected so far as the bit-depth range has been expanded.

In addition, it is important to understand the process of coding quantized levels, since it has undertaken several changes in the proposed codec due to the increased bit-depth range. The goal was to increase bypass coding bins in order to improve throughput, however without influencing performance of the codec, and also to use binarization schemes that will be adjusted to the new input data.

4.5.3 Coding of the Absolute Quantized Level

The coding of quantized coefficient levels starts with the coding of the significance map. It is composed of trailing flags for each individual coefficient within the SB, referred as significant coefficient flags *sigf*, assigning 1 for the non-zero coefficients and 0 for the coefficients with a level equal to zero, see Fig. 4.6. It is another level of significance coding in HEVC, and the last one, which corresponds to the significance of the individual coefficients within SB [179]. Thereafter, the sign for each significant coefficient is coded into the bit-stream.

To code the absolute quantized level, HEVC utilizes additional syntax elements such as

- *coeff_abs_level_greater1_flag* (*gr1*) indicates whether the absolute coefficient amplitude is larger than one.
- *coeff_abs_level_greater2_flag* (*gr2*) indicates whether the absolute coefficient amplitude is larger than two.
- *coeff_abs_level_remaining* (*remLevel*) indicates remaining absolute quantized level.

The aforementioned syntax elements have to be encoded into the bit-stream only for non-zero coefficients, which are indicated by the significance map. Again, Fig. 4.6 illustrates the use of the given syntax elements.

The given coding scheme in HEVC has been designed with the assumption that many quantized levels are zero or near zero. Also, the *gr1* and *gr2* flags are coded in regular mode. Because of that, HEVC uses a mechanism to increase throughput by limiting the number of occurrence of those two flags within the

one SB, smaller non-overlapping 4 × 4 processing units within TB, to 8 for *gr1* and to 1 for *gr2* [180]. In a case when the maximum number of the occurrence of the particular flags has been reached, it is assumed that the flag is set to 0, however without directly coding it into the bit-stream, see Fig. 4.6. In that way throughput is improved with negligible impact on performance. However, the CABAC in HEVC is still highly sequential. Many data dependencies come from the context modeling part, and the context update loop makes a bottleneck for CABAC. To improve the throughput, a reasonable number of bins that can be processed in one cycle has to be selected. This was achieved by pushing more bins to be coded in bypass mode, or limiting regularly coded bins as described in the previous example/approach. For example, a maximum of 16 *sigf* bins and up to additional 9 bins are coded in regular mode per each SB, making a total 25 regularly coded bins, e.g., 16 for *sigf*, 8 for *gr1*, and 1 for *gr2*.

Contrary to the standard HEVC's applications that naturally target higher compression ratios, in a given seismic data use-case scenario the focus is on achieving high quality of reconstructed signal in order to later-on analyze seismic data with more confidence. This can be only achieved with low and moderate compression ratios. That indicates that many of absolute levels after quantization still have high values when 32 b/p data has been used, since quantization will not cut the amplitudes to a significant extent so that most fall to a value close to zero (or to zero). As previously noted, the coding scheme in HEVC has been designed with the assumption that the quantized levels are zero or near zero. Thus, the HEVC's coding scheme which is described above is able to improve coding performance for a wide range of standard applications. However, it is feasible to assume that the standardized approach does not provide benefits when coding seismic data as it is case with natural video. Therefore, due to the high bit-depth expansion, low-level coefficients are less likely to occur, even at high frequencies (in the case of seismic data many of the coefficients after quantization still have high values of the absolute levels), and hence there is no benefit of using the original scheme. At the same time, if the coding scheme as in the original HEVC is still exploited, the complexity is increased if the context coded syntax elements like *gr1* and *gr2* are used, without any significant gain in coding performance. Instead, in this work it is further decided not to code the *gr1* and *gr2* flags. It is proposed to completely cut-off those regularly coded bins in order to additionally improve the throughput by pushing more bins to be coded in bypass mode. The proposed approach reduces the number of context models, which in addition lowers the memory requirements, and it reduces the

number of bins coded in the regular mode, which in turn reduces complexity. At the same time, coding performance has not been affected by the proposed simplification.

4.5.4 Binarization of Remaining Absolute Level

In the standard, the remaining absolute level is defined as $remLevel = level - (sigf + gr1 + gr2)$, where *sigf* represents a syntax element that indicates significance of the individual levels, e.g., whether it is zero or non-zero. However, all significant levels are coded as $remLevel = level - 1$, without exception, given the proposed changes with *gr1* and *gr2* syntax elements. Due to the large number of bins that are required to code quantized levels, all bins derived from *remLevel* are coded in bypass mode to improve throughput (bins coded in regular mode are processed sequentially, as one bin can affect others due to context modeling, as opposed to bypass bins that can be grouped and process in one cycle). Note, in the codec proposed in this work, *remLevel* is always present in the bit-stream, as opposed to standardized coefficient level coding in HEVC when, in some situations, the level can be inferred at the decoder side based on previously decoded flags. Hence, there is no need to be transmitted explicitly to the bit-stream every time. Such situation may be observed for example when $level = 1$, where $sigf = 1$ but $gr1 = 0$ and $gr2 = 0$. In this case, the codec can introduce some savings by omitting the encoding $remLevel = 0$ into the bit-stream every time. For example, in Fig. 4.6 this situation occur for quantized levels at scan position 5 and 9. Similarly, a quantized level at position 4 can be inferred at the decoder side without coding it into the bit-stream by only observing previously coded flags. Also, note that the coefficient at position 10 in the Fig. 4.6 cannot be inferred on the decoder side, even if $remLevel = 0$. However, due to the proposed changes in the coding of the $gr1$ and $gr2$ flags, which have been omitted in the proposed codec to improve throughput, the resulting codec cannot infer any level at the decoder side by just observing $sigf$. However, due to the targeted compression ratios (low and moderate compression ratios are targeted in seismic compression applications), where low levels around zero are less likely to occur, this change did not have any effect on the performance of the proposed codec, except that it has improved throughput (by not considering regularly coded $gr1$ and $gr2$ flags).

To perform binarization of *remLevel*, HEVC originally uses the adaptive process of the Rice parameter initialization to control the binarization, and the

combined Golomb-Rice (GR) and k-th order Exponential-Golomb (EGk) binarization scheme [170, 171]. GR codes are used for smaller values of *remLevel*, and the switching to EGk is done when the unary part (prefix) of the GR code reaches 4. Since the maximum value of the prefix part is known in advance, one bit can be truncated. In the standard, this is referred as Truncated Rice (TR) code. Then, the suffix is coded by using EGk codes, where $k = r + 1$, and r is the Rice parameter. In this way, HEVC encodes quantized levels using variable code lengths that consist of unary coded prefix and fixed length suffix (whose size depends on the prefix value and the Rice parameter [173]). More detailed explanation of binarization schemes is given in the previous section.

By using a given codes, codeword construction is not straightforward, and it depends on a tunable parameter r. For each SB, r is initialized with $r_{init} = 0$, and updated as follows

$$r_{next} = \begin{cases} min(r_{max}, r_{current} + 1), & remLevel > 3 * 2^{r_{current}} \\ r_{current}, & \text{otherwise,} \end{cases} \tag{4.19}$$

with the cap at $r_{max} = 4$ [181, 182, 183]. Thus, depending on the previously coded *remLevel*, r is either increased or kept constant, and it cannot be decreased inside the same SB. Due to extended range of absolute levels, aforementioned cap was omitted in order to better adapt binarization scheme to higher dynamic range. Also, for each SB, initialization has been changed in a way that r at the beginning of the SB is not necessarily set to 0. Rather it is given with

$$\begin{aligned} r_{init} &= round(\frac{c}{4}) \\ c_{next} &= \begin{cases} c + 1, & remLevel \geq 3 * 2^{round(c/4)} \\ c - 1, & 2 * remLevel < 2^{round(c/4)} \wedge c > 0 \\ c, & otherwise, \end{cases} \end{aligned} \tag{4.20}$$

where c is set to 0 at the beginning of the CTB and updated accordingly once for each SB. After the first level within SB is coded, the parameter c may be updated in order to be used in the next SB. Note that, in the extended bit-depth range, Rice parameter at the beginning of the SB is not necessary set to 0. However, it is updated in the same way as before, which is given with (4.19), with an additional modification that removes the maximum value limitation entirely. Also, the process of update is backward-adaptive, and exact derivation of the

coding parameter is supported at the decoder side. Thus, the proposed adaptive binarization scheme more efficiently adapts to the statistics of absolute quantized levels in the extended bit-depth range scenario. The given approach has adapted the Rice parameter faster than before to the real image statistics.

Also, the EGk codes with limited prefix length have been used for a larger amplitudes. At the time when GR code reaches 4 bins for the prefix part, switching to the EGk code has to be performed, in which case $remLevel = level - 4 * 2^r - 1$ has to be coded, and the resulting codeword is concatenated to the previously coded bins that emerged after GR binarization. By using the EGk binarization scheme, code length for $remLevel$ has been kept within 32 bins. It has been achieved by limiting prefix bins based on the maximum dynamic range (after the DCT), which in HEVC is given with $maxTrRange = max(15, B + 6)$. Then, the prefix length is limited in the way that $maxPrefixExtensionLength = 28 - maxTrRange$. When the maximum prefix length is reached, it is coded using truncated unary codes, and suffix is modified to accommodate the rest of the possible codes, maximizing its size. In that way, the suffix length is given with exactly $maxTrRange$ bins, and it is coded as fixed length representation. Obviously the given scheme is not applicable in the proposed codec since it cannot accommodate extended dynamic range after DCT. However, straightforward adjustment can make this approach practically feasible to empower the use of 32 b/p data. First, $maxTrRange = 37$ since DCT proposed here in this work keeps the norm of the original DCT. Then maximum prefix length has been modified as $maxPrefixExtensionLength = 47 - maxTrRange$, limiting prefix size to 10 bins. During the experiments, the selected prefix length has provided the best trade-off between performance and complexity.

By applying the simple modifications, this work was able to adopt CABAC to the extended bit-depth data statistics, reducing the complexity without compromising performance. It can be shown that even with straightforward modifications of CABAC the proposed codec performs better than other industry standards (see Chapter 6). Thus, the foundation of a conceivable entropy coding scheme has been established in this work for 32 b/p data. The part that is related to the enhancement of the current scheme is left for future research, e.g., to improve the performance a new binarization scheme and context initialization that would be specifically tailored for seismic data could be examined in future codec's evolution.

4.6 Rate-Distortion Optimization – A New Model for the Lagrange Multiplier

To decide on an optimal coding parameter set, for example, coding quadtree, prediction mode, motion vector and many others, encoder decisions are based on RD optimization that uses the method of Lagrangian multipliers [184,185,186]. In such case, the encoder tends to choose an optimal set of coding parameters $\{Para\}_{opt}$, where the goal is to minimize the distortion D with the given rate budget R_c. The expression is given with

$$\{Para\}_{opt} = arg \min_{Para}\{D\}, \quad \text{subject to} \quad R \leq R_c, \tag{4.21}$$

where $\{Para\}$ is the set of all possible combination of coding parameters defined by standard. In the RD optimization process, the previous constrained optimization problem can be transformed to an unconstrained optimization problem using Lagrangian multiplier method [186]. Hence, the next optimization problem where the encoder tends to minimize cost J over the set of coding parameters is given by

$$\{Para\}_{opt} = arg \min_{Para}\{J\} = arg \min_{Para}\{D + \lambda * R\}, \tag{4.22}$$

where $J = D + \lambda * R$, D is distortion measure, R is rate, and λ is the Lagrange multiplier which is applied to get the minimum RD cost.

In order to represent the distortion D, HEVC defines the distortion measure as

$$D = \sum_{i \in B} |s_i - s_i^*|^j, \tag{4.23}$$

where $j = 1$ for the sum of absolute differences (SAD) and $j = 2$ for the sum of squared differences (SSD). The sum is calculated between the original and reconstructed image samples s_i and s_i^* over a particular image block B. The SSD is used in all coding decisions, except for motion vector decision where SAD is used. On the other side, the rate R represents the number of bits required to signal all encoding parameters of a given block, including quantized coefficient levels and all associated side syntax elements. Also, due to different distortion measures, λ for motion estimation and λ used in other coding parameter decisions is not the same. However, a simple connection is established in HEVC,

where $\lambda_{motion} = \sqrt{\lambda}$.

Therefore, minimizing the given cost J, the encoder tends to select the most optimal quadtree structure, intra prediction modes, motion vectors, etc. The optimization problem given with (4.22) is used by the encoder to check every possible combination of coding parameters, and to find the best possible encoding result in order to achieve the highest possible compression. An exhaustive RD optimization process has been applied, which in brute-force manner (also referred as full RD) has to perform forward-pass (encoding) in order to get the rate R, and back-pass (decoding) to finally get the distortion D. It later compares all possible outcomes in terms of bit-rate and image quality degradation in order to make final decisions. For example, selecting coding parameters mean that one parameter set is better than the previous one whenever it occurs that the distortion increase is outweighed by the bit-rate savings multiplied by the Lagrange factor λ, e.g, $\Delta D > \lambda * \Delta R$. To clarify the previous statement consider the following example when we want to choose between two set of encoding parameters $Para\_1$ and $Para\_2$. Simply comparing the cost J of one parameter set, and the cost J obtained using another parameter set, the encoder can decide which parameters to use. Hence, $Parameter\_2$ should be used over $Parameter\_1$ whenever it occurs

$$\begin{aligned} &J(Para\_1) > J(Para\_2), \\ &D_{Para\_1} + \lambda * R_{Para\_1} > D_{Para\_2} + \lambda * R_{Para\_2}, \\ &D_{Para\_1} - D_{Para\_2} > \lambda * R_{Para\_2} - \lambda * R_{Para\_1}, \\ &\Delta D > \lambda * \Delta R, \end{aligned} \tag{4.24}$$

otherwise choose $Parameter\_1$. A given analysis can be applied for any encoder decision.

In addition, keep in mind that parameter optimization has to be performed on each block of an image. Accordingly, the cost of an image is approximated as the sum of the costs, given with (4.22), of the independent blocks B

$$\{J_{total}\} = \sum_{B} arg \min_{Para} \{J\}. \tag{4.25}$$

Similarly, using the principle of separable minimization and using a similar analysis as with (4.24), we can check if one CB should be divided into sub-CBs. Comparing the cost J of one CB at the depth d, with the sum of the costs for 4

sub-CBs at the depth $d+1$ we can make a split decision.

Also, RD optimization is not defined by the standard, but many of the foreseen mechanisms implicitly rely on it and significantly lose their efficiency if it is omitted. However, it is left to the implementer to choose the best way to select a set of the encoding parameters, which in essence will be dependent on the application and available computational resources.

As can already be concluded, the Lagrange multiplier directly controls the RD trade-off. For example, small values of λ correspond to high bit-rates and low distortion, while large values correspond to lower bit-rates and higher distortion. Therefore, the Lagrange multiplier plays a crucial role in operational encoder control. However, all three parameters of the cost function J are subject to optimization and a compromise has to be made. More importance can be given to D optimization or to R optimization by adjusting λ in the cost function J. In this regard, Q_p is the only effective parameter used to control bit-stream, and below we will see that λ depends directly on the predetermined value of Q_p (λ is function of Q_p). Steaming from the above, it is important to have precise encoder control in order to optimally allocate the desired resources, and the Lagrange multiplier plays a crucial role in that process.

In HEVC, λ is calculated based on the RD model that is provided for natural images, and it is given by

$$\lambda(Q_p) = 0.57 * 2^{\frac{Qp-12}{3}}. \tag{4.26}$$

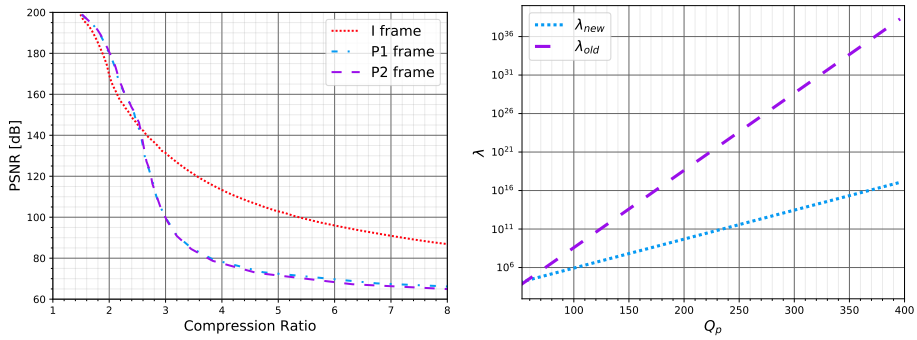
With the proposed extended Q_p range given in the Sec. 4.4, for certain higher values of Q_p, λ is also high which gives more importance to rate minimization without worrying much about distortion loss. As a result, the encoder tends to choose a parameter set that will reduce the rate as much as possible. In HEVC, the most rate conservative mode is the SKIP mode. By using the SKIP mode, the encoder can choose to force the residual block to be all zero, since it is worth on saving the number of the bits required to represent a given block, compared to the losses that are introduced in that way by truncating non-zero values, however with smaller overall cost. Since λ can be high, it will enforce the rate minimization and hence choose the SKIP mode much more often than it is desirable. As a result of the choice of SKIP without making compromise in the RD sense, a huge drop in performance is detected for Q_p values larger than 51, see Fig. 4.7a. The figure shows performance for the three consecutive frames of the same sequence (I frame and P frames that may be coded using SKIP mode), using the λ

as it is defined in HEVC's reference software. Naturally, P frames should have better performance than I frames since better prediction may be achieved, which in this case is not observed due to λ non-optimality. From now on λ given in (4.26) is referred as λ_{old}.

In order to achieve good coding performance using the proposed new codec, a critical consideration with the regard to the optimal coding parameter selection is determination of the proper Lagrange multiplier λ. It is crucial to have an appropriate λ for a high range of Q_p, in order to perform effective parameter allocation for the desired wide range of compression ratios and reconstructed image qualities. Thus, we propose to redefine HEVC's Lagrange multiplier in order to empower the use of extended bit-depth and quantization parameter range in the proposed new codec (as necessitated by 32 bit-depth extension).

A simple but effective heuristic approach was used in this work. Theoretically, λ can take values from 0 (highest rate, lowest distortion) to ∞ (lowest rate, highest distortion). However, in order to maintain the practical meaning of Lagrange optimization, suitable values for λ_{min} and λ_{max}, that will balance D and R, were found for those boundary points. Also, λ_{min} corresponds to $Q_p = 0$ and λ_{max} corresponds to $Q_p = MAX\_QP$. They are manually tuned to $\lambda_{min} = 0.0356$ and $\lambda_{max} = 1.702 * 10^{18}$. Also, by using the previously proposed extended Q_p and quantizer mapping function, a decision was made to limit MAX_QP to the value of 400. Thus, in this way, the search range can be reduced to $\lambda \in (\lambda_{min}, \lambda_{max})$ for any other Q_p in between 0 and MAX_QP. Since λ is a monotonically increasing function of Q_p, it stands that $\lambda(Q_p - 1) \leq \lambda(Q_p) \leq \lambda(Q_p + 1)$, and thus the search range can be additionally reduced. In this way we have been able to make it feasible to implement targets, with acceptable complexity and acceptable precision, that leads to a solution for any other values of λ. By using an extensive heuristic search, in this way we were able to find the exact value of the new λ for each of the newly extended Q_p. In addition, using this approach requires to store obtained λ for each Q_p in the look-up table.

To reach a solution that will not require look-up table, and thus relaxing memory requirements, newly obtained λ value can be compared with previous ones given by (4.26). It is proposed to compensate the large values of λ_{old} when $Q_p > 51$ and hence provide better compromise between D and R. We refer λ after we apply compensation as λ_{new}, and we can assume that it is still in the form of $k * 2^{\frac{Q_p - a}{b}}$ (the same form as λ_{old}). After observing newly proposed λ



(A) (B)

FIGURE 4.7: (A) Example of performance drop after $Q_p > 51$ using λ_{old} for the first three seismic slices of the same seismic sequence. (B) λ_{old} compared with redefined λ_{new}.

values that are stored in look-up table, it is decided to select the following values $k = 2.1$, $a = -49$, and $b = 8$. It leads to new recalculated λ that is given by

$$\lambda_{new}(Q_p) = 2.1 * 2^{\frac{Qp+49}{8}}. \tag{4.27}$$

Fig. 4.7b shows the comparison between the λ_{old} and the λ_{new}. It reveals a gap indicating that λ_{old} is very high compared with λ_{new} (the gap is increasing with Q_p) hence resulting in a low performance. However, as will be shown soon, the use of λ_{new} has improved the coding performance. More importantly, by redefining λ_{new} in the proposed new codec, precise coding parameter allocation is practically feasible.

4.7 Additional Coding Features

HEVC utilize several other (non-mandatory) coding features, for which it was decided to be completely excluded from the proposed codec. The reason behind is that they are either insufficient for a given use-case or they introduce additional complexity at the cost of insignificant coding gain. Thus, without going into a deeper discussion about their implementation, they were completely removed during the research phase of this thesis. Excluded features are: Intra PCM, Transform Skip, Residual Rotation, Implicit and Explicit Residual

DPCM, Intra Reference Smoothing, Strong Intra Smoothing, In-loop Filtering, Adaptive Q_p Selection, and RD Optimized Quantization. The only feature utilized by the proposed codec is Sign Data Hiding (SDH) [170], which is based on an approach in which the sign for the first non-zero coefficient in each SB does not have to be transmitted, but it can be inferred based on the parity of the sum of the absolute coefficient levels within SB. Since sign bits can take up a substantial proportion of compressed bit-stream according to [170], based on detailed experiments, this work choose to adapt SDH and keep it as an essential feature in the proposed codec. It is a consequence of the low computational complexity and improved overall coding gain. It is to note, the SDH is lossy technique since it changes the quantized level of the coefficients. Regardless, this technique is optional, and it can be enabled or disabled by setting a flag in the encoder configuration, e.g., when full lossless reproduction is required. In the following, only the SDH algorithm is described in detail, and the other methods are mentioned very briefly with references given for further reading.

- *Sign Data Hiding* (SDH) [170, 187, 188] is based on the approach where sign for the first non-zero coefficient in each SB does not have to be transmitted, but it can be inferred based on the parity of the sum of the absolute quantized coefficient levels within the SB (SDH is applied after quantization on a quantized levels). Hence, there is no need to always code the sign to the bit-stream explicitly, and it can be inferred at the decoder side. First, the sign will be coded in the normal way if the number of non-zero coefficients between the first and the last non-zero coefficient within a SB is below predefined threshold. In HEVC, the threshold is set to 4 non-zero coefficients between the last non-zero and the first non-zero coefficient within the SB [189]. If the number of non-zero coefficients is lower than the predefined threshold, the sign is codded in the normal way, otherwise it is encoded using the SDH approach. A predefined convention is used, in such way that the even sum corresponds to the positive sign "+", and the odd sum corresponds to the negative sign "-" of the coefficient level. If the parity of the sum of the quantized coefficient levels and the sign matches, the sign is inferred. If the parity does not match the sign, the level of the one coefficient within the current SB should be adopted accordingly. Hence, we can change one of the quantized coefficient levels by adding or removing 1, including the last and the first non-zero coefficient. However, in certain circumstances this change cannot be made on the first and the

last non-zero coefficient, e.g., if it becomes zero, since it can cause a mismatch between the encoder and the decoder (on the decoder side, the sign derived from the parity of the sum can be assigned to another coefficient, or it can lead to the number of non-zero coefficients being less than the predefined threshold). How to choose a coefficient whose value will be adjusted/changed, such that the parity of the sum corresponds to the sign, is left to the implementation. Two approaches have been applied in the HEVC reference software. The one we adopted for our codec is referred as RDOQ-off method in [188], which tries to find the coefficient whose change will have the least effect on RD performance. In other words, the choice is based on the minimal RD penalty that the level adjustment can produce. The difference between the original transform coefficient *coeff* and its reconstruction (dequantized value) has been calculated. The difference is referred as $deltaU$. If $deltaU$ is higher than 0, that means we should add 1 to the quantized level. Otherwise, if $deltaU$ is less than 0, we will decrease the quantized level by 1, in order to match the parity. The coefficient with the highest absolute value of $deltaU$ has been selected to be adapted. In that way, one quantized level has been adapted, so that the parity matches the sign, and the sign can be inferred. Note, since the SDH has been performed on the SB level, for larger TU sizes multiple sign bits can be embedded. Also keep in mind that this technique will change the quantized level of the coefficients, and hence cannot be used for lossless compression.

SDH is lossy technique since it changes the quantized level of the coefficients. However, the loss it introduce in the final RD performance is moderate even for the application like seismic image compression (that requires high quality reconstruction), but on the other side its compression potential is large. Regardless, this technique is optional, and we can choose to enable or disable it by setting the flag at the encoder configuration, e.g., when full lossless reproduction is required.

Since sign can represent a substantial proportion of a compressed bitstream according to [170], based on detailed experiments this work choose to adopt SDH and keep it as default option in the proposed codec, as consequence of the low computational cost and negligible distortion introduced for the price of the improved overall coding gain. Therefore, SDH has been shown to have a high impact on the final bit-rate.

- *In-loop filters*, such as Deblocking Filter [190] and Sample Adaptive Offset [191], are disabled in the proposed codec. Considering the application's scope, which usually requires high-quality image reconstruction, it is less likely that there is a significant amount of compression artifacts in the reconstructed data. Also, in-loop filtering has been designed to improve the visual quality of images, which does not necessary mean that it will carry a positive impact on a specific (possibly machine driven) analysis of seismic data. Also, in the related literature, there is a limited analysis on how artifacts that are introduced by using block based lossy codecs (by utilizing block structure, 2D transform, and quantization) further influence seismic interpretation and analysis. Thus, the implications of using (or not using) additional data filtering to correct compression artifacts (introduced by the lossy codec), within the scope of a specific application, may be the subject of additional work.

- *Intra PCM* is mode in which prediction, transform, quantization, and entropy coding are bypassed [192], which is why this mode is often referred as "copy-paste" mode (data samples are directly encoded into the bitstream).

- *Transform Skip* mode only omits transform from being applied to the residual data [193]. Residuals have been directly quantized and thereafter entropy coded. It was mainly adopted to improve the coding efficiency for specific video contents such as computer-generated graphics [194]. It has been designed to be used mainly in screen content coding extension of HEVC, where it provides significant coding gains, with at the same time less visible ringing and blurring artifacts and sharper edges [195]. However, it shows only marginal or no improvement for other classes (types of data).

- *Residual Rotation* applies the rotation of transform skipped TBs through 180 degrees prior to entropy coding [196]. It is noted that for intra predicted blocks that use the transform skip mode, the residue at the lower right part of the block may be relatively large, due to the distance between the reference samples (see Fig. 4.3a) and the predicted pixels [197]. This makes entropy coding, which is primarily designed for regular (transformed and quantized) blocks, inefficient when applied to transform skip

blocks [194], in which situation residual rotation may improve coding gain.

- *Implicit Residual DPCM* and *Explicit Residual DPCM*, where DPCM stands for differential pulse code modulation, are used to exploit the remaining correlation that could still exist in the residual signal, by using additional pixel wise prediction [198,199]. Sill, standard block based prediction is applied first, followed by pixel-by-pixel prediction. It exploits the immediately neighboring residual sample in order to predict the current one, which in addition reduce dynamic range prior to entropy coding. It is to note that DPCM is used only for transform skipped blocks.

- With *Intra Reference Smoothing* [200], the reference samples used for the intra prediction are filtered using 3-tap smoothing filter. Smoothing operations are adaptive, and they are based on the mode directionality and block size. While in a previous case only the first adjacent samples have been used, within *Strong Intra Smoothing* [201], an additional reference samples and bi-linear interpolation have been used for even stronger smoothing.

Chapter 5

Experimental Setup

5.1 Data Set

The used data set consists of seismic data taken from publicly available sources, such as those listed at https://wiki.seg.org/index.php/Open_data. This research uses a set of typical seismic data of different content and different resolutions. It is composed of 20 real-field 2D trace images, and 10 synthetic 3D wavefields each having 30 slices. For example, the Fig. 5.1a shows one representative 2D trace image from the used data set. The 808x3000 trace image (with 808 traces, 3000 samples per trace) correspond to the Alaska 2D survey, line 31-81 data. Raw image files are obtained by removing the SGY headers. Also, the seismic data from modern 3D surveys are typically represented with 32-bit floats. However, without losing on generality, data can be up-sampled to 32 bit integer values for more efficient compression (floats are harder to compress than integers). Since the codec proposed in this work uses integer arithmetic, a module is created in this work that selects scaling factor and converts the input floating-point image into 32-bit integers. Thus, the resulting integer image can be compressed using the HEVC based codec for better overall performance (since existing floating-point compression algorithms tend to compress floating-point data worse than those that rely on integer logic). On the other hand, scaling seismic data prior to compression does not affect the information content in the data, and without any fear of losses of information (other than those introduced by compression) the data can be scaled back to the original range at the decoder side with the same scaling factor that is used at the encoder.

Note that complete 2D trace data set consists of marine and land seismic survey data, downloaded from the aforementioned publicly available seismic exploration archives. No blank data regions (blank/empty slices) are presented

in the data used in this work. The 3D wavefield data set is composed of samples of the forward waveform propagation process generated by using synthetic velocity models and wave-equation methods (to simulate wave propagation trough defined subsurface models). In this way, by using numerical modeling, we are also allowed to record wavefields, also called snapshots. Each snapshot is represented by a 3D array, resembling video signal, e.g., characterized by two spatial and one temporal dimensions. For the purpose of illustration, two representative frames (which are also referred as time slices) from the 3D compression data set are presented in Fig. 5.1b and Fig. 5.1c. Note, the terms *frame* and *slice* are used with the same meaning in this work, thus establishing a connection with natural video.

In addition, the Fig. 5.2 shows two additional samples from the data set. These two samples comprise two consecutive frames of the same 3D wavefield sequence. Note the small structural differences between consecutive slices, which results in low-motion seismic data. If observed more closely, there are only minor differences between these two slices. However, they are structurally almost identical. This property will be analyzed in the next chapter of this work.

Also, the used data set is divided into two parts, namely *development set* and *test set*, in order to prevent over-fitting during the codec development, and in order to provide a fair comparison with other codecs. The development data set was used only during the proposed codec design. The test set was used to compare the proposed codec with other methods, and hence was not used at any stage that can be considered as a development stage during the codec design.

5.2 Experimental Setup

Experiments have been conducted on Linux workstation powered by two Xeon processors at 2.1GHz without hyper-treading, running 16 processes in parallel, and using 8.3.0 gcc compiler. The proposed codec has been implemented on top of the HM15.0+RExt-8.1 version of HEVC's reference software [149].

5.2.1 Code Setup

An anchored version of the codec was established to define the starting point for building the experiments. It defines a minimum set of coding tools that form

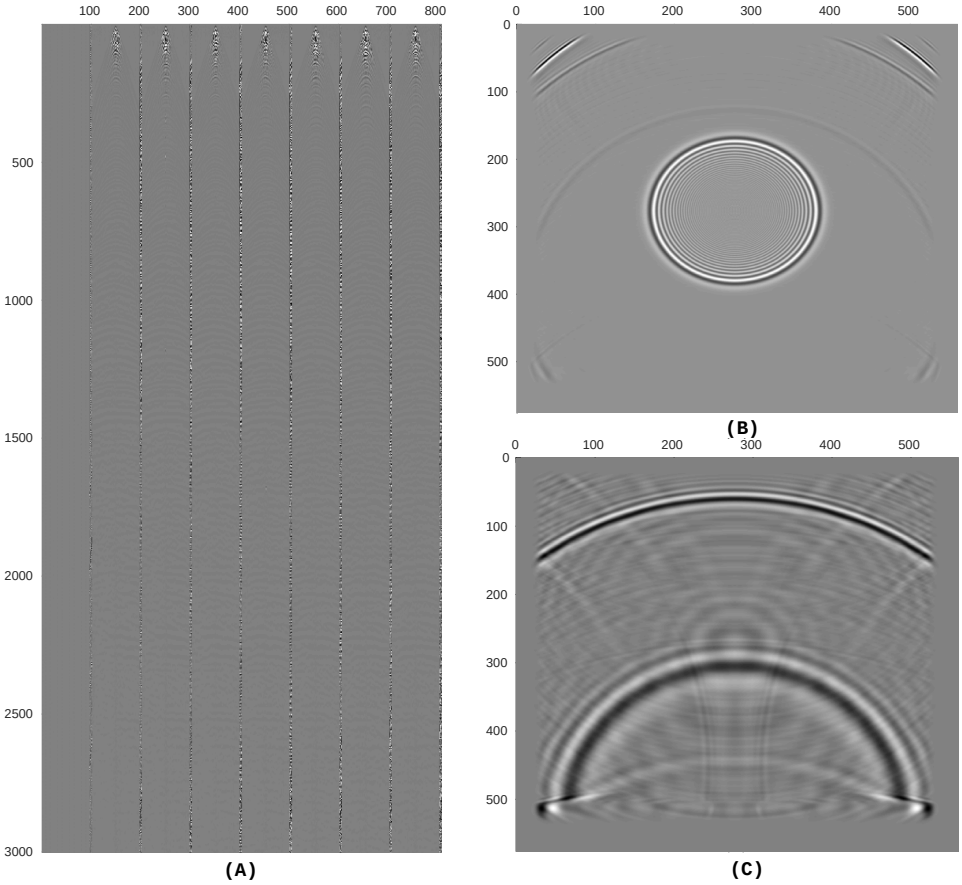


FIGURE 5.1: Sample input images. (A) Trace data corresponding to Alaska 2D land line 31-81 data; (B) 576x560 sample image which is part of a synthetic 3D wavefield; (C) The 2nd 576x560 synthetic slice.

the basis for further performance measurements of proposed modifications and approaches in the selection of coding parameters. In short, the proposed anchor version includes HEVC's variable block size and asymmetric motion partitions, as defined in Section 4.1, without further limiting block sizes below that already defined in standard. At first, only 8 intra modes are utilized directly in full RD optimization (in order to reduce unnecessary loads). Full motion vector search within a 64×64 window has been used (every integer point displacement has been searched), and half and quarter pixel accuracy have been utilized as well. Search range for bi-prediction ME has been applied in a neighborhood of 4, as

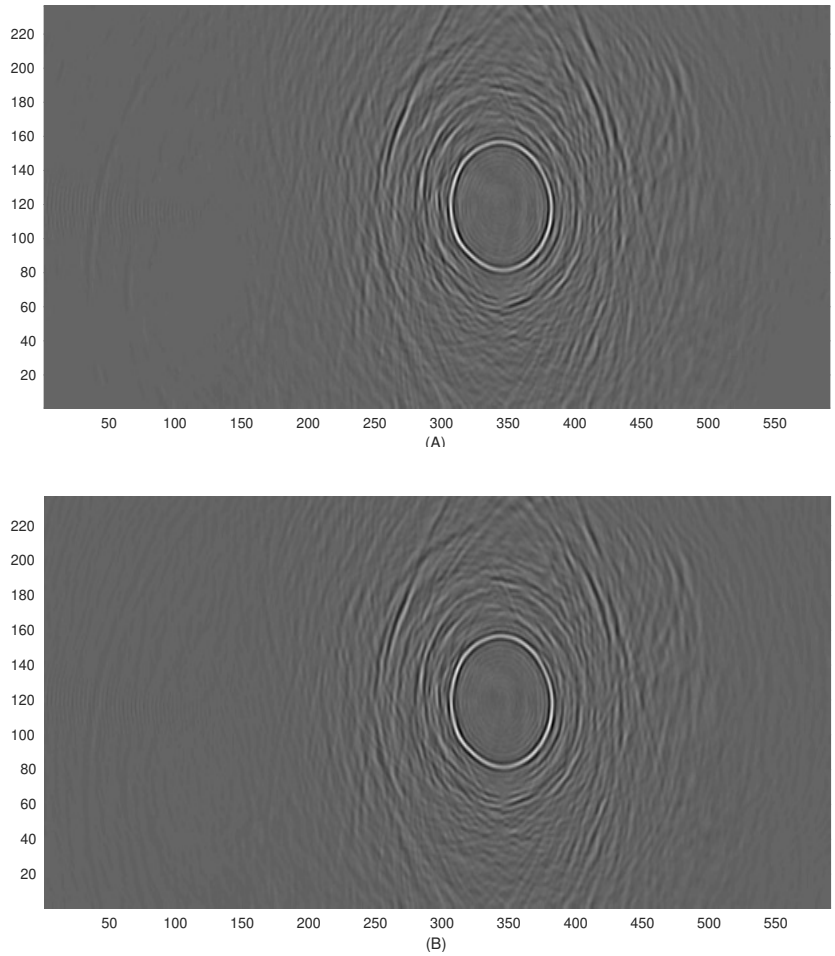


FIGURE 5.2: Two consecutive wavefield samples to illustrate low-motion seismic data. (A) 589x236 slice, which is 10th slice of the synthetic snapshot taken from the used data set. (B) 11th slice of the same snapshot.

it is default in the original HEVC setup. TZSearch is not used in anchor version. The number of AMVP candidates has not been changed, and temporally collocated predictor has been utilized. The same stands for the number of SKIP and MERGE predictor candidates. Quantization is given with (4.18). The anchor codec uses floating-point DCT, and slightly modified CABAC as explained in Section 4.5 that applies relaxed Rice parameter adaptation and changes in $gr1$ and $gr2$ flags (as this change does not have impact on the overall performance).

Also, the new λ function in (4.27) is exploited as an essential modification. SDH is the only additional coding tool used in the anchor version.

5.2.2 Objective Evaluation Criteria

Codec performance was evaluated using standard RD-curves, e.g., by measuring PSNR vs. compression ratio performance. Considering that PSNR-based quality metric is used during the development of HEVC, this work also applied the PSNR based metric during research. The maximum compression ratio shown in the graphs in the analysis of the results is 1:50, which is sufficient for targeted seismic data applications, even that a higher compression ratios are supported by the codec. In addition, the average difference between the two RD-curves, has been given in terms of Bjøntegaard delta PSNR measure (BD-PSNR), similarly as in [202,203]. However, compared to the standard BD-PSNR calculation, in this work the BD-PSNR performance corresponds to the average PSNR difference in decibels (dB) for the same compression ratio (CR) points, as it is illustrated in Fig. 5.3. In addition, RD-curves (reference and tested) are fitted by using third degree polynomial (cubic) interpolation. In the assessment used in this work, the average BD-PSNR is calculated between CR of 5:1 up to 45:1 (see shaded area in Fig. 5.3). Average performance is not measured outside this range for three reasons: 1) one assumptions is that lower compression ratios can be achieved with lossless techniques, acquiring perfect reconstruction, hence producing better performance than the presented lossy technique, 2) higher CR introduce notable losses in data, and would not be of particular interest for the targeted application, and 3) by limiting range of CR points used in BD-PSNR calculations, the use of extrapolations of the RD-curves outside the end points is avoided. Besides the average BD-PSNR characterization of the objective performance, for illustrative purpose, RD-curves are also plotted, with an additional enlarged part around CR 10:1 as it currently represents a typical compression ratio for seismic data applications.

At some points, the relative time of execution of encoding will be of great importance to approximate the complexity of one configuration relative to the other. It is given with $\Delta T = \frac{T_{tested} - T_{reference}}{T_{reference}} * 100\%$, where $T_{reference}$ is the encoding time using reference version, and T_{tested} is the encoding time using configuration whose complexity we want to compare with the reference configuration/method. Negative values reflect a time reduction in the execution of the

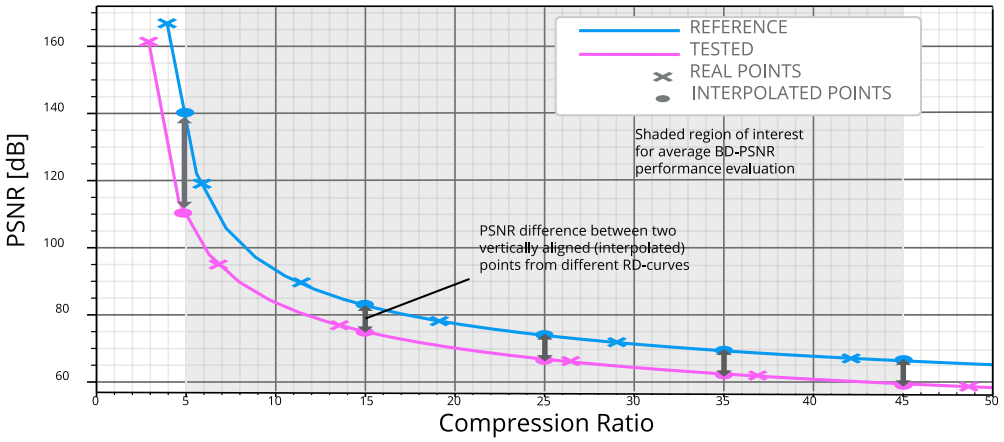


FIGURE 5.3: Illustration of objective performance evaluation by using the average difference in PSNR between two RD-curves.

tested approach. The same approach to roughly evaluate the complexity performance has been used during the development of HEVC.

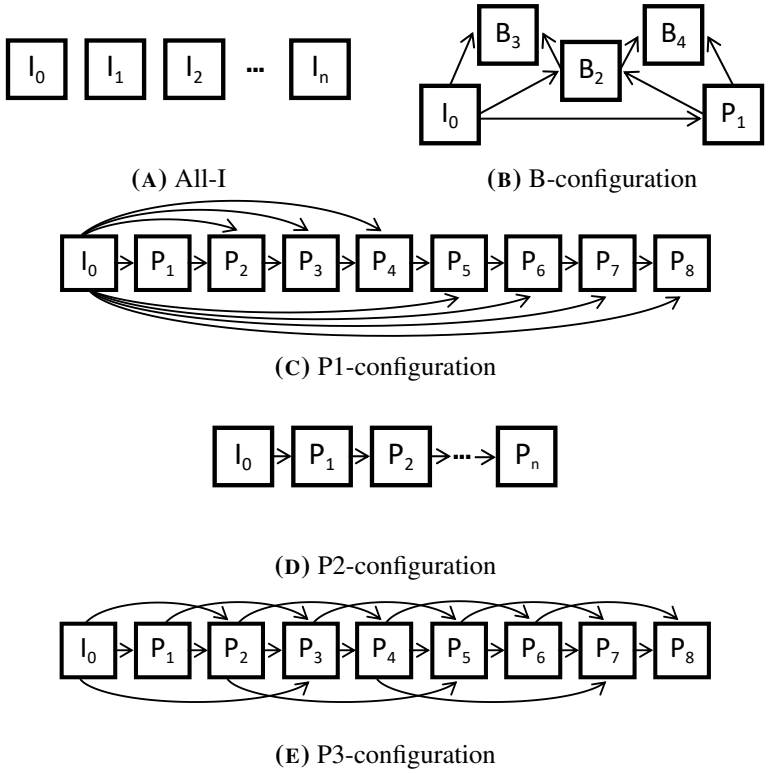
By the proposed quantitative assessment, BD-PSNR change of at least 0.2 dB, and computational complexity change of at least 5%, have been considered as significant.

5.2.3 Coding Configurations

During the experiments several coding structures were examined. Five coding configurations have been applied for the final testing. They were chosen to maximize performance under the constraints of the application (memory, processing time, low motion nature of seismic data, etc.). Long-term memory motion compensation and a larger number of reference images, sometimes as much as 50 frames according to [204], could be beneficial for standard video, e.g., when object is occluded in one frame but visible in others [205]. However, with seismic data, many scenarios that are valid for standard video would never apply. Even HEVC allows the use of up to 16 reference frames, the question is whether this would ever be useful in seismic data applications. Thus, there is a valid reason to adopt reference image management in order to relax memory requirements and simplify the encoding process (for more details on reference image management in HEVC see [205, 206, 207, 208]). Contrary to what is usually advised for

standard video, this work used a simple, yet effective strategy given the assumption that the correlation between seismic slices is stronger when their distance is smaller (from image order point of view). This assumption is based on the low-motion nature of seismic data. The nearest reconstructed images have been used as reference images, which we refer to as short-term memory referencing, and in some cases I frames are occasionally inserted into the bit-stream. Multi-reference image is still supported by the configurations, however it is tried to minimize their number in order to relax the size of the buffer. Therefore, coding configurations are given with, but not limited to, the following setup (see Fig. 5.4 for illustration)

- *All-I:* All seismic slices are encoded independently by using only intra prediction. This is equivalent to baseline still image coding approaches.
- *P1-configuration:* In addition, P frames are allowed to be used. All seismic slices within the group of pictures (GOP) use the I frame (the first frame in the GOP) and the first preceding P slice as a reference. The GOP size is set to 8, and the period of I frames is set to 8 as well. Hence, each GOP will start with an I frame. At most the two slices, that are used as a reference slices during predictive coding, are kept in the buffer at the same time.
- *P2-configuration:* The previous configuration is also compared with a configuration that uses only one previous P frame as a reference, without referencing to the beginning of the GOP, and without any additional periods of I frames. In such case, only one reference frame is used for prediction, and only one frame is stored within the buffer during the entire encoding process.
- *P3-configuration:* To test performance when using a different number of previous adjacent slices as a reference, instead to use only the previous slice, two and three preceding slices are used for referencing, which depends on a current slice position within the bit-stream. Even slices are selected to use two and odd slices to use three preceding reference frames. Only the first frame is I frame, and the rest of the sequence uses P frames, and GOP size is set to 8.
- *B-configuration:* In this configuration, some slices may have reference slices in both directions, hence it uses B frames as well. Such a reference



(A) All-I

(B) B-configuration

(C) P1-configuration

(D) P2-configuration

(E) P3-configuration

FIGURE 5.4: Cyclic GOP structure for different types of coding configurations.

structure can provide more efficient coding gain, however at the cost of increased complexity due to more demanding MV search. A maximum of three reference slices are stored in the buffer to support a given coding structure, and the GOP size is set to 4. Frames must be organized into the 3 temporal layers (compared to other configurations where all slices are the part of the same temporal layer). It can be used to provide fast access to data by reducing the sampling rate of the sequence.

Note that, the encoding order is the same as the decoding order, which is referred as a low-delay configuration, except for B-prediction where a different encoding order and output order occurs, but with as little delay as possible.

Chapter 6

Results and Performance Analyses

Several experiments are conducted in this work in order to evaluate performance of the proposed codec. The parameter configurations that are specifically adapted to the seismic data are also tested in this chapter.

6.1 Test 1 – Coding Configurations

The performance comparison between five different coding configurations given in Chapter 5 is shown. It is investigated how to adapt reference picture management in order to establish compromise between encoding efficiency and encoding complexity. From Fig. 6.1 and Table 6.1, it can be seen that B-configuration shows the best performance, followed by the P-configurations, and afterwards by image-based (All-I) configuration. Such performance is expected since many works for compressing natural videos have proved this. However, valuable insights arise when complexity is included in the analysis. Also, P1-configuration brings slightly lower performance than P3 due to the inserted I frames that are naturally harder to compress. The encoder complexity reduction of P2 coding configuration compared to other configurations comes from the fewer reference pictures, which entails a less demanding MV search. By observing presented results, Fig. 6.1 and Table 6.1, it can be seen that different P configurations show almost identical average coding performance. At the same time, the encoding complexity changes dramatically depending on the chosen configuration. It can be concluded that the increased number of reference slices does not improve coding gain for low-motion 3D seismic data, which is different from what is established for natural video [204]. Thus, using only one previous reference slice, as given with P2-configuration, is more than enough to maintain the best coding performance with the lowest computational complexity and the lowest memory

TABLE 6.1: Average BD-PSNR and complexity performance of different coding configurations, relative to the proposed B-configuration.

Configuration	P1	P2	P3	All-I
BD-PSNR [dB]	-2.57	-2.32	-2.31	-6.24
ΔT [%]	-2.89	-39.40	30.35	-98.20

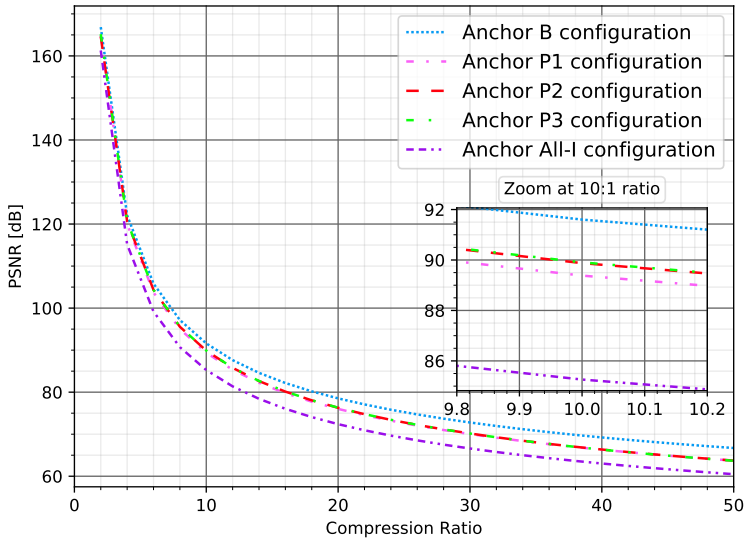


FIGURE 6.1: Average RD-curves for different coding configurations analyzed in Test 1.

requirements (in terms of the buffer size). For example, 39.4% less encoding time than B-configuration is required for P2, compared to only 2.89% less complexity when P1-configuration is used, and 30.35% increase in complexity for the most demanding P3-configuration. Note that All-I is the least demanding configuration, however with the price paid in lower performance. It requires 98.2% less processing time than that of B-configuration on average. Regarding to the above results, this work chose to keep the All-I, P2-configuration, and B-configuration in the following tests to support different performance-complexity trade-off and different use-cases (e.g., image based coding over predictive coding).

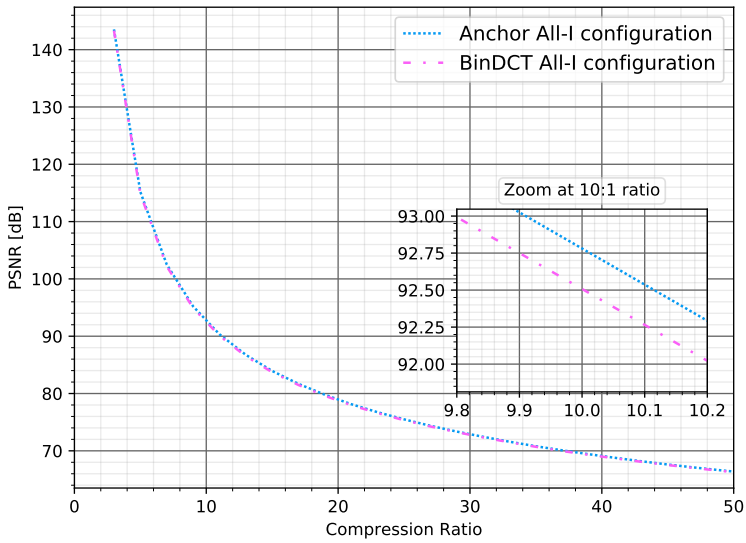
6.2 Test 2 – binDCT

Performance comparison between the original floating-point DCT and the proposed binDCT is summarized in the following. As expected, obtained results showed that binDCT gives slightly lower performance than the original floating-point DCT. Figure 6.2a and Figure 6.2b summarize average performance comparison of binDCT, for trace data and 3D wavefield data, respectively. For 3D wavefield sequences, average BD-PSNR loss of binDCT is 0.33 dB. Trace images showed lower coding gain drop, where observed BD-PSNR is 0.17 dB. At the same time the computational complexity savings are 21.3% on average, as compared to the original DCT with floating-point multiplications. Since the binDCT time savings are highly desirable and coding gain is still at high level, in this work it is decided to stick to binDCT in the sequel as a necessary change towards the practical solution. Also, as evaluated by industry experts, the binDCT time savings are highly desirable while coding gain is still at high level.

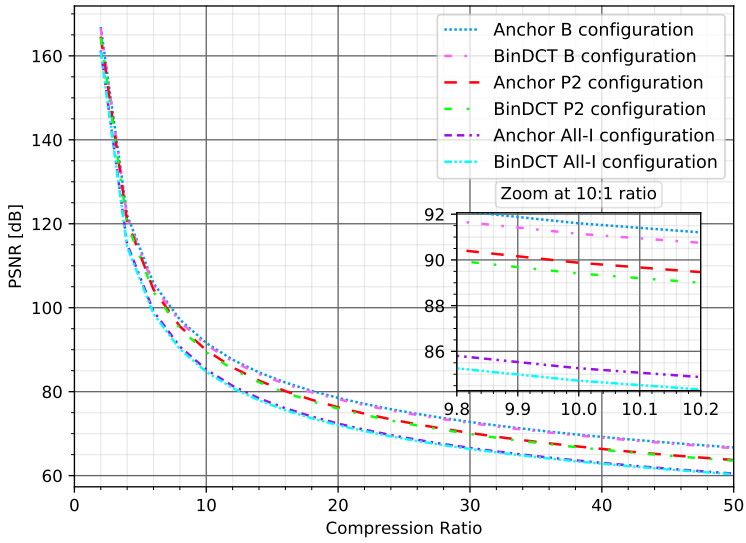
6.3 Test 3 – Block Size

From Table 6.2 it can be seen that, when the maximum allowed CB size has been reduced from 64×64 to 32×32 pixels, the negligible influence on performance has been observed, for both 3D wavefied and trace image data. In addition, when the maximum CB size has been reduced to 16×16, the coding efficiency drop (BD-PSNR) becomes significant. Note that the minimum CB size is not changed at this moment, as well as PB and TB standardized partitioning. Also, sensitivity of the proposed codec on the limited minimum CB size is investigated, while the maximum CB size has been kept at 64×64. When minimum CB size is increased to 16×16, the performance remains approximately the same, see Table 6.2. Increasing minimum CB size further to 32×32 brings a notable drop in performance that is no longer negligible.

Thereafter, asymmetric prediction block partitioning has been disabled in order to measure its contribution to the overall coding gain. In such case, the average coding efficiency is slightly affected by the imposed limitation. On top of that, remaining rectangular partition modes have been removed from the available codec's block structure, which show negligible coding efficiency drop. Bottom part of Table 6.2 summarize the coding performance of such PB partitioning limitations. We can see that significant computational time savings can



(A)



(B)

FIGURE 6.2: Average RD-curves for Test 2. (A) floating-point vs. binDCT average performance for trace image dataset; (B) floating-point vs. binDCT average performance for 3D wavefield dataset;

be achieved when all rectangular partition modes (including asymmetric) are disabled, with very low coding gain drop. Note that All-I configuration, and 2D (still image based) coding have not been affected by this experiment.

Furthermore, results show that restricting the maximum TB size to 16×16 lead to a significant coding efficiency loss. Also, performance when minimum TB size is set to 8×8 while keeping maximum transform size at 32×32 is evaluated, which also requires to set minimum CB to 16×16. Given that the standard requires that minimum CB size must be strictly greater than minimum TB size, here it is required to set minimum CB to 16×16, which in the previous experiment has been shown to have no influence on coding performance. Note that limited minimum TB size resulted in decreased coding efficiency, Table 6.2. However, also note that trace images are more affected by the test limitation compared to 3D wavefields. Given the previous results, we can conclude that the codec is highly sensitive to TB partitioning constraints, and that variable transform size contributes greatly to improving the performance of the codec.

Guided by the previous experiments, it is decided to adapt block partitioning, and to optimize encoder by continuing with the following coding choices: 1) maximum CB size is set to 32×32 instead to 64×64; 2) minimum CB size is set to 16×16 instead of 8×8; 3) only square PB partition modes have been used; 4) minimum and maximum TB size has been kept unchanged, as well as relative residual quadtree depth which is set to 3 (as default in HM). Thereafter, the accumulated coding efficiency loss vs. total encoder speed-up after adjusting the block partitioning is analyzed. Results are illustrated on Fig. 6.3a and Fig. 6.3b for trace and wavefield data, respectively. It is observed that the proposed adaptations lead to substantially large speed-up gain, which on average goes to 85.22% for P and B configurations, and 26.62% for All-I configuration (when compared to the previous binDCT test). At the same time, almost the same coding performance has been observed, with 0.15 dB drop in PSNR for 3D predictive coding and only 0.05 dB drop for All-I configuration. It is important to note that the further experiments and performance comparisons are provided on top of aforementioned configuration choices.

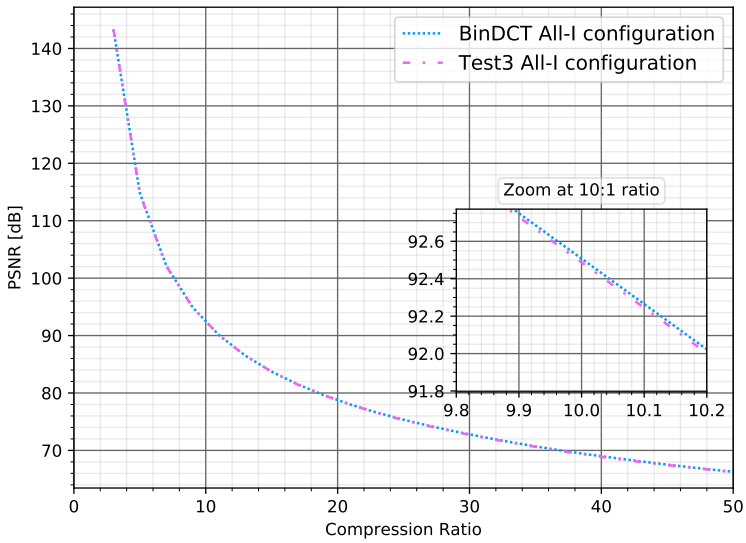
6.4 Test 4 – Intra Modes

After compressed bit-streams are analyzed, it can be observed that modes 0, 1, 10 and 26 (planar, DC, horizontal and vertical) are chosen more often, followed

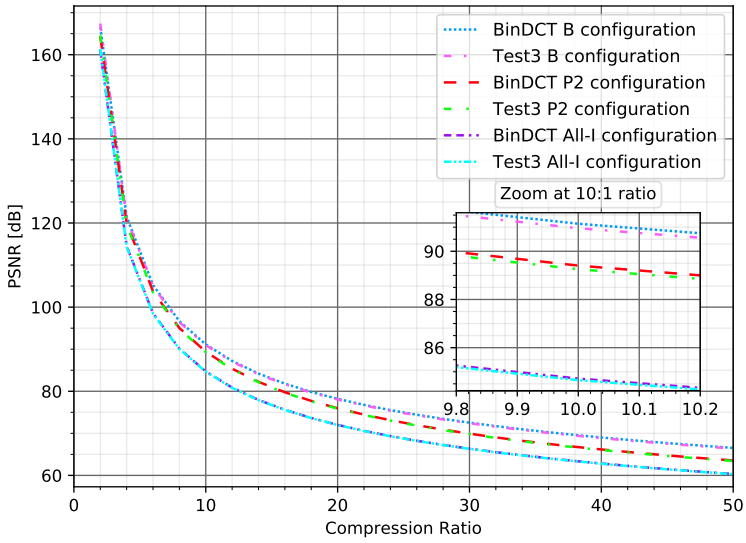
TABLE 6.2: Average BD-PSNR and complexity performance for different block partitioning choices.

Maximum CB size restrictions performance relative to 64x64 CB				
	32x32		16x16	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.03	-14.33	-0.88	-37.56
Wavefield	-0.02	-23.33	-1.32	-50.47
Minimum CB size restrictions performance relative to 8x8 CB				
	16x16		32x32	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.09	-11.21	-0.28	-22.14
Wavefield	-0.05	-21.8	-0.35	-39.34
Prediction block partitioning limitations				
	asymmetric PB off		all rectangular PB off	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Wavefield	-0.03	-47.32	-0.09	-80.15
TB size restrictions performance				
	maximum TB 16x16		minimum TB 8x8	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.92	-8.70	-1.23	-35.49
Wavefield	-1.24	-3.22	-0.12	-30.25

by those around the vertical and horizontal directions. The same pattern of mode distribution for different Q_p values was observed. In such way, it is decided to use the following 23 intra modes in total {0, 1, 2, 4, 6, 8, 9, 10, 11, 12, 14, 16, 18, 20, 22, 24, 25, 26, 27, 28, 30, 32, 34}, which combines modes 0, 1, 10, and 26, than those close to the vertical and horizontal directions, and every second mode in between. Additionally, for a benchmark purpose, test are conducted when only the following 12 intra modes have been used {0, 1, 6, 8, 10, 12, 14, 22, 24, 26, 28, 30}. PSNR vs. compression ratio performance evaluation, Table 6.3 and Figures 6.4a and 6.4b, indicate that the average PSNR loss is almost negligible when 23 subsampled modes have been used, and distinctly higher PSNR drop has been observed by using only 12 subsampled modes. We comment that the modest speedup (ΔT) is the result of already optimized intra coding loop



(A)



(B)

FIGURE 6.3: Average RD-curves for Test 3. (A) final performance after Test 3 for trace dataset; (B) final performance after Test 3 for 3D wavefield dataset.

that uses a simplified HEVC encoder (see Chapter 4). Even when the number

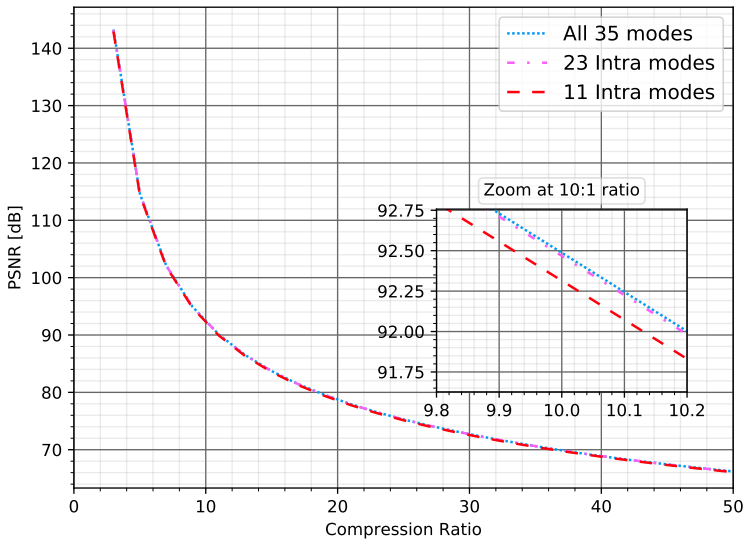
of total modes is significantly reduced, the same number of modes is always utilized in much more demanding brute force (full RD) optimization approach. Despite that, it is decided to keep 23 modes in follow-up experiments since negligible performance loss has been detected in such way. It is worthwhile to point out that uniformly decimated version of the 35 modes is tested by taking every second and every third angular mode, however results showed lower coding performance when compared to the proposed subsampling scheme.

Next, it is investigated how to configure the optimal N candidates for full RD optimization in the simplified HEVC encoder. It is to note, in the anchor version 8 intra modes are used in full RD optimization loop. In this test, N is varied between the following set of values $N \in \{16, 12, 8, 4, 3, 2, 1\}$. Performance is summarized in Table 6.3, and Fig. 6.5a and Fig. 6.5b. Experiment reveals that increased number of candidates (12 or 16) does not lead to changes in compression performance, however computing resources have increased significantly, see Table 6.3. It also can be seen that, the use of only 4 modes for full RD optimization provides optimal trade-off between computational savings when compared to the utilization of 8 modes, with at the same time negligible compression performance change. Reducing in addition N to 3, 2, or 1 shows significant performance loss that is no longer negligible. Thus, it is proposed to use $N = 4$ as an optimal number of intra modes for full RD optimization (together with 23 intra modes in total) which does not degrade compression performance. By doing so, computational requirements have been relaxed significantly.

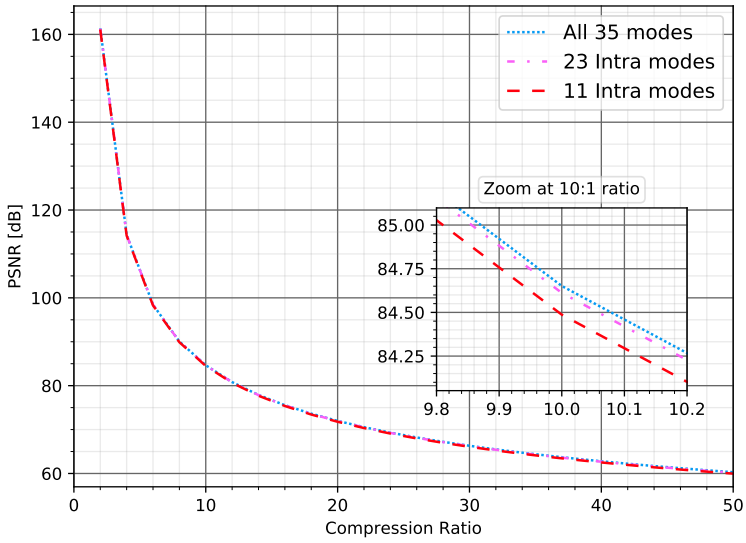
6.5 Test 5 – Motion Estimation

ME is computationally the most time consuming part of the codec. However, by taking into account the low-motion activity of seismic data, the encoder can be unburden of unnecessary calculations. Table 6.4 reports the performance during several stages of adjusting the motion estimation process to the seismic data.

It is started with the default motion search range within the 64-pixel neighborhood in anchor version, and in addition, the search range is reduced to only 1-pixel neighborhood. Also, the search range for bi-prediction ME refinement has been set to 1. Thus, integer-precision MV search has been reduced significantly without compromising compression performance.

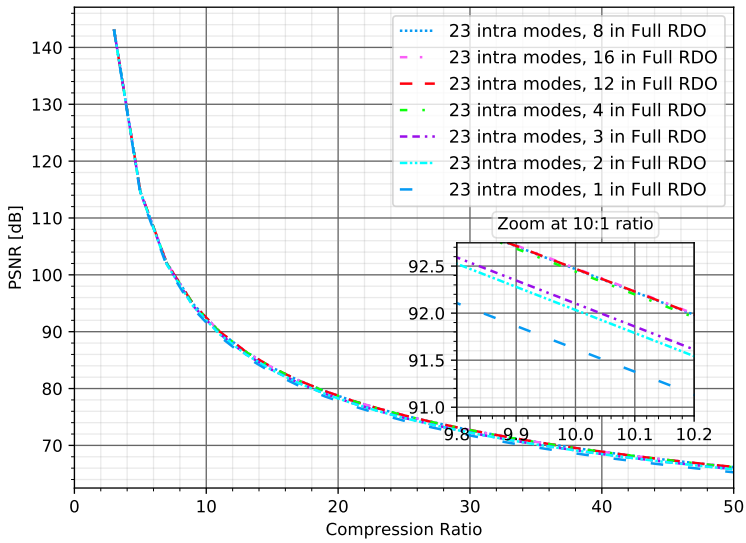


(A)

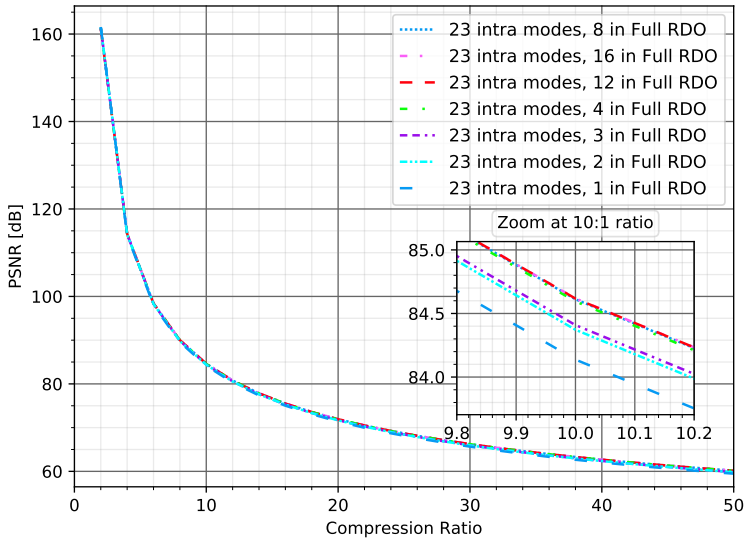


(B)

FIGURE 6.4: Average RD-curves for Test 4. (A) All 35 intra modes vs. reduce number of intra modes for trace dataset; (B) All 35 intra modes vs. reduce number of intra modes for 3D wavefield dataset.



(A)



(B)

FIGURE 6.5: Another average RD-curves for Test 4. (A) Reduced modes for full RDO for trace dataset; (B) Reduced modes for full RDO for 3D wavefield dataset.

In order to additionally reduce the number of integer-precision points, the

TABLE 6.3: Average BD-PSNR and complexity performance for intra mode modifications.

Reduced number of intra modes compared to 35 modes utilization				
	23 modes		12 modes	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.01	-2.87	-0.09	-6.33
Wavefield	-0.07	-3.09	-0.18	-6.82
Optimal N candidates for full RD compared to N=8				
	N=16		N=12	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	0.01	87.37	0.003	43.54
Wavefield	0.006	86.26	0.005	42.86
	N=4		N=3	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.02	-41.74	-0.39	-51.64
Wavefield	-0.01	-41.54	-0.24	-51.41
	N=2		N=1	
	BD-PSNR	ΔT [%]	BD-PSNR	ΔT [%]
Trace	-0.43	-60.68	-0.88	-67.89
Wavefield	-0.26	-60.57	-0.54	-68.10

modified TZSearch is investigated. Since search range is already reduced to 1, TZSearch use only diamond search without proceeding to raster search in a second stage. However, in the refinement stage, which is derived if the distance obtained from the previous stage is not zero, additional points may be checked in the RD sense. Since in the star refinement the number of additional points is not known in advance (search is repeated until the best distance equals zero), it is decided to prevent the use of this fine tuning step. That led us to only 4 integer-precision points that have to be checked within RD optimization. In addition, TZSearch employs additional motion vector prediction (different than the one given with AMVP). It uses left, upper, upper right, median, and zero predictors. However, after conducting several experiments, it is decided to allow only the use of median predictor and zero-MV predictor to compete for MV prediction. Temporal predictor and other spatial predictors are not used in the proposed

setup. Performance of modified search range and TZSearch is presented in Table 6.4 and Fig. 6.6a. Proposed modification lead to almost identical overall coding performance, introducing significant complexity reduction. Even modified TZSearch may lead to the suboptimal MV estimation, it is able to maintain high coding gain with negligible performance changes.

Thereafter, in this work, quarter-precision and afterwards half-precision MV estimation is disabled in order to quantify its contribution to the overall coding gain. According to the experiments, it is found out that low-motion seismic data highly benefit of such fine MV tuning. Notable BD-PSNR drop has been observed when quarter-precision estimation is disabled. When half-sample precision is removed from the consideration (on top of the excluded quarter-precision), the loss in performance became much more evident. Bear in mind that higher losses are noticed for P-configuration since B-configuration could compensate lack of fractional motion estimation precision by using averaged motion predictions from both directions. Also, if fractional MV estimation is completely excluded, computing resources are slightly lower, which in comparison to the performance losses that are quite pronounced does not bring equally proportional computational saving. Even that interpolations used to obtain fractional samples introduce additional complexity, and additional points have to be checked in the RD sense, such a small time reduction is result of the simplified search process which use Hadamard measure for fractional ME. Thus, this work still fully exploits fractional ME, since the ability of the encoder to track motion vectors by going below integer pixel precision proves to be beneficial in order to maintain high performance for low-motion seismic data.

Since temporal predictor within AMVP scheme requires considerable storage capacity for motion related data of the collocated slice, in this work it is decided not to use it in the proposed setup. Also, its impact on the improved codec's performance is negligible. Lastly, the number of competitors in MERGE mode is adjusted according to the tests, where the number of merge candidates is between 1 and 5. After conducting results of the tests, the number of merge candidates is set to 2, which represents the best trade-off between coding performance and encoding complexity.

Also, for a benchmark purpose, and in order to show the importance of using even *minimal (truncated) ME*, performance when there is no-motion estimation is evaluated. Option to skip all motions is provided in order to use only one-to-one prediction. Prediction in inter coded PB is obtained as a difference between current PB and the corresponding block at the exactly same position in

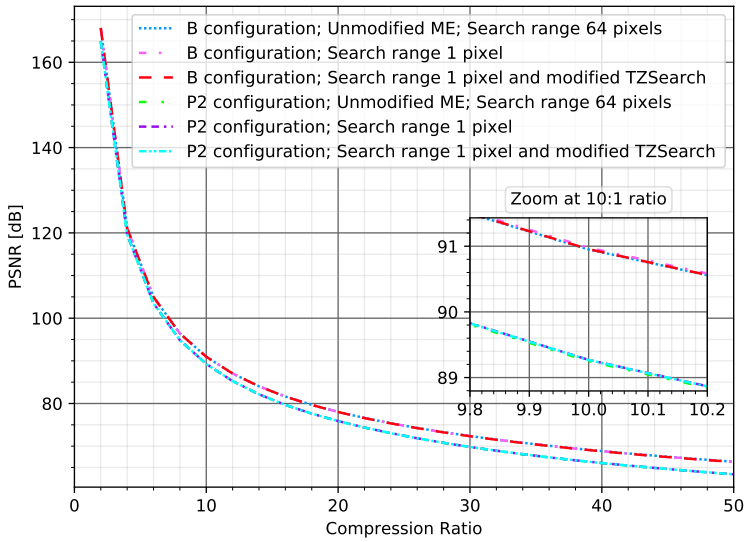
TABLE 6.4: Performance comparison of the proposed modifications within motion estimation loop.

	Search range = 1	**Modified TZSearch**
BD-PSNR [dB]	-0.01	-0.01
ΔT [%]	-84.03	-85.21
	Quarter-precision off	**Half-precision off**
BD-PSNR [dB]	-0.29	-1.28
ΔT [%]	-4.32	-6.77
	TMP off	**Merge candidates = 2**
BD-PSNR [dB]	-0.15	-0.04
ΔT [%]	-0.16	-23.32

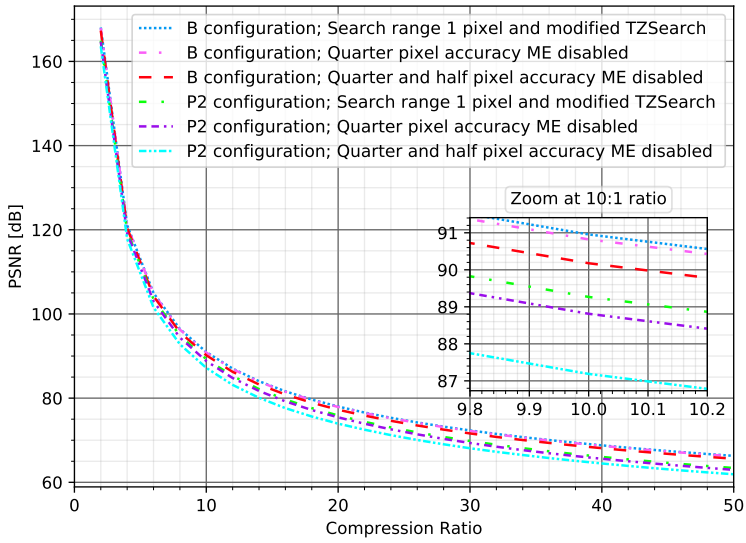
the reference slice. Therefore, residual signal can be computed on the slice level subtracting the whole slice with its reference slice (e.g., one time subtraction of a complete slice in pre-processing stage). Clearly, the one-to-one prediction performance is degraded (compared with truncated ME), and PSNR loss of 3.5 dB compared to P-configuration and 5 dB compared to B-configuration is evaluated when no motion compensation is used. Therefore, the use of even minimal (truncated) ME has been shown to be substantial for performance improvement.

6.6 Summary of Test 1 to Test 5

The given results reveal some useful findings that helped to propose the optimized encoder according to the application scope. Results also imply that the optimized encoder may find better trade-off between the coding efficiency and computational complexity. The performance difference due to the proposed adjustments is plotted in Fig. 6.7. When linked together, the proposed optimized codec reduces complexity by the 67.17% for All-I configuration on trace image dataset, and 67.39% for All-I, 97.96% for predictive P-configuration, and 98.63% for B-configuration on 3D wavefield dataset. At the same time, BD-PSNR loss is 0.27 dB on trace data, and 0.49, 0.65, and 0.64 dB on 3D wavefield dataset for All-I, P- and B-configuration, respectively. The coding performance is given relative to the initial non-optimized (anchor) codec, see Fig. 6.7. Lastly,



(A)



(B)

FIGURE 6.6: Average RD-curves for Test 5. (A) Search range set to 1-pixel neighborhood and modified TZSearch performance; (B) Performance comparison when fractional ME is disabled.

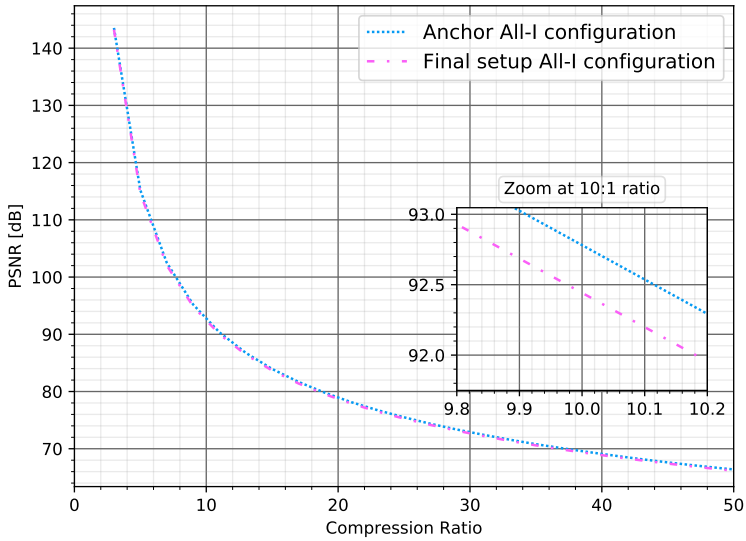
we note that decoder computational performance is approximately the same for all configurations on both datasets. On average it is 16.6% less than that of the anchor decoder implementation. This final set of tools and methods reflect the application needs as envisioned by the industry (in terms of performance and complexity).

6.7 Comparative Results With Other Methods

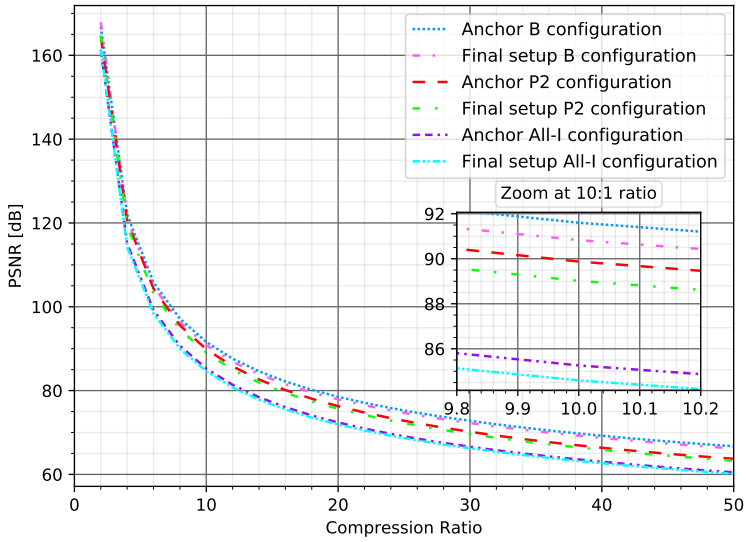
Fig. 6.8 plots the PSNR vs. compression ratio performance of the proposed codec with JPEG-XR. In addition, Fig. 6.9 plots performance comparison of the wavelet-based codec for one trace image (Alaska 31-81 line), and one 3D wavefield sequence. Unfortunately, during this work we have passed Shell's contract period for the project and hence do not have a license to run the wavelet-based codec to generate results for the entire dataset. However, based on the available results, it can be seen that wavelet-based codec performs similarly as JPEG-XR over the wide range of compression ratios, see Fig. 6.9. Thus, average performance comparable to the one obtained with JPEG-XR could be expected over the complete dataset for the second codec as well. Therefore, compared to a JPEG-XR, which performs on par with licensed commercial wavelet-based codec, the proposed new codec significantly improves the PSNR vs. compression ratio performance.

For JPEG-XR (similar improvement in PSNR is expected for the second codec) average BD-PSNR gain is 5.86 dB for the trace dataset (All-I configuration). For the 3D wavefield dataset, performance improvement is 3.39 dB for All-I, 7.15 dB for P2, and 9.48 dB gain in PSNR is observed for B configuration, on average. A particularly pronounced improvement in PSNR when compared to other codecs can be observed for inter based predictive configurations (P2 and B).

The overall performance of the new proposed 32 b/p codec shows that the modified HEVC framework is suitable for seismic data compression since it improves coding gain by a large margin when compared to the benchmarked solutions. The executable version of the proposed codec is also given to the geologists to evaluate the impact of lossy compression, e.g., uniform quantization noise, on interpretation of seismic images. Since the proposed new codec is performing much better than the one reported in [1] that the geologists currently use, they rate codec's performance as highly satisfactory.

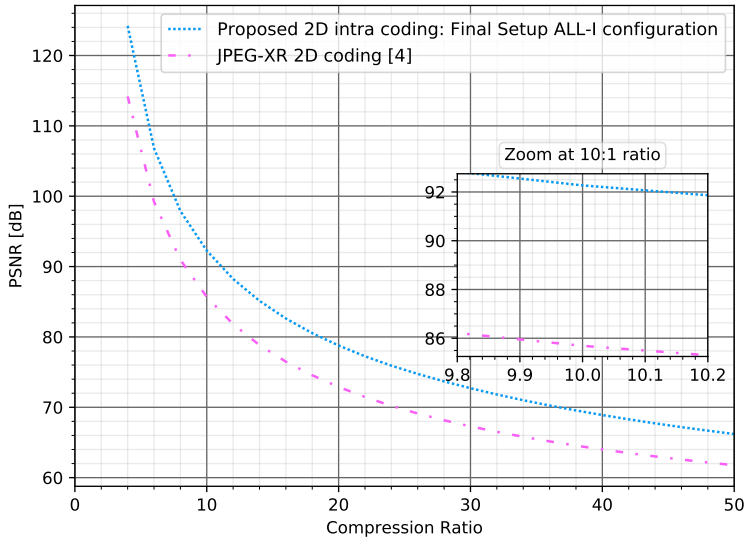


(A)

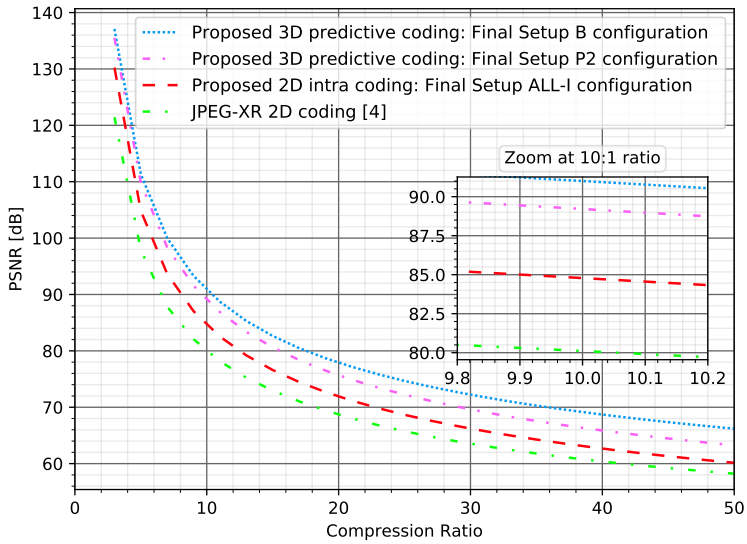


(B)

FIGURE 6.7: Average performance comparison between the anchor and the optimized codec after Test 5 using: (A) trace image dataset, and (B) 3D wavefield dataset.

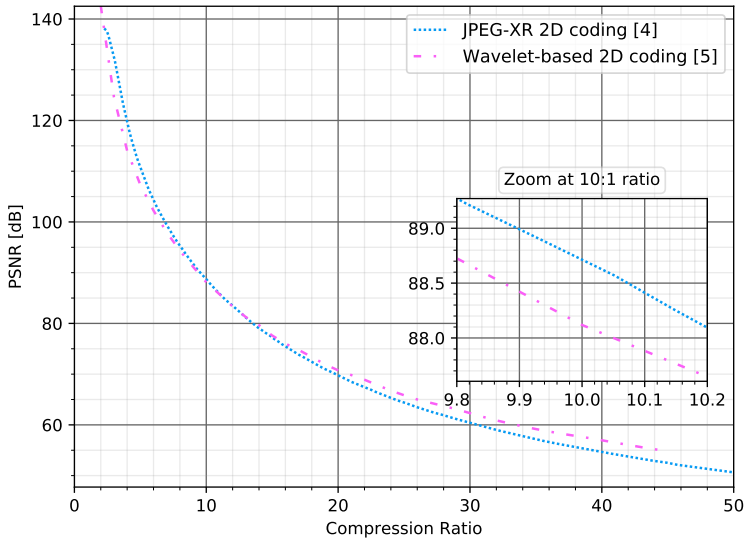


(A)

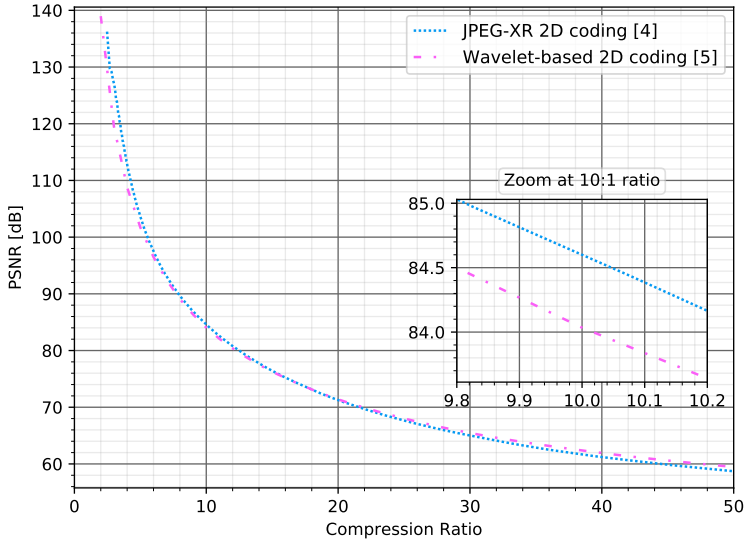


(B)

FIGURE 6.8: Average PSNR vs. compression ratio performance of the proposed codec. JPEG-XR [2] is included for comparison. Figure (A) corresponds to trace dataset, and (B) to 3D wavefield dataset.



(A)



(B)

FIGURE 6.9: The licensed commercial codec based on [1] vs. JPEG-XR [2]. (A) trace image, and (B) 3D wavefield sequence.

Chapter 7

Conclusion

The apparent need for research in the field of the seismic data compression is obvious and is constantly increasing. Thus, this work presented the design of a new, custom tailored, codec for high bit-depth data under the framework of HEVC, and reported very good results using it for the compression of 32 b/p seismic data. Almost all parts of the original HEVC have been modified. The proposed results show that the codec already performs significantly better than existing solutions used in the industry, and codec's subject performance is rated by Shell's geologists as highly satisfactory. This work also accomplished to find better tradeoffs between the coding efficiency and computational complexity, resulting in the proposed optimized encoder. In addition, the presented work broke down the new method into its structural parts as much as possible and gave a detailed analysis of potential bottlenecks and pointed out the parts that could be improved in order to evolve the proposed codec. Therefore, this work represents an initial effort to provide valuable insights of using one well established coding approach, such as HEVC, for the purpose of seismic data compression. New possible features, that will be specifically tailored for seismic data, should be explored in the future. Also, enhancements and additional modifications of existing features, in order to advance the proposed codec, should be the focus of future research to lead to even greater compression gains. At the same time, besides working on the codec's enhancement, future emphasis could be on GPU acceleration and further reduction of the codec's computational complexity (to make the codec run in real time). In that sense, the plan is to provide detailed speed-up numbers and complexity comparisons with other methods (since the HM code is not optimized to a commercial application level). Without code optimizations (HM reference software optimizations), a fair comparison with other methods in terms of run-time complexity is not straight-forward.

As a side contribution of this paper, HEVC is analyzed within all its functional units, so that the presented work itself can serve as a specific overview of the methods incorporated into the standard.

Although a significant part of the effort around this work was related to the development of an efficient compression scheme for high bit-depth seismic data, a different type of issues that are worth considering in the future are encountered. These issues are not technical in nature and mainly relate to: 1) Data availability and data documentation. Seismic data are not common data in the compression community. Some well-documented and diverse data set specifically tailored for the purpose of seismic data compression research could be very useful for reducing learning curve and encouraging further scientific research. A one well established off-the-shelf data set can be very useful for researchers that does not necessarily have a geoscience background; 2) A unique benchmark procedure to compare results among different methods and groups, without need to run source codes; and 3) Availability of other source codes for more detailed comparison of the proposed codec with other methods. In this study, we were limited to the codecs that Shell provided to us in order to compare our method. If these issues are addressed in the future, the author believes that the benefits to the research community and industry could be significant, and that this could encourage further interest within research communities for compression of seismic data.

Bibliography

[1] A. Z. Averbuch, F. Meyer, J.-O. Stromberg, R. Coifman, and A. Vassiliou, "Low bit-rate efficient compression for seismic data," *IEEE Trans. Image Process.*, vol. 10, no. 12, pp. 1801–1814, 2001.

[2] D. Taubman and M. Marcellin, *Recommendation T.832 (01/2012): Information technology - JPEG XR image coding system - Part 2: Image coding specification*, ITU-T Std., Jan. 2012.

[3] "World Population Prospects 2019: Highlights (ST/ESA/SER.A/423)," in *United Nations (UN), Population Division of the UN Department of Economic and Social Affairs*, New York, 2019.

[4] Shell International BV, "New Lens Scenarios: A Shift in Perspective for a World in Transition," 2013.

[5] ——, "Sky: Meeting the Goals of the Paris Agreement," in *SHELL SCENARIOS, 2016, Available Online [25/7/2020]: https://www.shell.com/energy-and-innovation/the-energy-future/scenarios/shell-scenario-sky.html*.

[6] International Energy Agency, "World Energy Outlook 2019," IEA (2019).

[7] World Energy Council, "World energy scenarios: Exploring innovation pathways to 2040," 2019.

[8] British Petroleum Energy Economics, "BP Energy Outlook: 2019 edition," Technical Report, 2019.

[9] British Petroleum, "Statistical Review of World Energy," Technical Report, 69th edition, 2020.

[10] "Adoption of the Paris Agreement," in *UNFCCC, 2015, Available Online [25/7/2020]: https://unfccc.int/process/conferences/pastconferences/paris-climate-change-conference-november-2015/paris-agreement*.

[11] "INDCs as communicated by Parties," in *UNFCCC, 2015, Available Online [25/7/2020]: https://www4.unfccc.int/sites/submissions/indc/Submission Pages/submissions.aspx*.

[12] J. Rogelj, M. Den Elzen, N. Höhne, T. Fransen, H. Fekete, H. Winkler, R. Schaeffer, F. Sha, K. Riahi, and M. Meinshausen, "Paris Agreement climate proposals need a boost to keep warming well below 2°C," *Nature*, vol. 534, no. 7609, p. 631, 2016.

[13] "Shell Energy Transition Report," in *SHELL REPORT, 2016, Available Online [25/7/2020]: https://www.shell.com/energy-and-innovation/the-energy-future/shell-energy-transition-report.html*.

[14] C. A. Fajardo, O. M. Reyes, and J. Castillo, "Reducing the I/O Bottleneck by a Compression Strategy," *Engineering Letters*, vol. 26, no. 2, 2018.

[15] A. A. Aqrawi and A. C. Elster, "Accelerating disk access using compression for large seismic datasets on modern GPU and CPU," *Scientific and Parallel Computing*, 2010.

[16] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1649–1668, 2012.

[17] K. Ugur, A. Alshin, E. Alshina, F. Bossen, W.-J. Han, J.-H. Park, and J. Lainema, "Motion compensated prediction and interpolation filter design in H.265/HEVC," *IEEE J. Sel. Top. Signa.*, vol. 7, pp. 946–956, 2013.

[18] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1792–1801, 2012.

[19] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards including high efficiency video coding (HEVC)," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1669–1684, 2012.

[20] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1685–1696, 2012.

[21] G. Correa, P. Assuncao, L. Agostini, and L. A. da Silva Cruz, "Performance and computational complexity assessment of high-efficiency video encoders," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1899–1909, 2012.

[22] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, G. Clare, F. Henry, S. Pateux, and T. Schierl, "Parallel scalability and efficiency of HEVC parallelization approaches," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1827–1838, 2012.

[23] D. Flynn, D. Marpe, M. Naccari, T. Nguyen, C. Rosewarne, K. Sharman, J. Sole, and J. Xu, "Overview of the range extensions for the HEVC standard: Tools, profiles, and performance," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 4–19, 2016.

[24] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1697–1706, 2012.

[25] M. Budagavi, A. Fuldseth, G. Bjontegaard, V. Sze, and M. Sadafale, "Core transform design in the high efficiency video coding (HEVC) standard," *IEEE J. Sel. Top. Signa.*, vol. 7, no. 6, pp. 1029–1041, 2013.

[26] V. Sze and M. Budagavi, "High throughput CABAC entropy coding in HEVC," *IEEE T. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1778–1791, 2012.

[27] L. C. Wood, "Seismic data compression methods," *Geophysics*, vol. 39, no. 4, pp. 499–525, 1974.

[28] A. S. Spanias, S. B. Jonsson, and S. D. Stearns, "Transform methods for seismic data compression," *IEEE Trans. Geosci. Remote.*, vol. 29, no. 3, pp. 407–416, 1991.

[29] J. D. Villasenor, R. Ergas, and P. Donoho, "Seismic data compression using high-dimensional wavelet transforms," in *In Proc. Data Compr. Conf.* IEEE, 1996, pp. 396–405.

[30] A. Vassiliou and V. Wickerhauser, "Comparison of wavelet image coding schemes for seismic data compression," in *SEG Technical Program Expanded Abstracts*, 1997, pp. 1334–1337.

[31] M. Khene and S. Abdul-Jauwad, "Adaptive seismic compression by wavelet shrinkage," in *In Proc. Workshop Stat. Signal and Array Process.* IEEE, 2000, pp. 544–548.

[32] C. Fajardo, O. M Reyes, and A. Ramirez, "Seismic data compression using 2D lifting-wavelet algorithms," *Ingeniería y Ciencia*, vol. 11, no. 21, pp. 221–238, 2015.

[33] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 247–269, 1998.

[34] A. Said, W. Pearlman, *et al.*, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circ. Syst. Vid.*, vol. 6, no. 3, pp. 243–250, 1996.

[35] K. Xie *et al.*, "Fast seismic data compression based on high-efficiency SPIHT," *Electron. Lett.*, vol. 50, no. 5, pp. 365–367, 2014.

[36] F. G. Meyer, A. Z. Averbuch, and J.-O. Stromberg, "Fast adaptive wavelet packet image compression," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 792–800, 2000.

[37] W. Wu, Z. Yang, Q. Qin, and F. Hu, "Adaptive seismic data compression using wavelet packets," in *Int. Geosci. Remote Se.*, 2006, pp. 787–789.

[38] F. G. Meyer, "Image compression with adaptive local cosines: A comparative study," *IEEE Trans. Image Process.*, vol. 11, no. 6, pp. 616–629, 2002.

[39] A. Z. Averbuch, V. A. Zheludev, M. Guttmann, and D. D. Kosloff, "LCT-wavelet based algorithms for data compression," *Int. J. Wavelets Multi.*, vol. 11, no. 05, p. 1350032, 2013.

[40] F. G. Meyer, A. Z. Averbuch, and R. R. Coifman, "Multilayered image representation: Application to image compression," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1072–1080, 2002.

[41] Y. Wang and R.-S. Wu, "Seismic data compression by an adaptive local cosine/sine transform and its effects on migration," *Geophys. Prospect.*, vol. 48, no. 6, pp. 1009–1031, 2000.

[42] L. C. Duval and T. Q. Nguyen, "Seismic data compression: a comparative study between GenLOT and wavelet compression," in *Wavelet Applications in Signal and Image Processing VII*, vol. 3813. Int. Soc. Optics and Photonics, 1999, pp. 802–810.

[43] R. L. De Queiroz, T. Q. Nguyen, and K. R. Rao, "The GenLOT: Generalized linear-phase lapped orthogonal transform," *IEEE Trans. Sign. Process.*, vol. 44, no. 3, pp. 497–507, 1996.

[44] L. C. Duval and T. Nagai, "Seismic data compression using GULLOTS," in *In Proc. ICASSP*, vol. 3. IEEE, 2001, pp. 1765–1768.

[45] T. Nagai, M. Ikehara, M. Kaneko, and A. Kurematsu, "Generalized unequal length lapped orthogonal transform for subband image coding," *IEEE Trans. Sign. Process.*, vol. 48, no. 12, pp. 3365–3378, 2000.

[46] J. M. Lervik, T. Rosten, and T. A. Ramstad, "Subband seismic data compression: optimization and evaluation," in *In Proc. Dig. Sign. Process. Workshop*. IEEE, 1996, pp. 65–68.

[47] L. C. Duval and T. Røsten, "Filter bank decomposition of seismic data with application to compression and denoising," in *SEG Technical Program Expanded Abstracts*, 2000, pp. 2055–2058.

[48] T. Røsten, T. A. Ramstad, and L. Amundsen, "Optimization of sub-band coding method for seismic data compression," *Geophysical Prospecting*, vol. 52, no. 5, pp. 359–378, 2004.

[49] B. Liu *et al.*, "A distributed principal component analysis compression for smart seismic acquisition networks," *IEEE Trans. Geosci. Remote*, vol. 56, no. 6, pp. 3020–3029, 2018.

[50] F. Zheng, Y. Ling, Y. Tang, S. Hui, and H. Yang, "A fidelity-restricted distributed principal component analysis compression algorithm for non-cable seismographs," *J. Appl. Geophys.*, 2019.

[51] J. Hu, S. Fomel, *et al.*, "A fast butterfly algorithm for generalized Radon transforms," *Geophysics*, vol. 78, no. 4, pp. U41–U51, 2013.

[52] S. Fomel, "Seismic data decomposition into spectral components using regularized nonstationary autoregression," *Geophysics*, vol. 78, no. 6, pp. 69–76, 2013.

[53] J. Ma, G. Plonka, and H. Chauris, "A new sparse representation of seismic data using adaptive easy-path wavelet transform," *IEEE Trans. Geosci. Remote*, vol. 7, no. 3, pp. 540–544, 2010.

[54] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, 2005.

[55] S. Fomel, "Towards the seislet transform," in *SEG Technical Program Expanded Abstracts 2006*. SEG, 2006, pp. 2847–2851.

[56] S. Fomel and Y. Liu, "Seislet transform and seislet frame," *Geophysics*, vol. 75, no. 3, pp. V25–V38, 2010.

[57] F. G. Meyer and R. R. Coifman, "Brushlets: a tool for directional image analysis and image compression," *Appl. Comput. Harmon. A.*, vol. 4, no. 2, pp. 147–187, 1997.

[58] Y. Geng, R.-S. Wu, and J. Gao, "Dreamlet transform applied to seismic data compression and its effects on migration," in *SEG Technical Program Expanded Abstracts 2009*. SEG, 2009, pp. 3640–3644.

[59] E. J. Candès and D. L. Donoho, "New tight frames of curvelets and optimal representations of objects with piecewise c2 singularities," *Commun. Pur. Appl. Math.*, vol. 57, no. 2, pp. 219–266, 2004.

[60] E. Candes, L. Demanet, D. Donoho, and L. Ying, "Fast discrete curvelet transforms," *Multiscale Model. Sim.*, vol. 5, no. 3, pp. 861–899, 2006.

[61] J. Ma and G. Plonka, "The curvelet transform," *IEEE Signal Proc. Mag.*, vol. 27, no. 2, pp. 118–133, 2010.

[62] S. Wang, J. Li, S. K. Chiu, and P. D. Anno, "Seismic data compression and regularization via wave packets," in *SEG Technical Program Expanded Abstracts 2010*. SEG, 2010, pp. 3650–3655.

[63] A. A. Duchkov, F. Andersson, and V. Maarten, "Discrete almost-symmetric wave packets and multiscale geometrical representation of (seismic) waves," *IEEE Trans. Geosci. Remote.*, vol. 48, no. 9, pp. 3408–3423, 2010.

[64] M. N. Do and M. Vetterli, "Contourlets: a directional multiresolution image representation," in *In Proc. Int. Conf. Image Process. (ICIP)*, vol. 1. IEEE, 2002, pp. I–I.

[65] ——, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2091–2106, 2005.

[66] Y. Lu and M. N. Do, "CRISP contourlets: a critically sampled directional multiresolution image representation," in *Wavelets: Applications in Signal and Image Processing X*, vol. 5207. International Society for Optics and Photonics, 2003, pp. 655–665.

[67] M. Xu, S. Li, *et al.*, "Compressibility constrained sparse representation with learnt dictionary for low bit-rate image compression," *IEEE Trans. Circ. Syst. Vid.*, vol. 24, no. 10, pp. 1743–1757, 2014.

[68] X. Zhang, W. Lin, Y. Zhang, S. Wang, S. Ma, L. Duan, and W. Gao, "Rate-distortion optimized sparse coding with ordered dictionary for image set compression," *IEEE Trans. Circ. Syst. Vid.*, vol. 28, no. 12, pp. 3387–3397, 2017.

[69] A. Payani, A. Abdi, X. Tian, F. Fekri, and M. Mohandes, "Advances in Seismic Data Compression via Learning from Data: Compression for Seismic Data Acquisition," *IEEE Signal Proc. Mag.*, vol. 35, no. 2, pp. 51–61, 2018.

[70] X. Tian, A. Abdi, E. Liu, and F. Fekri, "Memory-assisted seismic signal compression based on dictionary learning and sparse coding," in *In Proc. GlobalSIP*. IEEE, 2017, pp. 358–362.

[71] X. Tian and S. Li, "Seismic Signal Compression Using Nonparametric Bayesian Dictionary Learning via Clustering," *Algorithms*, vol. 10, no. 2, p. 65, 2017.

[72] X. Tian, A. Abdi, E. Liu, and F. Fekri, "Seismic Signal Compression Through Delay Compensated and Entropy Constrained Dictionary Learning," in *Int. Work. Sign. Proc. (SPAWC)*. IEEE, 2018, pp. 1–5.

[73] X. Tian, "Multiscale sparse dictionary learning with rate constraint for seismic data compression," *IEEE Access*, pp. 86 651–86 663, 2019.

[74] A. Abdi, A. Payani, and F. Fekri, "Learning dictionary for efficient signal compression," in *Int. Conf. Acoust. Spee. (ICASSP)*. IEEE, 2017, pp. 3689–3693.

[75] E. Liu, A. Payani, and F. Fekri, "Seismic data compression using online double-sparse dictionary learning schemes," in *2017 Data Compression Conference (DCC)*. IEEE, 2017, pp. 449–449.

[76] N. Allegar, F. J. Herrmann, and C. C. Mosher, "Introduction to this special section: Impact of compressive sensing on seismic data acquisition and processing," *The Leading Edge*, vol. 36, no. 8, pp. 640–640, 2017.

[77] D. L. Donoho *et al.*, "Compressed sensing," *IEEE Trans. Informa. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[78] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Informa. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[79] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Proc. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[80] F. J. Herrmann and G. Hennenfent, "Non-parametric seismic data recovery with curvelet frames," *Geophys. J. Int.*, vol. 173, no. 1, pp. 233–248, 2008.

[81] R.-S. Wu, Y. Geng, and L. Ye, "Preliminary study on dreamlet based compressive sensing data recovery," in *SEG Technical Program Expanded Abstracts 2013*. SEG, 2013, pp. 3585–3590.

[82] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pur. Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.

[83] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Top. Signa.*, vol. 1, no. 4, pp. 586–597, 2007.

[84] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted $\ell 1$ minimization," *J. Fourier Anal. Appl.*, vol. 14, no. 5-6, pp. 877–905, 2008.

[85] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. Inform. Theory*, vol. 58, no. 2, pp. 1094–1121, 2012.

[86] G. Hennenfent and F. J. Herrmann, "Simply denoise: Wavefield reconstruction via jittered undersampling," *Geophysics*, vol. 73, no. 3, pp. V19–V28, 2008.

[87] F. J. Herrmann, "Randomized sampling and sparsity: Getting more information from fewer samples," *Geophysics*, vol. 75, no. 6, pp. WB173–WB187, 2010.

[88] J. Cao, Y. Wang, J. Zhao, and C. Yang, "A review on restoration of seismic wavefields based on regularization and compressive sensing," *Inverse Probl. Sci. En.*, vol. 19, no. 5, pp. 679–704, 2011.

[89] M. J. Rubin, M. B. Wakin, and T. Camp, "Lossy compression for wireless seismic data acquisition," *IEEE J. Sel. Top. Appl.*, vol. 9, no. 1, pp. 236–252, 2015.

[90] A. Latif and W. A. Mousa, "An efficient undersampled high-resolution radon transform for exploration seismic data processing," *IEEE Trans. Geosci. Remote.*, vol. 55, no. 2, pp. 1010–1024, 2016.

[91] C. C. Mosher, C. Li, F. D. Janiszewski, L. S. Williams, T. C. Carey, and Y. Ji, "Operational deployment of compressive sensing systems for seismic data acquisition," *The Leading Edge*, vol. 36, no. 8, pp. 661–669, 2017.

[92] F. Oghenekohwo, H. Wason, E. Esser, and F. J. Herrmann, "Low-cost time-lapse seismic with distributed compressive sensing—part 1: Exploiting common information among the vintages," *Geophysics*, vol. 82, no. 3, pp. P1–P13, 2017.

[93] H. Wang, C. Tao, S. Chen, Z. Wu, Y. Du, J. Zhou, L. Qiu, H. Shen, W. Xu, and Y. Liu, "High-precision seismic data reconstruction with multi-domain sparsity constraints based on curvelet and high-resolution radon transforms," *J. Appl. Geophys.*, vol. 162, pp. 128–137, 2019.

[94] F. J. Herrmann, M. P. Friedlander, and O. Yilmaz, "Fighting the curse of dimensionality: Compressive sensing in exploration seismology," *IEEE Signal Proc. Mag.*, vol. 29, no. 3, pp. 88–100, 2012.

[95] R. G. Baraniuk and P. Steeghs, "Compressive sensing: A new approach to seismic data acquisition," *The Leading Edge*, vol. 36, no. 8, pp. 642–645, 2017.

[96] H. Nuha, M. Mohandes, and B. Liu, "Seismic data compression using auto-associative neural network and restricted boltzmann machine," in *SEG Technical Program Expanded Abstracts*, 2018, pp. 186–190.

[97] A. P. Schiavon, J. P. Navarro, M. B. Vieira, and P. M. C. e Silva, "Low bit rate 2d seismic image compression with deep autoencoders," in *Int. Conf. Comput. Sc. and Its App.* Springer, 2019, pp. 397–407.

[98] H. H. Nuha, A. Balghonaim, B. Liu, M. Mohandes, M. Deriche, and F. Fekri, "Deep Neural Networks with Extreme Learning Machine for Seismic Data Compression," *Arab. J. Sci. Eng.*, pp. 1–11, 2019.

[99] X. Zhang, S. Zhang, J. Lin, *et al.*, "An Efficient Seismic Data Acquisition Based on Compressed Sensing Architecture With Generative Adversarial Networks," *IEEE Access*, vol. 7, pp. 105 948–105 961, 2019.

[100] P. Lu, Y. Xiao, Y. Zhang, and N. Mitsakos, "Deep learning for 3D seismic compressive-sensing technique: A novel approach," *The Leading Edge*, vol. 38, no. 9, pp. 698–705, 2019.

[101] H. H. Nuha, A. Balghonaim, B. Liu, M. Mohandes, and F. Fekri, "Seismic data compression using deep neural network predictors," in *SEG Technical Program Expanded Abstracts*, 2019, pp. 258–262.

[102] A. Payani, F. Fekri, G. Alregib, M. Mohandes, and M. Deriche, "Compression of seismic signals via recurrent neural networks: Lossy and lossless algorithms," in *SEG Technical Program Expanded Abstracts*, 2019, pp. 4082–4086.

[103] S. Savazzi, U. Spagnolini, L. Goratti, D. Molteni, M. Latva-aho, and M. Nicoli, "Ultra-wide band sensor networks in oil and gas explorations," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 150–160, 2013.

[104] S. Savazzi and U. Spagnolini, "Compression and coding for cable-free land acquisition systems," *Geophysics*, vol. 76, no. 5, pp. Q29–Q39, 2011.

[105] B. Liu, M. Mohandes, H. Nuha, M. Deriche, N. Iqbal, F. Fekri, and J. H. McClellan, "A multi tone modeling for seismic data compression," in *SEG Technical Program Expanded Abstracts*, 2019, pp. 263–267.

[106] S. A. Zummo *et al.*, "Functional Quantization-Based Data Compression in Seismic Acquisition," *Arab. J. Sci. Eng.*, vol. 44, no. 3, pp. 2151–2163, 2019.

[107] D. Taubman and M. Marcellin, "Image compression fundamentals, standards and practice," *JPEG-2000*, 2001.

[108] Y. Liu, Z. Xiong, L. Lu, and D. Hohl, "Fast SNR and rate control for JPEG XR," in *Signal Proc. and Comm. Syst. (ICSPCS)*, 2016, pp. 1–7.

[109] "Information Technology-JPEG 2000 Image Coding System: Part 10-Extensions for three-dimensional data," *ISO/IEC JTC1/SC29/WG1 FDIS*, 2007.

[110] "Cisco Visual Networking Index: Forecast and Trends, 2017-2022," *White paper, Cisco public*, Nov. 2018.

[111] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall, and A. Vetro, "Standardized extensions of high efficiency video coding (HEVC)," *IEEE J. Sel. Top. Signa.*, vol. 7, no. 6, pp. 1001–1016, 2013.

[112] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 20–34, 2016.

[113] G. Tech, Y. Chen, K. Müller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 35–49, 2016.

[114] J. Xu, R. Joshi, and R. A. Cohen, "Overview of the emerging HEVC screen content coding extension," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 50–62, 2016.

[115] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 13, no. 7, pp. 560–576, 2003.

[116] T. Nguyen and D. Marpe, "Objective performance evaluation of the HEVC main still picture profile," *IEEE Trans. Circ. Syst. Vid.*, vol. 25, no. 5, pp. 790–797, 2014.

[117] M. Radosavljević, B. Brkljač, P. Lugonja, V. Crnojević, Ž. Trpovski, Z. Xiong, and D. Vukobratović, "Lossy Compression of Multispectral Satellite Images with Application to Crop Thematic Mapping: A HEVC Comparative Study," *Remote Sensing*, vol. 12, no. 10, p. 1590, 2020.

[118] M. Radosavljević, M. Adamović, B. Brkljač, Ž. Trpovski, Z. Xiong, and D. Vukobratović, "Satellite image compression based on High Efficency Video Coding standard - An experimental comparison with JPEG 2000," in *In Proc. Int. Conf. on Big Data from Space*. ESA and DLR, Feb. 2019, pp. 257–260.

[119] R. Monteiro, L. Lucas, C. Conti, P. Nunes, N. Rodrigues, S. Faria, C. Pagliari, E. da Silva, and L. Soares, "Light field HEVC-based image coding using locally linear embedding and self-similarity compensated prediction," in *IEEE Int. Conf. on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2016, pp. 1–4.

[120] X. Jin, H. Han, and Q. Dai, "Plenoptic Image Coding Using Macropixel-Based Intra Prediction," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3954–3968, 2018.

[121] U. Pestel-Schiller and J. Ostermann, "Subjective Evaluation of Compressed SAR Images using JPEG and HEVC intra coding: Sometimes, Compression Improves Usability," in *15th European Radar Conference (EuRAD)*. IEEE, 2018, pp. 154–157.

[122] R. Dusselaar, M. Paul, and T. Bossomaier, "Hyperspectral image coding using spectral prediction modelling in HEVC coding framework," in *International Conference on Image and Vision Computing New Zealand (IVCNZ)*. IEEE, 2015, pp. 1–6.

[123] F. Gao, X. Ji, C. Yan, and Q. Dai, "Compression of multispectral image using HEVC," in *Optoelectronic Imaging and Multimedia Technology III*, vol. 9273. International Society for Optics and Photonics, 2014, p. 92732X.

[124] A. F. Guarda, J. M. Santos, L. A. da Silva Cruz, P. A. Assunção, N. M. Rodrigues, and S. M. de Faria, "A method to improve HEVC lossless coding of volumetric medical images," *Signal Processing: Image Communication*, vol. 59, pp. 96–104, 2017.

[125] V. Sanchez, F. A. Llinas, J. B. Rapesta, and J. S. Sagrista, "Improvements to HEVC intra coding for lossless medical image compression," in *Data Compression Conference (DCC), 2014*. IEEE, 2014, pp. 423–423.

[126] V. Sanchez and J. Bartrina-Rapesta, "Lossless compression of medical images based on HEVC intra coding," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 6622–6626.

[127] S. S. Parikh, D. Ruiz, H. Kalva, G. Fernández-Escribano, and V. Adzic, "High bit-depth medical image compression with hevc," *IEEE journal of biomedical and health informatics*, vol. 22, no. 2, pp. 552–560, 2018.

[128] W. Zhu, W. Ding, J. Xu, Y. Shi, and B. Yin, "Hash-based block matching for screen content coding," *IEEE Trans. Multimedia*, vol. 17, no. 7, pp. 935–944, 2015.

[129] Z. Ma, W. Wang, M. Xu, and H. Yu, "Advanced screen content coding using color table and index map," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4399–4412, 2014.

[130] C.-C. Chen and W.-H. Peng, "Intra line copy for HEVC screen content intra-picture prediction," *IEEE Trans. Circ. Syst. Vid.*, vol. 27, no. 7, pp. 1568–1579, 2017.

[131] W. Pu, M. Karczewicz, R. Joshi, V. Seregin, F. Zou, J. Sole, Y.-C. Sun, T.-D. Chuang, P. Lai, S. Liu, *et al.*, "Palette mode coding in HEVC screen content coding extension," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 6, no. 4, pp. 420–432, 2016.

[132] W. Zhu, W. Ding, J. Xu, Y. Shi, and B. Yin, "Screen content coding based on HEVC framework," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1316–1326, 2014.

[133] S. Wang, X. Zhang, X. Liu, J. Zhang, S. Ma, and W. Gao, "Utility-driven adaptive preprocessing for screen content video compression," *IEEE Trans. Multimedia*, vol. 19, no. 3, pp. 660–667, 2017.

[134] M. Radosavljević, Z. Xiong, L. Lu, D. Hohl, and D. Vukobratović, "High Bit-Depth Seismic Data Compression: A Novel Codec Under the Framework of HEVC," *IEEE Access*, vol. 8, pp. 114 443–114 459, 2020.

[135] M. Radosavljević, Z. Xiong, L. Lu, and D. Vukobratović, "High bit-depth image compression with application to seismic data," in *Visual Comm. and Image Processing (VCIP), 2016*. IEEE, 2016, pp. 1–4.

[136] M. Radosavljević, Z. Xiong, L. Lu, D. Hohl, and D. Vukobratović, "HEVC-based compression of high bit-depth 3D seismic data," in *Image Processing (ICIP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4028–4032.

[137] M. Chan, Y. Yu, and A. Constantinides, "Variable size block matching motion compensation with applications to video coding," *IEEE Proc. Commun., speech, and vision*, vol. 137, no. 4, pp. 205–212, 1990.

[138] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 327–331, 1994.

[139] J. Zhang, M. O. Ahmad, and M. Swamy, "Quadtree structured region-wise motion compensation for video compression," *IEEE Trans. Circ. Syst. Vid.*, vol. 9, no. 5, pp. 808–822, 1999.

[140] D. Marpe, H. Schwarz, S. Bosse, B. Bross, P. Helle, T. Hinz, H. Kirchhoffer, H. Lakshman, T. Nguyen, S. Oudin, *et al.*, "Video compression using nested quadtree structures, leaf merging, and improved techniques for motion representation and entropy coding," *IEEE Trans. Circ. Syst. Vid.*, vol. 20, no. 12, pp. 1676–1687, 2010.

[141] W.-J. Han, J. Min, I.-K. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. M. Hong, *et al.*, "Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools," *IEEE Trans. Circ. Syst. Vid.*, vol. 20, no. 12, pp. 1709–1720, 2010.

[142] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465–470, 2013.

[143] S. Cho and M. Kim, "Fast CU splitting and pruning for suboptimal CU partitioning in HEVC intra coding," *IEEE Trans. Circ. Syst. Vid.*, vol. 23, no. 9, pp. 1555–1564, 2013.

[144] H.-S. Kim and R.-H. Park, "Fast CU partitioning algorithm for HEVC using an online-learning-based Bayesian decision rule," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 130–138, 2016.

[145] Y. Yuan, I.-K. Kim, X. Zheng, L. Liu, X. Cao, S. Lee, M.-S. Cheon, T. Lee, Y. He, and J.-H. Park, "Quadtree based nonsquare block structure for inter frame coding in high efficiency video coding," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1707–1719, 2012.

[146] A. Minezawa, K. Sugimoto, and S. Sekiguchi, "An improved intra vertical and horizontal prediction," in *Document F172, 6th JCT-VC Meeting, Torino, IT*, July 2011.

[147] S. Kanumuri, T. Tan, and F. Bossen, "Enhancements to Intra Coding," in *Document D235, 4th JCT-VC Meeting, Daegu, KR*, Jan. 2011.

[148] G. Van Wallendael, S. Van Leuven, J. De Cock, P. Lambert, R. Van de Walle, J. Barbarien, and A. Munteanu, "Improved intra mode signaling for HEVC," in *IEEE Int. Conf. Multimedia and Expo (ICME)*. IEEE, 2011, pp. 1–6.

[149] HEVC reference software HM. Accessed: 25/7/2020. [Online]. Available: https://vcgit.hhi.fraunhofer.de/jct-vc/HM

[150] Y. Piao, J. Min, and J. Chen, "Encoder improvement of unified intra prediction," in *Doc. C207, JCT-VC Meeting, Guangzhou, CN*, Oct. 2010.

[151] T. L. Da Silva, L. V. Agostini, and L. A. da Silva Cruz, "Fast HEVC intra prediction mode decision based on EDGE direction information," in *20th Proc. European Sign. Proc. Conf. (EUSIPCO)*. IEEE, 2012, pp. 1214–1218.

[152] W. Jiang, H. Ma, and Y. Chen, "Gradient based fast mode decision algorithm for intra prediction in HEVC," in *2nd Int. Conf. Consum. Electron., Comm. and Net. (CECNet)*. IEEE, 2012, pp. 1836–1840.

[153] L. Shen, Z. Zhang, and P. An, "Fast CU size decision and mode decision algorithm for HEVC intra coding," *IEEE Trans. Consum. Electron.*, vol. 59, no. 1, pp. 207–213, 2013.

[154] K. McCann, W. Han, I. Kim, J. Min, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. Hong, M. Cheon, and N. Shlyakhov, "Samsung's response to the call far proposals on video compression technology," in *Document A124, 1st JCT-VC Meeting, Dresden, DE*, April 2010.

[155] T. Sugio and T. Nishi, "Parsing robustness for merge/AMVP," in *Document F470, 6th JCT-VC Meeting, Torino, IT*, July 2011.

[156] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Trans. Comm.*, vol. 41, no. 4, pp. 604–612, 1993.

[157] N. Purnachand, L. N. Alves, and A. Navarro, "Improvements to TZ search motion estimation algorithm for multiview video coding," in *Int. Conf. Sys., Sign. and Image Proc. (IWSSIP)*. IEEE, 2012, pp. 388–391.

[158] P. Helle, S. Oudin, B. Bross, D. Marpe, M. O. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for quadtree-based partitioning in hevc," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, pp. 1720–1731, 2012.

[159] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Computers*, vol. 100, no. 1, pp. 90–93, 1974.

[160] A. Fuldseth, G. Bjontegaard, M. Budagavi, and V. Sze, "Core transform design for HEVC: proposal for current HEVC transform," in *Document G495, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

[161] M. Tikekar, C.-T. Huang, C. Juvekar, and A. Chandrakasan, "Core Transform Property for Practical Throughput Hardware Design," *Document G265, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

[162] W.-H. Chen, C. Smith, and S. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. Commun.*, vol. 25, no. 9, pp. 1004–1009, 1977.

[163] J. Liang and T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme," *IEEE Trans. Sig. Proces.*, vol. 49, no. 12, pp. 3032–3044, 2001.

[164] Source code: BinDCT. Accessed: 25/7/2020. [Online]. Available: https://github.com/Sh0lim/BinDCT

[165] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H. 264/AVC video compression standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 13, no. 7, pp. 620–636, 2003.

[166] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[167] J. Rissanen and G. G. Langdon, "Arithmetic coding," *IBM J. Res. Dev.*, vol. 23, no. 2, pp. 149–162, 1979.

[168] J. Rissanen and G. Langdon, "Universal modeling and coding," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 12–23, 1981.

[169] A. Moffat, R. M. Neal, and I. H. Witten, "Arithmetic coding revisited," *ACM Trans. Inf. Sys. (TOIS)*, vol. 16, no. 3, pp. 256–294, 1998.

[170] J. Sole, R. Joshi, N. Nguyen, T. Ji, M. Karczewicz, G. Clare, F. Henry, and A. Duenas, "Transform coefficient coding in HEVC," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1765–1777, 2012.

[171] T. Nguyen, P. Helle, M. Winken, B. Bross, D. Marpe, H. Schwarz, and T. Wiegand, "Transform coding techniques in HEVC," *IEEE J. Sel. Top. Signa.*, vol. 7, no. 6, pp. 978–989, 2013.

[172] S. Golomb, "Run-length encodings (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 12, no. 3, pp. 399–401, 1966.

[173] R. Rice and J. Plaunt, "Adaptive variable-length coding for efficient compression of spacecraft television data," *IEEE Trans. Comm.*, vol. 19, no. 6, pp. 889–897, 1971.

[174] R. Gallager and D. Van Voorhis, "Optimal source codes for geometrically distributed integer alphabets (corresp.)," *IEEE Trans. Inf. Theory*, vol. 21, no. 2, pp. 228–230, 1975.

[175] J. Teuhola, "A compression method for clustered bit-vectors," *Inf. Process. Lett.*, vol. 7, no. 6, pp. 308–311, 1978.

[176] T. Nguyen, D. Marpe, H. Schwarz, and T. Wiegand, "Reduced-complexity entropy coding of transform coefficient levels using truncated Golomb-Rice codes in video compression," in *18th IEEE Int. Conf. on Image Processing (ICIP)*. IEEE, 2011, pp. 753–756.

[177] J. Sole, R. Joshi, and M. Karczewicz, "Non-CE11: Diagonal sub-block scan for HE residual coding," in *Document G323, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

[178] Y. Zheng, M. Coban, X. Wang, J. Sole, R. Joshi, and M. Karczewicz, "CE11: Mode Dependent Coefficient Scanning," in *Document D393, 4th JCT-VC Meeting, Daegu, KR*, Jan. 2011.

[179] N. Nguyen, T. Ji, D. He, G. Martin-Cocher, and L. Song, "Multi-level Significant Maps for Large Transform Units," in *Document G644, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

[180] J. Chen, W.-J. Chien, R. Joshi, J. Sole, and M. Karczewicz, "Non-CE1: throughput improvement on CABAC coefficients level coding," in *Document H554, 8th JCT-VC Meeting, San Jose, USA*, Feb. 2012.

[181] C. Kim, J. Kim, and J. Park, "Simplification of Golomb-Rice Parameter Update," in *Document I0124, 9th JCT-VC Meeting, Geneva, CH*, Apr. 2012.

[182] W.-J. Chien, M. Karczewicz, J. Sole, and J. Chen, "On coefficient level remaining coding," in *Document I0487, 9th JCT-VC Meeting, Geneva, CH*, Apr. 2012.

[183] M. Budagavi and V. Sze, "coeff_abs_level_remaining maximum codeword length reduction," in *Document J0142, 10th JCT-VC Meeting, Stockholm, SWE*, Jul. 2012.

[184] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Int. Conf. on Image Proces.*, vol. 3, 2001, pp. 542–545.

[185] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Proc. Mag.*, vol. 15, no. 6, pp. 74–90, 1998.

[186] H. Everett III, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operations research*, vol. 11, no. 3, pp. 399–417, 1963.

[187] G. Clare, F. Henry, and J. Jung, "Sign Data Hiding," in *Document G271, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

[188] X. Yu, J. Wang, D. He, G. Martin-Cocher, and S. Campbell, "Multiple Sign Bits Hiding," in *Document H0481, 8th JCT-VC Meeting, San Jose, USA*, Feb. 2012.

[189] X. Yu, J. Wang, D. He, and G. Martin-Cocher, "Simplification of Multiple Sign Bit Hiding," in *Document I0156, 9th JCT-VC Meeting, Geneva, CH*, Apr. 2012.

[190] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera, "HEVC deblocking filter," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1746–1754, 2012.

[191] C.-M. Fu, E. Alshina, A. Alshin, Y.-W. Huang, C.-Y. Chen, C.-Y. Tsai, C.-W. Hsu, S.-M. Lei, J.-H. Park, and W.-J. Han, "Sample adaptive offset in the HEVC standard," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1755–1764, 2012.

[192] C. Keiichi, A. Hirofumi, W. Viktor, and L. ChongSoon, "Proposal of enhanced PCM coding in HEVC," in *Document E192, 5th JCT-VC Meeting, Geneva, CH*, March 2011.

[193] P. Xiulian, X. Jizheng, L. Bin, G. Liwei, S. Joel, and M. Karczewicz, "Non-RCE2: Transform skip on large TUs," in *Document N0288, 14th JCT-VC Meeting, Vienna, AT*, July 2013.

[194] J. Xu, R. Joshi, and R. A. Cohen, "Overview of the emerging HEVC screen content coding extension," *IEEE Trans. Circ. Syst. Vid.*, vol. 26, no. 1, pp. 50–62, 2015.

[195] L. Cuiling, X. Jizheng, G. J. Sullivan, and W. Feng, "Intra transform skipping," in *Document I4048, 9th JCT-VC Meeting, Geneva, CH*, April 2012.

[196] H. Dake, W. Jing, and M.-C. Gaelle, "Rotation of Residual Block for Transform Skipping ," in *Document J0093, 10th JCT-VC Meeting, Stockholm, SE*, July 2012.

[197] P. Xiulian, L. Bin, and X. Jizheng, "On residual rotation for Inter and Intra BC modes," in *Document O0186, 15th JCT-VC Meeting, Geneva, CH*, Oct. 2013.

[198] J. Rajan, S. Joel, and K. Marta, "AHG8: Residual DPCM for visually lossless coding," in *Document M0351, 13th JCT-VC Meeting, Incheon, KR*, Apr. 2013.

[199] N. Matteo, G. Andrea, M. Marta, B. Saverio, and I. Ebroul, "Inter-Prediction Residual DPCM," in *Document M0442, 13th JCT-VC Meeting, Incheon, KR*, Apr. 2013.

[200] G. Van der Auwera, X. Wang, and M. Karczewicz, "CE6.e: Mode-Dependent Intra Smoothing Modifications," in *Document F126, 6th JCT-VC Meeting, Torino, IT*, July 2011.

[201] X. Jun, T. Ali, N. Ohji, and S. Teruhiko, "Fix for Strong Intra Smoothing in RExt," in *Document Q0128, 17th JCT-VC Meeting, Valencia, ES*, March 2014.

[202] G. Bjontegarrd, "Calculation of average PSNR differences between RD-curves," *VCEG-M33*, 2001.

[203] G. Bjontegaard, "Improvements of the BD-PSNR model, VCEG-AI11," in *ITU-T Q. 6/SG16, 34th VCEG Meeting, Berlin, Germany, July*, 2008.

[204] T. Wiegand, Z. Xiaozheng, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Trans. Circ. Syst. Vid.*, vol. 9, no. 1, pp. 70–84, 1999.

[205] H. Li, B. Li, and J. Xu, "Rate-distortion Optimized Reference Picture Management for High Efficiency Video Coding," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1844–1857, 2012.

[206] R. Sjoberg, Y. Chen, A. Fujibayashi, M. M. Hannuksela, J. Samuelsson, T. K. Tan, Y.-K. Wang, and S. Wenger, "Overview of HEVC high-level syntax and reference picture management," *IEEE Trans. Circ. Syst. Vid.*, vol. 22, no. 12, pp. 1858–1870, 2012.

[207] S. Rickard and S. Jonatan, "Absolute signaling of reference pictures," in *Document F493, 6th JCT-VC Meeting, Torino, IT*, Jul 2011.

[208] T. K. Tan and C. S. Boon, "Inter reference picture set prediction syntax and semantics." in *Document G198, 7th JCT-VC Meeting, Geneva, CH*, Nov. 2011.

Овај Образац чини саставни део докторске дисертације, односно докторског уметничког пројекта који се брани на Универзитету у Новом Саду. Попуњен Образац укоричити иза текста докторске дисертације, односно докторског уметничког пројекта.

План третмана података

Назив пројекта/истраживања
Назив институције/институција у оквиру којих се спроводи истраживање
а) Факултет техничких наука, Нови Сад б) в)
Назив програма у оквиру ког се реализује истраживање
Енергетика, електроника и телекомуникације
1. Опис података
1.1 Врста студије *Укратко описати тип студије у оквиру које се подаци прикупљају* _____Истраживање на пољу компресије сеизмичких података_____ _____ _____ 1.2 Врсте података а) квантитативни б) квалитативни 1.3. Начин прикупљања података а) анкете, упитници, тестови

б) клиничке процене, медицински записи, електронски здравствени записи

в) генотипови: навести врсту _____

г) административни подаци: навести врсту _____

д) узорци ткива: навести врсту_____

ђ) снимци, фотографије: сеизмичке слике _____

е) текст, навести врсту _____

ж) мапа, навести врсту _____

з) остало: описати _____

1.3 Формат података, употребљене скале, количина података

Сирове сеизмичке слике

1.3.1 Употребљени софтвер и формат датотеке:

a) Excel фајл, датотека _____

b) SPSS фајл, датотека _____

c) PDF фајл, датотека _____

d) Текст фајл, датотека _____

e) JPG фајл, датотека _____

f) Остало, датотека _____

1.3.2. Број записа (код квантитативних података)

а) број варијабли _____

б) број мерења (испитаника, процена, снимака и сл.) _____

1.3.3. Поновљена мерења

а) да

б) не

Уколико је одговор да, одговорити на следећа питања:

а) временски размак између поновљених мера је _____

б) варијабле које се више пута мере односе се на _____

в) нове верзије фајлова који садрже поновљена мерења су именоване као _____

Напомене: _____

Да ли формати и софтвер омогућавају дељење и дугорочну валидност података?

а) Да

б) Не

Ако је одговор не, образложити _____

_____

2. Прикупљање података

2.1 Методологија за прикупљање/генерисање података

2.1.1. У оквиру ког истраживачког нацрта су подаци прикупљени?

а) експеримент, навести тип _____

б) корелационо истраживање, навести тип _____

ц) анализа текста, навести тип _____

д) остало, навести шта _____

2.1.2 Навести врсте мерних инструмената или стандарде података специфичних за одређену научну дисциплину (ако постоје).

_____

_____

2.2 Квалитет података и стандарди

2.2.1. Третман недостајућих података

а) Да ли матрица садржи недостајуће податке? Да Не

Ако је одговор да, одговорити на следећа питања:

а) Колики је број недостајућих података? _____

б) Да ли се кориснику матрице препоручује замена недостајућих података? Да Не

в) Ако је одговор да, навести сугестије за третман замене недостајућих података

_____

2.2.2. На који начин је контролисан квалитет података? Описати

_____

_____

2.2.3. На који начин је извршена контрола уноса података у матрицу?

_____

_____

3. Третман података и пратећа документација

3.1. Третман и чување података

3.1.1. Подаци ће бити депоновани у _____ репозиторијум.

3.1.2. URL адреса _____

3.1.3. DOI _____

3.1.4. Да ли ће подаци бити у отвореном приступу?

а) Да

б) Да, али после ембарга који ће трајати до _____

в) Не

Ако је одговор не, навести разлог __Аутор нема право да дели сеизмичке слике које су коришћене у овом раду _____

3.1.5. Подаци неће бити депоновани у репозиторијум, али ће бити чувани.

Образложење

_____

_____

3.2 Метаподаци и документација података

3.2.1. Који стандард за метаподатке ће бити примењен? _____

3.2.1. Навести метаподатке на основу којих су подаци депоновани у репозиторијум.

_____

_____

Ако је потребно, навести методе које се користе за преузимање података, аналитичке и процедуралне информације, њихово кодирање, детаљне описе варијабли, записа итд.

_____

_____

_____

_____

3.3 Стратегија и стандарди за чување података

3.3.1. До ког периода ће подаци бити чувани у репозиторијуму? _____

3.3.2. Да ли ће подаци бити депоновани под шифром? Да Не

3.3.3. Да ли ће шифра бити доступна одређеном кругу истраживача? Да Не

3.3.4. Да ли се подаци морају уклонити из отвореног приступа после извесног времена?

Да Не

Образложити

_____

_____

4. Безбедност података и заштита поверљивих информација

Овај одељак МОРА бити попуњен ако ваши подаци укључују личне податке који се односе на

учеснике у истраживању. За друга истраживања треба такође размотрити заштиту и сигурност података.

4.1 Формални стандарди за сигурност информација/података

Истраживачи који спроводе испитивања с људима морају да се придржавају Закона о заштити података о личности (*https://www.paragraf.rs/propisi/zakon_o_zastiti_podataka_o_licnosti.html*) и одговарајућег институционалног кодекса о академском интегритету.

4.1.2. Да ли је истраживање одобрено од стране етичке комисије? Да Не

Ако је одговор Да, навести датум и назив етичке комисије која је одобрила истраживање

_____

4.1.2. Да ли подаци укључују личне податке учесника у истраживању? Да (Не)

Ако је одговор да, наведите на који начин сте осигурали поверљивост и сигурност информација везаних за испитанике:

а) Подаци нису у отвореном приступу

б) Подаци су анонимизирани

ц) Остало, навести шта

_____

_____

5. Доступност података

5.1. Подаци ће бити

а) јавно доступни

б) доступни само уском кругу истраживача у одређеној научној области

(ц) затворени

Ако су подаци доступни само уском кругу истраживача, навести под којим условима могу да их користе:

_____

_____

Ако су подаци доступни само уском кругу истраживача, навести на који начин могу приступити подацима:

_____

_____

5.4. Навести лиценцу под којом ће прикупљени подаци бити архивирани.

_____

6. Улоге и одговорност

6.1. Навести име и презиме и мејл адресу власника (аутора) података

__Милош Радосављевић, milosradosavljevic@yahoo.com_____

6.2. Навести име и презиме и мејл адресу особе која одржава матрицу с подацима

_____

6.3. Навести име и презиме и мејл адресу особе која омогућује приступ подацима другим истраживачима

__Милош Радосављевић, milosradosavljevic@yahoo.com_____