PASSIVE AND ACTIVE SEISMIC DATA ANALYSES FOR HIGH TEMPORAL-RESOLUTION MONITORING AND HIGH SPATIAL-RESOLUTION IMAGING OF THE EARTH

フェルナンド, ラウレンス, フタペア

PASSIVE AND ACTIVE SEISMIC DATA ANALYSES FOR HIGH TEMPORAL-RESOLUTION MONITORING AND HIGH SPATIAL-RESOLUTION IMAGING OF THE EARTH

FERNANDO LAWRENS HUTAPEA

2020

PASSIVE AND ACTIVE SEISMIC DATA ANALYSES FOR HIGH TEMPORAL-RESOLUTION MONITORING AND HIGH SPATIAL-RESOLUTION IMAGING OF THE EARTH

By

FERNANDO LAWRENS HUTAPEA

A dissertation submitted in partial fulfillment of requirements

for the degree of

Doctoral Engineering

Examination Committee:

Prof. Takeshi Tsuji (Chairman)

Prof. Noriyuki Yasufuku

Assoc. Prof. Jun Nishijima

九州大学

KYUSHU UNIVERSITY

Department of Earth Resources Engineering

Graduate School of Engineering

Kyushu University

Japan

July 2020

Abstract

In seismic exploration, seismic source can be divided into two categories: active source and passive source. Active source refers to the artificial/man-made source. The examples of active source, that is related to seismic exploration, are dynamite, hammer, vibroseis, sparker and water gun. Passive source refers to the seismic source generated from natural phenomena. The examples of passive source related to seismic exploration are earthquake and ambient noise (e.g., wind and traffic noise). In general, the active source seismic exploration has higher spatial resolution image in comparison to passive source exploration. On the other hand, the active source exploration requires high cost, especially for long-term operation (i.e., time-lapse monitoring) with high temporal resolution. In this study, (1) the passive source was used for constructing real-time monitoring system with "high temporal resolution" and (2) the active source was used for constructing "high spatial resolution" profiles for hydrocarbon exploration. This dissertation contains four chapters, and the description for all chapters are shown below:

In Chapter 1, research objectives, background, and motivation are described. In this study, the ambient noise data was used for passive source exploration and deeply-towed multi-channel seismic data was used for active source exploration. In this chapter, some previous studies related to the utilization of ambient noise and the exploration using deep-towed data are introduced.

In Chapter 2, the ambient noise data was used for developing a spatio-temporal monitoring system in Japanese islands. This monitoring system with high temporal resolution is useful to continuously monitor crustal behavior. Since the monitoring results are associated with

earthquakes, magmatic activities, and several environmental effects (e.g. tides and rain precipitation), it can be utilized for disaster mitigation. Several sophisticated methods were used for processing of ambient noise data, such as seismic interferometry, stretching interpolation and temporal normalization. For estimating temporal velocity variation with high accuracy, the coda wave was used rather than direct wave. To improve the stability in real-time monitoring, Sliding Reference Method (SRM) was developed in this study. By using SRM, temporal velocity variation with high temporal resolution and stability can be obtained. There is also a module for removing the glitch/spike in temporal velocity variation automatically, based on Median Absolute Deviation (MAD) and median filter. Moreover, I described how to select an optimum parameter to achieve high temporal resolution result while keeping the stability. By using high parallel computation approach, this system can process huge data in short time. The temporal velocity variation derived from this continuous monitoring system is open through Kyushu University website.

In Chapter 3, the deep-towed Autonomous Cable Seismic (ACS) data was processed for finding hydrocarbon (e.g., free gas, gas chimney and gas hydrate) on Joetsu Basin, Japan. In this ACS, high spatial resolution imaging can be achieved by generating high-frequency signal close to the seafloor. The common problem that occurred on deep-towed ACS acquisition is the unstable depth position of source-receiver. The unstable depth of source-receiver caused an inaccurate reflection signal on pre-stack data. Moreover, the source signature of ACS data has sidelobes signal. The existence of sidelobes will decrease the frequency content on pre-stack data. It is crucial to remove the effect on unstable depth of source-receiver and sidelobes to make the stack image of subsurface result more obvious. Therefore, I developed a method to correct the unstable source-receiver depth, and the reflection signal can be accurately recovered. In order to suppress the sidelobes, a filter was designed so that the source signature becomes more focused. The result

of data processing showed significant improvements of results (i.e., reflection profiles): the velocity semblance more focused and the frequency content became wider. In order to characterize the gas chimney, gas hydrate and free-gas, seismic attributes were calculated. From the seismic attributes, the hydrate area is shown as high reflection amplitude on the seafloor, and the gas area is shown by a low amplitude signal and frequency fluctuation.

In Chapter 4, the result and key findings of this thesis are summarized, and future vision are described. The results derived from the proposed methods demonstrate that the ambient noise can be available for real-time (high temporal resolution) monitoring of velocity variation, and the deep-towed ACS data acquisition can be beneficial to construct high spatial resolution profile for exploring the hydrocarbon distribution. The interferometry and several other sophisticated computation technologies for ambient noise data have been used for developing an accurate, stable, and fast velocity monitoring system, hence this monitoring system has been currently operated for long term monitoring. In the future, not only the speed of computation performance will be faster, but also the number of observation locations that can record ambient noise will be larger using either conventional seismometer or fiber optic sensor. By applying this monitoring system for such huge data, monitoring of velocity variation on a larger scale (worldwide) and with more detailed spatial result can be achieved. Moreover, it can contribute for understanding the newest global earth phenomena and disaster mitigation. Applying the proposed data processing schemes for the deep-towed acquisition has given a promising result for high spatial resolution exploration. The deep-towed system is not only for exploring the hydrocarbon, but also for imaging other geological features, such as deep-sea ore deposits and slumping features.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research objectives

In this dissertation, the passive and active seismic source can be used for several applications such as:

a) Building a spatio-temporal monitoring system using the passive source waveform, which is natural earth vibration or ambient noise. This study made a monitoring system that can estimate spatio-temporal velocity variation accurately, has high temporal resolution result, can process a huge data quickly, and has ability to remove the glitch on velocity estimation automatically.

b) Exploring the location of free-gas, gas chimney and gas hydrate on deep marine environment by processing the deep-towed Autonomous Cable Seismic (ACS) acquisition which used a high frequency acoustic source.

1.2 Research background and motivation

Interferometry is a method that can be used to generate the virtual seismogram, which reproduce the wave propagation from one point observation to other point observation, by using cross-correlation (Wapenaar et al. 2010). The ambient noise is a kind of passive seismic source which obtained from natural earth vibration. By extracting the virtual seismogram form ambient noise data, the velocity variation on subsurface can be estimated by using stretching interpolation method (Liu et al. 2010; Nimiya et al. 2017; Ikeda and Tsuji 2018) or moving window cross spectrum (Liu et al. 2010; Clarke et al. 2011). Since the source cost is free, so it can be used for monitoring velocity variation with low cost budget in comparison to monitoring using active source (Lehujeur et al. 2015; Tsuji et al. 2016).

The interferometry method can be used for several monitoring purposes such as: monitoring volcano (Duputel et al. 2009; Mordret et al. 2010; Nagaoka et al. 2010; Taira and Brenguier 2016), monitoring geothermal reservoir (Lehujeur et al. 2015; Taira et al. 2018), estimating the velocity changes due to strain accumulation process (Ikeda and Tsuji 2018), and monitoring oil reservoir due to production process (Behm 2017). However, monitoring spatio-temporal velocity variation (for large region and long period) has not been well established due to unstable noise characteristic and scatters (Zhan et al. 2013). Indeed, Taira and Brenguier (2016) has made a small scale quasi-realtime monitoring system to monitor Lassen Volcanic Center in United States.

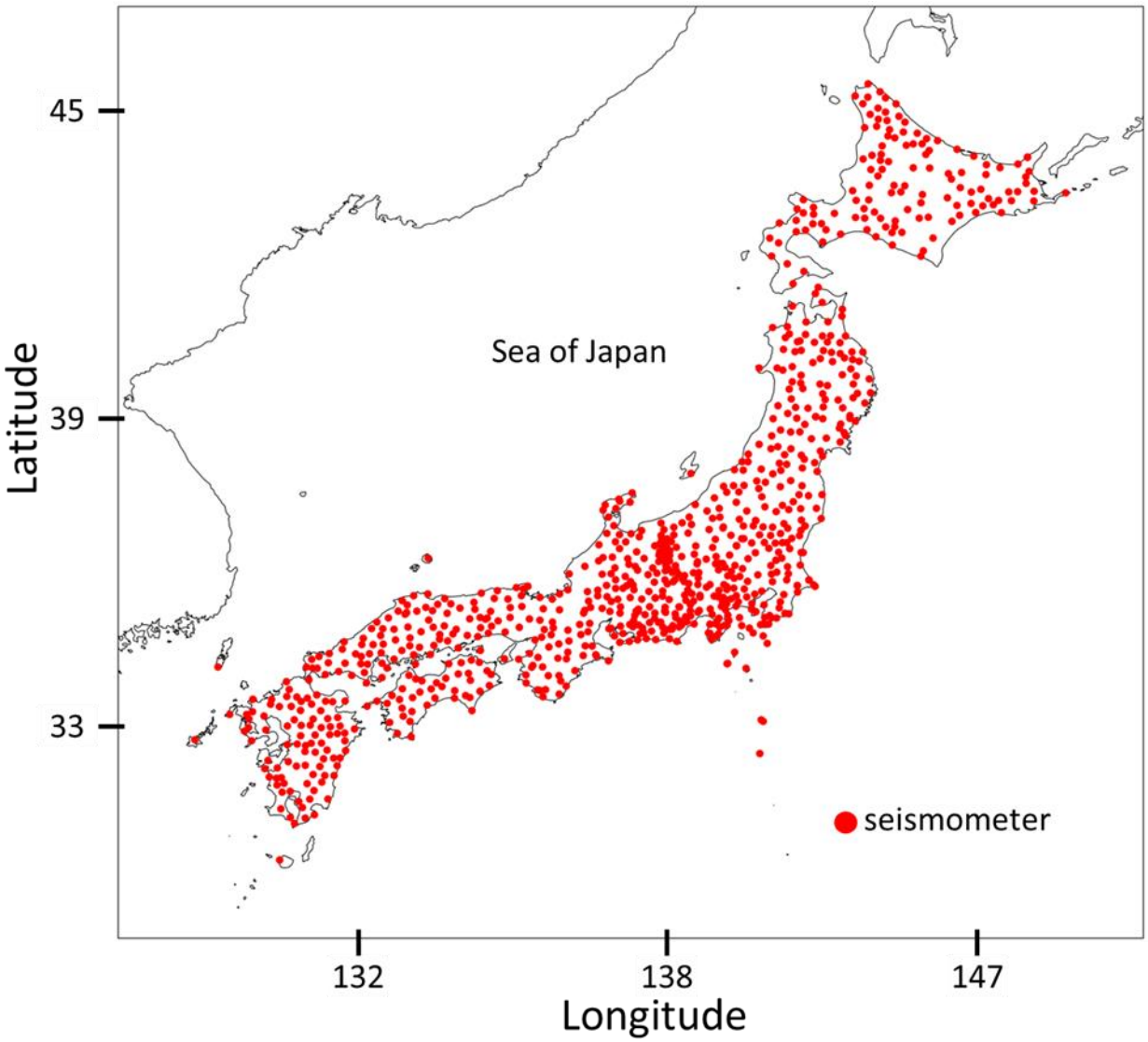

Figure 1.1 The location of NIED seismometers in Japanese islands. There are more than 750 seismometers

Japan has dense seismometer arrays that record the ambient noise continuously (Okada et al. 2004; Kaneda et al. 2015; Kawaguchi et al. 2015). This seismometer arrays are distributed along Hokkaido Island into Kyushu Island with the total station more than 750 (Figure 1.1) and under the management of National Research Institute for Earth Science and Disaster Resilience (NIED). By using this seismometer array and interferometry method, this study wants to develop a high-

temporal monitoring system that can estimate the spatio-temporal velocity variation continuously. Furthermore, this study wants to make the monitoring system that can calculate velocity variation accurately, can process huge ambient noise data quickly, and obtain high temporal resolution. Hopefully this monitoring system can give contribution for disaster prevention.

Deep towed survey is a data acquisition technique that designed for water environment such as: ocean, river, and lake. The deep-towed system can be used for doing active source acquisition such as: acoustic imaging survey (Asakawa et al. 2016, 2017, 2019), seismic reflection survey (Ross Chapman et al. 2002; Breitzke and Bialas 2003; Tokarev et al. 2008a; Leon et al. 2009; Ker et al. 2010), and electric resistivity survey (Goto et al. 2008). Also, it can be used for doing passive source acquisition such as: gravity survey (Zumberge et al. 1997) and magnetic survey (Gee et al. 2001). The JGI, Inc/J-MARES has developed a high resolution deep-towed acquisition system, called as Autonomous Cable Seismic (ACS) (Asakawa et al. 2016, 2019). The ACS use high frequency acoustic source and 32 hydrophone receivers (Figure 1.2). This study used the ACS data that located in Joetsu Basin Japan (Figure 1.3). Several studies has reported about the existence of gas hydrate, gas chimney, and free gas in Joetsu Basin (Figure 1.3) (Matsumoto et al. 2009; Freire et al. 2011; JAMSTEC 2012a, b).

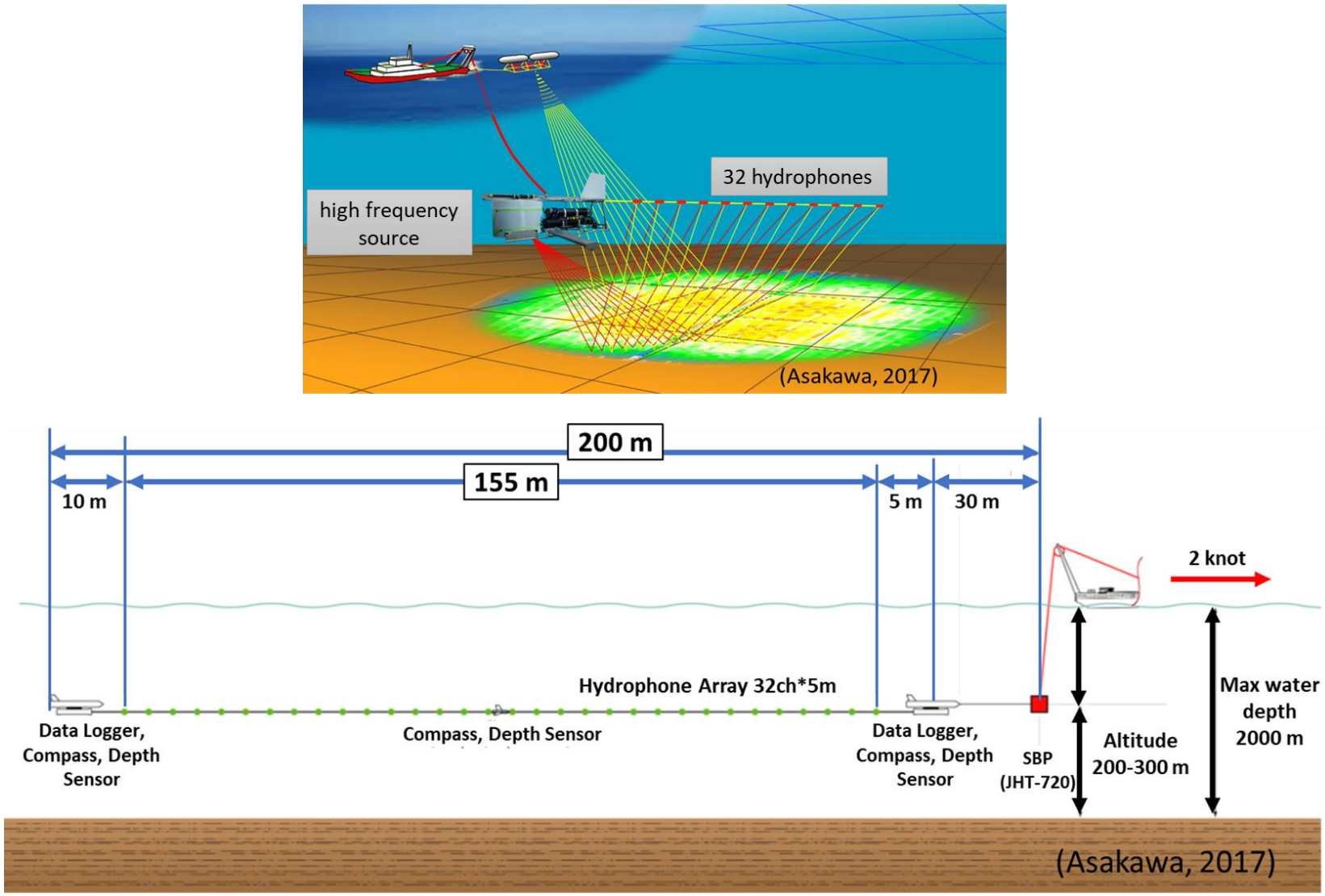

Figure 1.2 The schematic diagram of ACS developed by JGI Inc/J-MARES. The ACS has high frequency acoustic source and 32 receivers.

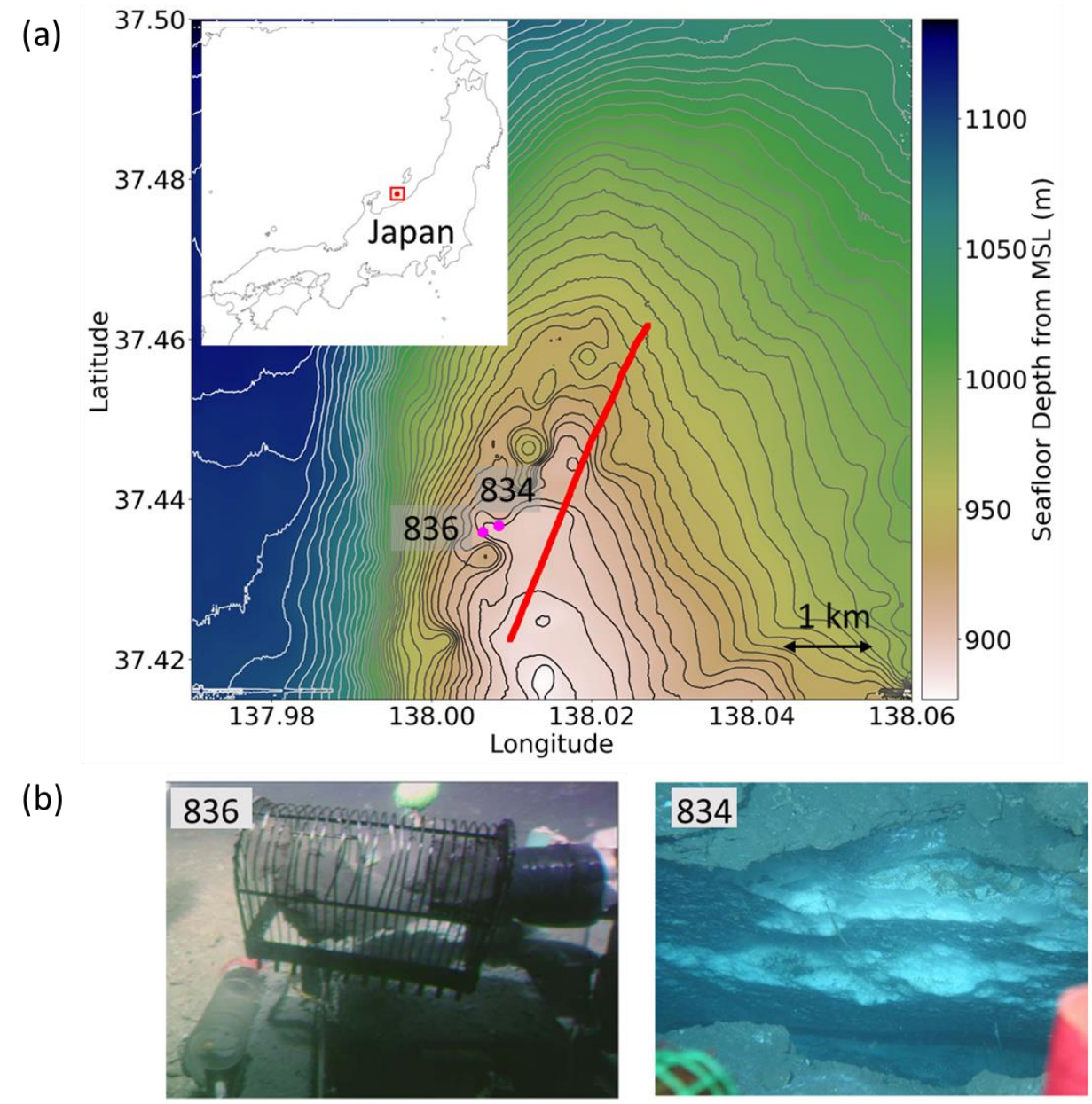

Figure 1.3 The base-map of ACS data obtained in Joetsu Basin, Japan. In the panel a), the red line is the location of ACS profile that used in this research, and the magenta points are the location of hydrate outcrops. The panel b) show the example of hydrate outcrops near the seafloor that reported by JAMSTEC.

The common issue when doing deep-towed acquisition is the difficulty to stabilize the depth position of sources and receivers (Tokarev et al. 2008a; Marsset et al. 2014). The unstable source and receiver may make an inaccurate reflection waveform. Further, it can cause the destructive on stack profile (Yilmaz 2001; Kluesner et al. 2019). Moreover, the existence of sidelobes on source signature can decrease the frequency content of ACS data (Nelson 1989). This study wants to eliminate the effect of unstable source-receivers on pre-stack ACS data and suppress the sidelobes hence the resulted stack section will give an appropriate image. The existence of gas chimney, gas hydrate and free-gas make a disruption on amplitude and frequency of reflection signal. These features can be emphasized by post-stack seismic attributes (Tsuji et al. 2012). The seismic attributes were used for characterizing those gases such as: chaos, root mean square (RMS) amplitude, instantaneous frequency, and absolute amplitude (Barnes 1992, 2016; Chopra and Marfurt 2007; Ojha and Sain 2009; Kumar et al. 2019).

1.3 Chapter Description

In this dissertation there two mains topics for the utilization of active and passive such as: building a continuous spatio-temporal monitoring system in Japanese island using ambient noise data (Chapter 2); using Autonomous Cable Seismic data acquisition to characterize the hydrocarbon (gas) potential (Chapter 3).

Chapter 2 describes about developing a system to continuously monitor the spatio-temporal velocity variation in Japanese Island with high-temporal resolution. This system used the passive seismic source, which is natural earth vibration or ambient noise. There are four steps that need to be applied for processing the ambient noise data into spatio-temporal velocity

changes such as: preparing ambient-noise data; creating virtual seismograms between pairs of seismometer stations by applying seismic interferometry; estimating temporal velocity variations from virtual seismograms by stretching interpolation approach, and mapping spatio-temporal velocity variations. In the part of estimating temporal velocity variation, this study proposes a method called as "Sliding Reference Method (SRM)". The SRM can obtain velocity variation result with high temporal resolution and near to newest geologic condition. In addition to data processing, this study developed a module to remove the unstable velocity variation automatically by using Median Absolute Deviation (MAD) and median filter. Moreover, this system was prepared to process data with huge size quickly. Therefore, high performance parallel computation was applied in this monitoring system. (*The contents on this chapter was published on Earth, Planets, and Space (Hutapea et al. 2020))*

Chapter 3 describes about processing deep-marine data that used a high frequency active source. This data was obtained by using a deep-towed acquisition system named as Autonomous Cable Seismic (ACS) that developed by JGI, Inc/J-MARES. The common problem on deep-towed data acquisition is the difficulty to control the depth position of sources and receivers due to strong ocean current or rough seafloor topography. Because of the unstable depth-receiver, the reflected wave of lithology target was recorded inaccurately. This study proposed a data processing sequence that can minimize the effect of unstable source-receiver hence the shape of reflected wave can be improved. Furthermore, a filter for focusing the source signature of ACS was made. This filter was useful for boosting the frequency content of the ACS pre-stack data. In addition to that, the seismic attribute was carried out for identification the presence of free-gas, gas-hydrate, and gas chimney. *(The materials on this chapter will submitted on Journal of Natural Gas Science and Engineering 2020 (under revision)).*

Chapter 4 summarizes the dissertation content. The passive source can be an option for building a system that can monitor the spatio-temporal velocity variation with long term

duration. The active source (especially high frequency source) can be used for doing high resolution crustal imaging on deep-marine environment.

1.4 References

Asakawa, E., Murakami, F., Saito, S., Tara, K., Kato, M., Tsukahara, H., Lee, S., Sumi, T., Yamakawa, T., 2017. New seismic survey techniques for seafloor massive sulphide (SMS) Exploration. 23rd Eur. Meet. Environ. Eng. Geophys. https://doi.org/10.3997/2214-4609.201702091

Asakawa, E., Sumi, T., Yamakawa, T., Kose, M., 2016. Multi-stage and integrated approach for seafloor massive sulfide (SMS) exploration, in: 22nd European Meeting of Environmental and Engineering Geophysics, Near Surface Geoscience 2016.

Asakawa, E., Tara, K., Saito, S., Murakami, F., Tsukahara, H., 2019. Approach to Focusing the SMS Potential Area using High-Resolution Seismic Surveys, in: OCEANS 2019 - Marseille. IEEE, pp. 1–5. https://doi.org/10.1109/OCEANSE.2019.8867141

Barnes, A.E., 2016. Handbook of Poststack Seismic Attributes, Handbook of Poststack Seismic Attributes. Society of Exploration Geophysicists. https://doi.org/10.1190/1.9781560803324

Barnes, A.E., 1992. The calculation of instantaneous frequency and instantaneous bandwidth. GEOPHYSICS. https://doi.org/10.1190/1.1443220

Behm, M., 2017. Feasibility of borehole ambient noise interferometry for permanent reservoir monitoring. Geophys. Prospect. https://doi.org/10.1111/1365-2478.12424

Breitzke, M., Bialas, J., 2003. A deep-towed multichannel seismic streamer for very high-resolution surveys in full ocean depth. First Break 21, 59–65.

Chopra, S., Marfurt, K.J., 2007. Seismic Attributes for Prospect Identification and Reservoir

Characterization. Society of Exploration Geophysicists and European Association of Geoscientists and Engineers. https://doi.org/10.1190/1.9781560801900

Clarke, D., Zaccarelli, L., Shapiro, N.M., Brenguier, F., 2011. Assessment of resolution and accuracy of the Moving Window Cross Spectral technique for monitoring crustal temporal variations using ambient seismic noise. Geophys. J. Int. 186, 867–882. https://doi.org/10.1111/j.1365-246X.2011.05074.x

Duputel, Z., Ferrazzini, V., Brenguier, F., Shapiro, N., Campillo, M., Nercessian, A., 2009. Real time monitoring of relative velocity changes using ambient seismic noise at the Piton de la Fournaise volcano (La Réunion) from January 2006 to June 2007. J. Volcanol. Geotherm. Res. https://doi.org/10.1016/j.jvolgeores.2008.11.024

Freire, A.F.M., Matsumoto, R., Santos, L.A., 2011. Structural-stratigraphic control on the Umitaka Spur gas hydrates of Joetsu Basin in the eastern margin of Japan Sea. Mar. Pet. Geol. 28, 1967–1978. https://doi.org/10.1016/j.marpetgeo.2010.10.004

Gee, J.S., Webb, S.C., Ridgway, J., Staudigel, H., Zumberge, M.A., 2001. A deep tow magnetic survey of Middle Valley, Juan de Fuca Ridge. Geochemistry, Geophys. Geosystems. https://doi.org/10.1029/2001gc000170

Goto, T.N., Kasaya, T., MacHiyama, H., Takagi, R., Matsumoto, R., Okuda, Y., Satoh, M., Watanabe, T., Seama, N., Mikada, H., Sanada, Y., Kinoshita, M., 2008. A marine deep-towed DC resistivity survey in a methane hydrate area, Japan Sea. Explor. Geophys. https://doi.org/10.1071/EG08003

Ikeda, T., Tsuji, T., 2018. Temporal change in seismic velocity associated with an offshore MW 5.9 Off-Mie earthquake in the Nankai subduction zone from ambient noise cross-correlation. Prog. Earth Planet. Sci. 5. https://doi.org/10.1186/s40645-018-0211-8

JAMSTEC, 2012a. NATSUSHIMA NT08-09 Leg2 Cruise Data. Japan Agency for Marine-Earth Science and Technology (JAMSTEC). https://doi.org/10.17596/0000488

JAMSTEC, 2012b. NATSUSHIMA NT08-09 Leg1 Cruise Data. Japan Agency for Marine-Earth Science and Technology (JAMSTEC). https://doi.org/10.17596/0000487

Kaneda, Y., Kawaguchi, K., Araki, E., Matsumoto, H., Nakamura, T., Kamiya, S., Ariyoshi, K., Hori, T., Baba, T., Takahashi, N., 2015. Development and application of an advanced ocean floor network system for megathrust earthquakes and tsunamis, in: Seafloor Observatories: A New Vision of the Earth from the Abyss. https://doi.org/10.1007/978-3-642-11374-1_25

Kawaguchi, K., Kaneko, S., Nishida, T., Komine, T., 2015. Construction of the DONET real-time seafloor observatory for earthquakes and tsunami monitoring, in: Seafloor Observatories: A New Vision of the Earth from the Abyss. https://doi.org/10.1007/978-3-642-11374-1_10

Ker, S., Marsset, B., Garziglia, S., Le Gonidec, Y., Gibert, D., Voisset, M., Adamy, J., 2010. High-resolution seismic imaging in deep sea from a joint deep-towed/OBH reflection experiment: application to a Mass Transport Complex offshore Nigeria. Geophys. J. Int. 182, 1524–1542. https://doi.org/10.1111/j.1365-246X.2010.04700.x

Kluesner, J., Brothers, D., Hart, P., Miller, N., Hatcher, G., 2019. Practical approaches to maximizing the resolution of sparker seismic reflection data. Mar. Geophys. Res. 40, 279–301. https://doi.org/10.1007/s11001-018-9367-2

Kumar, J., Sain, K., Arun, K.P., 2019. Seismic attributes for characterizing gas hydrates: a study from the Mahanadi offshore, India. Mar. Geophys. Res. 40, 73–86. https://doi.org/10.1007/s11001-018-9357-4

Lehujeur, M., Vergne, J., Schmittbuhl, J., Maggi, A., 2015. Characterization of ambient seismic noise near a deep geothermal reservoir and implications for interferometric methods: a case study in northern Alsace, France. Geotherm. Energy. https://doi.org/10.1186/s40517-014-0020-2

Leon, P., Ker, S., Gall, Y.L., Marsset, B., Voisset, M., 2009. SYSIF a new seismic tool for near bottom very High resolution profiling in deep water, in: OCEANS 2009-EUROPE. IEEE, pp. 1–5. https://doi.org/10.1109/OCEANSE.2009.5278154

Liu, Z., Huang, J., Li, J., 2010. Comparison of four techniques for estimating temporal change of seismic velocity with passive image interferometry. Earthq. Sci. https://doi.org/10.1007/s11589-010-0749-z

Marsset, B., Menut, E., Ker, S., Thomas, Y., Regnault, J.P., Leon, P., Martinossi, H., Artzner, L., Chenot, D., Dentrecolas, S., Spychalski, B., Mellier, G., Sultan, N., 2014. Deep-towed high resolution multichannel seismic imaging. Deep. Res. Part I Oceanogr. Res. Pap. 93, 83–90. https://doi.org/10.1016/j.dsr.2014.07.013

Matsumoto, R., Okuda, Y., Hiruta, A., Tomaru, H., Takeuchi, E., Sanno, R., Suzuki, M., Tsuchinaga, K., Ishida, Y., Ishizaki, O., Takeuchi, R., Komatsubara, J., Freire, A.F., Machiyama, H., Aoyama, C., Joshima, M., Hiromatsu, M., Snyder, G., Numanami, H., Satoh, M., Matoba, Y., Nakagawa, H., Kakuwa, Y., Ogihara, S., Yanagawa, K., Sunamura, M., Goto, T., Lu, H., Kobayashi, T., 2009. Formation and Collapse of Gas Hydrate Deposits in High Methane Flux Area of the Joetsu Basin, Eastern Margin of Japan Sea. Chigaku Zasshi (Jounal Geogr. 118, 43–71. https://doi.org/10.5026/jgeography.118.43

Mordret, A., Jolly, A.D., Duputel, Z., Fournier, N., 2010. Monitoring of phreatic eruptions

using Interferometry on Retrieved Cross-Correlation Function from Ambient Seismic Noise: Results from Mt. Ruapehu, New Zealand. J. Volcanol. Geotherm. Res. https://doi.org/10.1016/j.jvolgeores.2010.01.010

Nagaoka, Y., Nishida, K., Aoki, Y., Takeo, M., 2010. Temporal change of phase velocity beneath Mt. Asama, Japan, inferred from coda wave interferometry. Geophys. Res. Lett. 37, 1–5. https://doi.org/10.1029/2010GL045289

Nelson, D.E., 1989. Methods and apparatus for reducing correlation sidelobe interference in seismic profiling systems. J. Acoust. Soc. Am. https://doi.org/10.1121/1.398650

Nimiya, H., Ikeda, T., Tsuji, T., 2017. Spatial and temporal seismic velocity changes on Kyushu Island during the 2016 Kumamoto earthquake. Sci. Adv. 3. https://doi.org/10.1126/sciadv.1700813

Ojha, M., Sain, K., 2009. Seismic attributes for identifying gas-hydrates and free-gas zones: application to the Makran accretionary prism. Episodes 32, 264–270. https://doi.org/10.18814/epiiugs/2009/v32i4/003

Okada, Y., Kasahara, K., Hori, S., Obara, K., Sekiguchi, S., Fujiwara, H., Yamamoto, A., 2004. Recent progress of seismic observation networks in Japan - Hi-net, F-net, K-net and KiK-net -. Earth, Planets Sp. https://doi.org/10.1186/BF03353076

Ross Chapman, N., Gettrust, J.F., Walia, R., Hannay, D., Spence, G.D., Wood, W.T., Hyndman, R.D., 2002. High-resolution, deep-towed, multichannel seismic survey of deep-sea gas hydrates off western Canada. Geophysics 67, 1038–1047. https://doi.org/10.1190/1.1500364

Taira, T., Avinash, N., Brenguier, F., Manga, M., 2018. Monitoring reservoir response to earthquakes and fluid extraction, salton sea geothermal field, California. Sci. Adv.

https://doi.org/10.1126/sciadv.1701536

Taira, T., Brenguier, F., 2016. Response of hydrothermal system to stress transients at Lassen Volcanic Center, California, inferred from seismic interferometry with ambient noise 4. Seismology. Earth, Planets Sp. https://doi.org/10.1186/s40623-016-0538-6

Tokarev, M., Kuzub, N., Pevzner, R., Kalmykov, D., Bouriak, S., 2008. High resolution 2D deep-towed seismic system for shallow water investigation. First Break 26.

Tsuji, T., Ikeda, T., Johansen, T.A., Ole Ruud, B., 2016. Using seismic noise derived from fluid injection well for continuous reservoir monitoring. Interpretation. https://doi.org/10.1190/INT-2016-0019.1

Tsuji, T., Takai, K., Oiwane, H., Nakamura, Y., Masaki, Y., Kumagai, H., Kinoshita, M., Yamamoto, F., Okano, T., Kuramoto, S., 2012. Hydrothermal fluid flow system around the Iheya North Knoll in the mid-Okinawa trough based on seismic reflection data. J. Volcanol. Geotherm. Res. 213–214, 41–50. https://doi.org/10.1016/j.jvolgeores.2011.11.007

Wapenaar, K., Draganov, D., Snieder, R., Campman, X., Verdel, A., 2010. Tutorial on seismic interferometry: Part 1 — Basic principles and applications. Geophysics 75, 75A195-75A209. https://doi.org/10.1190/1.3457445

Yilmaz, Ö., 2001. Seismic Data Analysis: Processing, Inversion, and Interpretation of Seismic Data, Investigations in geophysics. https://doi.org/10.1190/1.9781560801580

Zhan, Z., Tsai, V.C., Clayton, R.W., 2013. Spurious velocity changes caused by temporal variations in ambient noise frequency content. Geophys. J. Int. 194, 1574–1581. https://doi.org/10.1093/gji/ggt170

Zumberge, M.A., Ridgway, J.R., Hildebrand, J.A., 1997. A towed marine gravity meter for near-bottom surveys. Geophysics. https://doi.org/10.1190/1.1444243

Chapter 2

Real-time crustal monitoring system of Japanese Islands based on spatio-temporal seismic velocity variation

Abstract

To continuously monitor crustal behavior associated with earthquakes, magmatic activities and other environmental effects (e.g., tides and rain precipitation), this study has developed a continuous monitoring system of seismic velocity of the Japanese Islands. The system includes four main processing procedures to obtain spatio-temporal velocity changes: (1) preparing ambient-noise data; (2) creating virtual seismograms between pairs of seismometer stations by applying seismic interferometry; (3) estimating temporal velocity variations from virtual seismograms by stretching interpolation approach, and (4) mapping

spatio-temporal velocity variations. Moreover, this study has developed data-processing scheme to removes unstable stretching interpolation results by using the median absolute deviation technique and a median filter. To map velocity changes with high stability and high temporal resolution during long-term (i.e., longer-term monitoring), the “Sliding Reference Method” was proposed. This study also developed evaluation method to select the optimum parameters related to stability and temporal resolution. To reduce computation time for continuous monitoring, parallel computation methods were applied, such as shared memory and hybrid distributed memory parallelization. This efficient and stable monitoring system was succeeded to continuously monitor the spatio-temporal velocity variation of the whole Japanese Islands using ambient-noise data from 767 seismometers. Finally, a web application was developed to displays spatio-temporal velocity changes. In the monitoring results that has been opened through the website, the velocity variation (e.g., pore pressure variation) that could be related to earthquake, aftershock, magmatic activities and environmental effects can be identified in a stable manner.

Keywords: crustal monitoring, seismic velocity, parallel and high-performance computing, big data, seismic interferometry, ambient noise

2.1 Introduction

Changes in the subsurface environment have been often monitored through repeated seismic surveys using active sources (Chadwick et al. 2010), but the high cost of active-source seismic surveys makes it problematic for continuously monitoring subsurface behavior. The use of a permanent seismic source system such as the accurately controlled routinely operated signal system (ACROSS) (Kumazawa and Takei 1994; Yamaoka et al. 2008) is an effective approach to enhance temporal resolution and source repeatability. In ACROSS, repeatable signals can be continuously generated by rotating eccentric mass. Previous ACROSS-based monitoring succeeded in identifying temporal changes in seismic velocity associated with earthquakes (Ikuta et al. 2002; Ikuta and Yamaoka 2004), volcanic activity (Maeda et al. 2015), ground freezing (Ikeda et al. 2017), and secular velocity change (Tsuji et al. 2018). However, the number of ACROSS units is currently limited, resulting in low spatial resolution in the monitoring.

Seismic interferometry is an alternative method which can estimate variations in seismic velocity using ambient-noise data (e.g., Brenguier et al. 2008; Wegler et al. 2009). This method avoids the cost of active sources by relying on natural earth vibrations as the seismic source (e.g., Brenguier et al. 2008; Tsuji et al. 2016). Seismic interferometry can extract virtual seismograms, simulating the signal recorded at a station from a seismic event occurring at the location of another station, by computing the cross-correlation of ambient-noise data (Wapenaar et al. 2010). To estimate velocity changes, coda waves in the virtual seismograms are used rather than direct waves, because coda waves are more sensitive to velocity changes in geological materials (Meier et al. 2010) and are less affected by variations of noise sources (e.g., Chaves and Schwartz 2016). Several studies have successfully used this method to

estimate temporal seismic velocity variations (Brenguier et al. 2008, 2014; Ikeda and Tsuji 2018; Meier et al. 2010; Obermann et al. 2013; Wegler et al. 2009). However, mapping of temporal velocity variation for longer-term (i.e., spatio-temporal monitoring; Wang et al. 2017) has not been well established, particularly when unstable velocity variations due to temporal variation of noise source characteristics (Zhan et al. 2013) and scatterers preclude us to map the monitoring results in a stable manner. Therefore, this study develops a method to obtain stable monitoring results from many seismometer pairs for long-term with high temporal resolution. Furthermore, this study implements a high-performance computation scheme to continuously calculate velocity variation from huge ambient-noise data. Note that Nagaoka et al. (2010) monitored temporal change in Mt. Asama, Japan using S-coda for multiple natural earthquakes based on seismic interferometry. Their approach has a potential to exclude the biases introduced by temporal variation of noise source characteristics, but the temporal resolution depends on the occurrence of natural earthquakes. Monitoring of seismic velocity variations has a wide range of applications. Velocity changes can indicate subsurface changes caused by natural activities such as earthquakes, volcanic eruptions, and seasonal cycles. Continuous monitoring may lead to the use of seismic velocity changes in forecasting volcanic eruptions or characterizing aftershock sequences (Brenguier et al. 2008; DeVries et al. 2018; Nimiya et al. 2017). Furthermore, monitoring velocities can be used to track sub-surface behaviors due to human activities such as CO_2 geological storage, oil/gas production, and geothermal reservoir managements. If seismic velocities can be used to monitor pore pressure variations in resource reservoirs due to fluid injection or extraction, the information derived from monitoring could help avoid induced earthquakes. Indeed, Taira et al. (2018) estimated the reservoir response due to fluid extraction at the Salton Sea geothermal field. Permanent seismic networks now provide continuous seismic data in the United States, Japan and elsewhere (Kaneda et al. 2015; Kawaguchi et al. 2015; Okada et al. 2004). Ambient-noise data

(i.e., big data) derived from such networks can be used to conduct real-time monitoring of seismic velocity. For example, Taira and Brenguier (2016) constructed a quasi-real-time monitoring system at the Lassen volcanic center in California using ambient-noise data.

Recently, seismic velocity variation beneath the seafloor due to strain accumulation process has been monitored by ocean bottom seismometers (Ikeda and Tsuji 2018). In ambient-noise seismic monitoring, velocity variations were calculated between two seismic traces acquired at different times. One trace, called the current trace, is defined by stacking cross-correlations over a short period (e.g., 10 days), and the other is a reference trace, typically obtained by stacking cross-correlation over a longer period (e.g., 1 year) (Nimiya et al. 2017). Because the velocity change is estimated with respect to the reference trace, the definition of the reference trace is crucial. In previous studies (e.g., Hobiger et al. 2016; Ikeda and Tsuji 2018; Meier et al. 2010; Nimiya et al. 2017; Obermann et al. 2013), reference traces were defined by stacking all the cross-correlations for the respective study periods. However, when compiling monitoring results for longer terms, it is problematic to use such a fixed reference trace. Because the characteristics of cross-correlations vary with time due to earthquakes and magmatic activities, the quality of a reference trace may decrease even when the number of stacking was increased. The results derived from ambient-noise monitoring are influenced by temporal variation of scatters and noise sources (Nimiya et al. 2017). Furthermore, there is a trade-off with current traces in that longer periods of data yield more stable results, but shorter periods offer better temporal resolution. This study built a system for continuous velocity monitoring with high temporal resolution using ambient noise and applied the system to the seismometers distributed on the whole Japanese Islands. To evaluate processing parameters in details, seismometers in Kyushu Island were used. In this area, velocity changes associated with the 2016 Kuma-moto earthquake (Mw 7.0) and eruption of Aso volcano could be identified (Nimiya et al. 2017). Two different methods were considered for calculating

reference traces: (1) the Absolute Reference Method (ARM) and (2) the Sliding Reference Method (SRM) that is a new approach for defining reference and current traces. To select an optimum time window of current traces for these two methods, a stability evaluation was performed. Furthermore, this study developed an outlier removal method based on Median Absolute Deviation (MAD) (Leys et al. 2013) and a median filter to remove unstable results of temporal velocity changes. Because the continuous monitoring system requires fast computation with proper memory management, this system was designed by using shared memory parallelization and hybrid distributed memory parallelization (cluster parallelization). This system was constructed with Python 3.6.4 (Python Software Foundation 2017) from the Anaconda Distribution to support any Linux distribution. The parallel computation design used seven server nodes to achieve greater than six fold increases in computation speed. To display the monitoring results, a web application was installed on a virtual private server to give free access to the updated 1 year monitoring results.

2.2 Methods

The monitoring system retrieved velocity information from ambient-noise data in four processing steps (Figure 2.1) described in this section.

2.2.1 Preparing data

This study obtained ambient-noise data recorded by the Hi-net seismic network from the National Research Institute for Earth Science and Disaster Prevention (NIED) server (NIED 2019). In the monitoring of whole Japanese Islands, the ambient-noise data of 767 Hi-net stations were used. The vertical component of ambient-noise data was selected, then applied de-meaning, de-trending, and instrument correction (adjusting the recorded signal that has frequency below the instrument natural frequency) for the data from each station. A band-pass filter was applied with the frequency range of 0.1–0.9 Hertz (Hz) because this frequency range has been proven to estimate velocity changes using Hi-net stations (Minato et al. 2012; Nimiya et al. 2017). Furthermore, this study divided daily data into 30-min-long segments with 50% overlap. To reduce the data volume, the output from this process was saved in the frequency domain. Data segments with 1 s or more of missing waveforms were rejected. Because earthquakes degrade the quality of ambient noise, one-bit normalization was applied to increase the ambient-noise signal and reduce other noise such as: earthquake, instrument instability, non-stationary source signal close to station (e.g., Bensen et al. 2007; Hobiger et al. 2016; Meier et al. 2010; Minato et al. 2012). One-bit normalization has been shown to enhance the stability of monitoring results for the receiver and noise characteristics. The previous study (Hutapea et al. 2019) demonstrated that one-bit normalization allows us to obtain stable velocity variations in Hi-net ambient-noise monitoring with similar processing parameters.

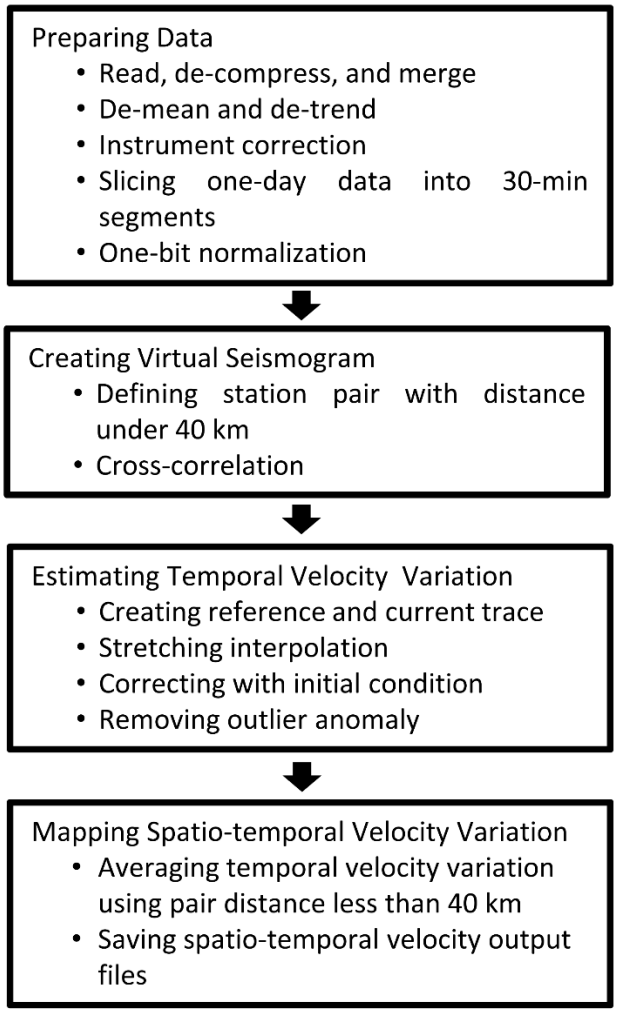

Figure 2.1 Schematic diagram of data processing workflow for estimating spatio-temporal velocity variations

2.2.2 Creating virtual seismograms

By computing daily cross-correlations between the sites of two seismometers, virtual seismograms (i.e., Green's functions) between the two seismometers can be retrieved (Wapenaar et al. 2010). The power-normalized cross-correlations (cross-coherence) was computed in the frequency domain (e.g., Nakata et al. 2011) from

$$CC_{AB}(f) = \frac{F_A(f)\, F_B^*(f)}{|F_A(f)||F_B(f)|}, \quad (2.1)$$

where CC_{AB} is the power-normalized cross-correlation at frequency f between seismometers A and B, F_A and F_B are the Fourier transforms of the seismograms at each seismometer, and F_B^* is the complex conjugate version of F_B. The resulting cross-correlation consists of the positive time (causal) and negative time (anticausal) parts, corresponding to the wave propagation from seismometer B to A and from seismometer A to B, respectively. This study use the trace side that the waveform propagates to Mount Aso (positive time) in order to reduce strong time-variant tremor noise from Mount Aso (Kawakatsu et al. 2000; Nimiya et al. 2017; Hendriyana and Tsuji 2019) (Figure 2.2). Cross-correlations were calculated for station pairs whose distance apart, as computed by using the Haversine equation (Van Brummelen et al. 2012), was less than 40 km. Whereas, several seismometer pairs were added whose distance is > 40 km where the station pair is sparse. In total, there are 7235 pairs of seismic stations that distributed on the Japanese Islands.

2.2.3 Estimating temporal velocity variation

To obtain temporal velocity variations from cross-correlations between two seismometers, this study can use either the stretching interpolation method (e.g., Meier et al. 2010; Minato et al. 2012; Nimiya et al. 2017; Yukutake et al. 2016) or the moving-window cross-spectral (MWCS) analysis (e.g., Ratdomopurbo and Poupinet 1995; Clarke et al. 2011) to estimate the velocity changes between a current trace and a reference trace (Figure 2.2). Nimiya et al. (2017) compared velocity changes estimated by the stretching method and the MWCS for Hi-net data in Kyushu Island, and demonstrated that when using a single side of

cross-correlation to reduce the influence of noise from Mount Aso, the stretching method produced more reliable results.

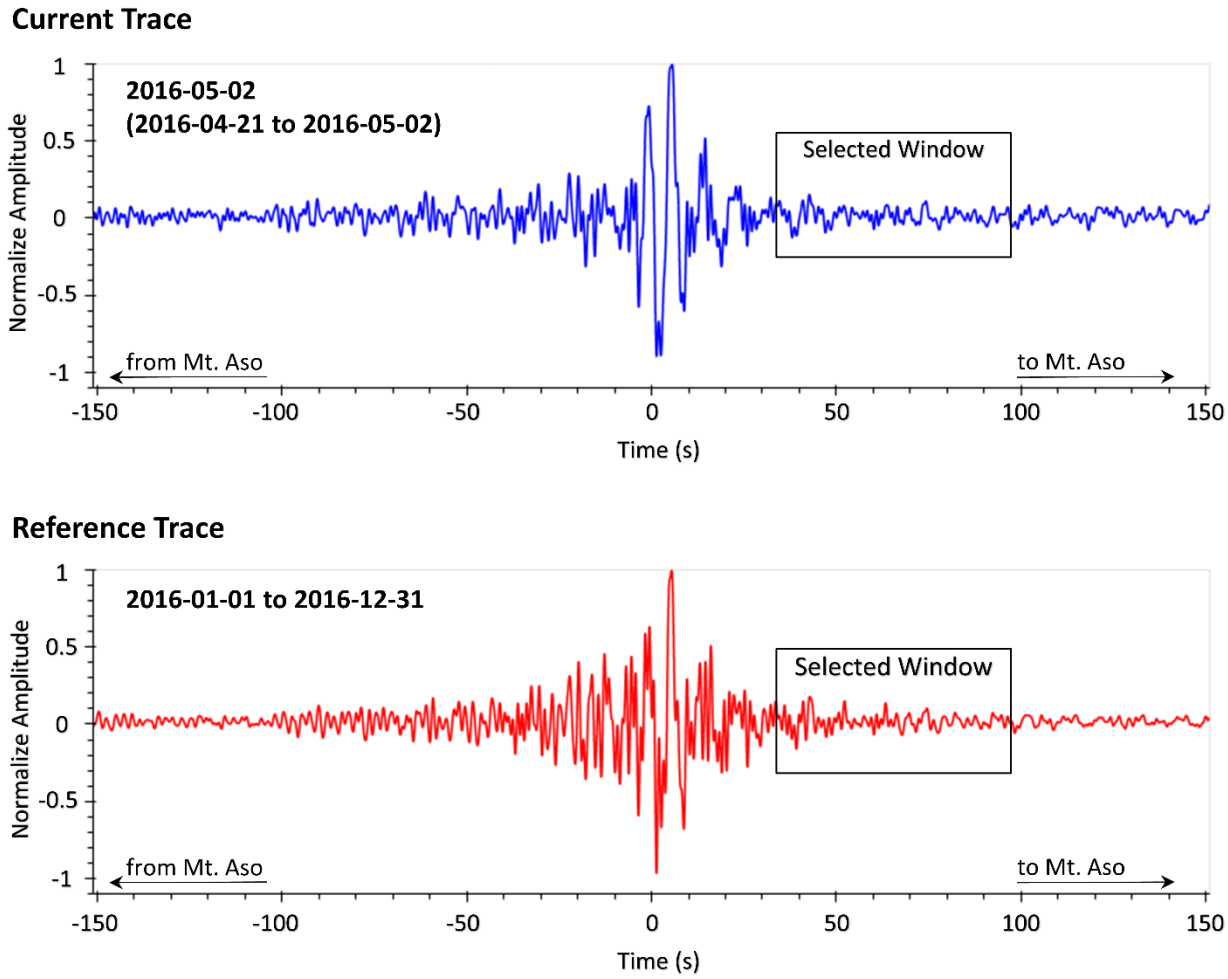

Figure 2.2 Example of the current trace (blue) and the reference trace (red). The rectangles represent the time windows of the coda part of the seismograms, used to estimate seismic velocity changes by the stretching interpolation method. In order to reduce strong noise from Mt Aso, the trace side that the waveform propagates to Mt. Aso (positive time) was used.

Furthermore, the MWCS method underestimates large changes in seismic velocity due to period skipping in the coda of the cross-correlation (Olivier et al. 2017). Therefore, to obtain reliable velocity variation including large velocity changes, the stretching interpolation

technique was used. The current trace f^{cur} was stretched (shrink and expand) and compute the similarity between the stretched current trace f_E^{cur} and the reference trace f^{ref} as follows:

$$C(E) = \frac{\int f_E^{cur}(t) f^{ref}(t) dt}{\left(\int (f_E^{cur}(t))^2 dt \int (f^{ref}(t))^2 dt \right)^{1/2}} \tag{2.2}$$

$$f_E^{cur}(t) = f^{cur}(t(1+E)) \tag{2.3}$$

where E ($= -dV/V$) is the relative velocity change with respect to the reference trace and $C(E)$ is the cross-correlation coefficient between the reference trace and the stretched version of the current trace. The relative velocity change E that maximizes $C(E)$ was estimated by applying a grid search algorithm that searched E in the range $-0.025 < E < 0.025$ with a step size of 0.0005. After the grid search result was obtained, then ternary search algorithm was applied to increase the resolution of E values and to search maximum local around obtained $C(E)$. The ternary search was applied 10 times with the new step size is half of the previous iteration step size. This study used 100 s of coda waves to obtain velocity changes using stretching interpolation. The starting time of coda waves was defined as $t_{coda} = d/v_{min}$, where d is the distance between a station pair and v_{min} is the minimum apparent velocity between the stations. As v_{min}, this study used a relatively slow velocity of 1 km/s to select coda parts.

2.2.3.1 Define current and reference traces

The choice of periods of current and reference traces is important to obtain stable velocity changes with high temporal resolution. To estimate daily velocity changes, the time window (stacking period) for the current trace is fixed (N days) and the window is slid forward each day (Figure 2.3 and 2.4). The latest day of the time window was used for the current trace

as the date of the monitoring result. To define the reference traces, two methods were evaluated such as: the Absolute Reference Method (ARM) and the Sliding Reference Method (SRM).

In the ARM, the reference trace is defined by stacking *M* days of data fixed at a particular time period (Figure 2.3). Furthermore, it slides the current trace, with its shorter time window (*N* days), and estimate the velocity change between reference and current traces by stretching interpolation. This approach (ARM) was often used in previous studies (e.g., Meier et al. 2010; Nimiya et al. 2017). On the other hand, SRM is a new approach that this study proposed. In the SRM, the reference trace is also defined by stacking *M* days of data, but it slides each day (Figure 2.4). The current trace (*N* days) slides within the period of the reference trace (*M* days), and the latest days of both traces are always the same.

Because of the velocity variations estimated from different reference traces are difficult to compare in SRM, this study calculated temporal velocity changes within the fixed reference period each day. To do this, this study applied stretching interpolation $M-(N+1)$ times for each station pair, using the fixed reference trace and sliding the current trace. Therefore, SRM needs a longer computation time than ARM. Furthermore, this study subtracted the average value of *E* for the first 30 days of the monitoring period (E_0) from the estimated value of *E* as follows:

$$E' = E - E_0, \tag{2.4}$$

where E' is the relative velocity changes with respect to E_0. In the monitoring system, the estimated relative velocity change $dV/V' = -E'$ was used in both the ARM and SRM.

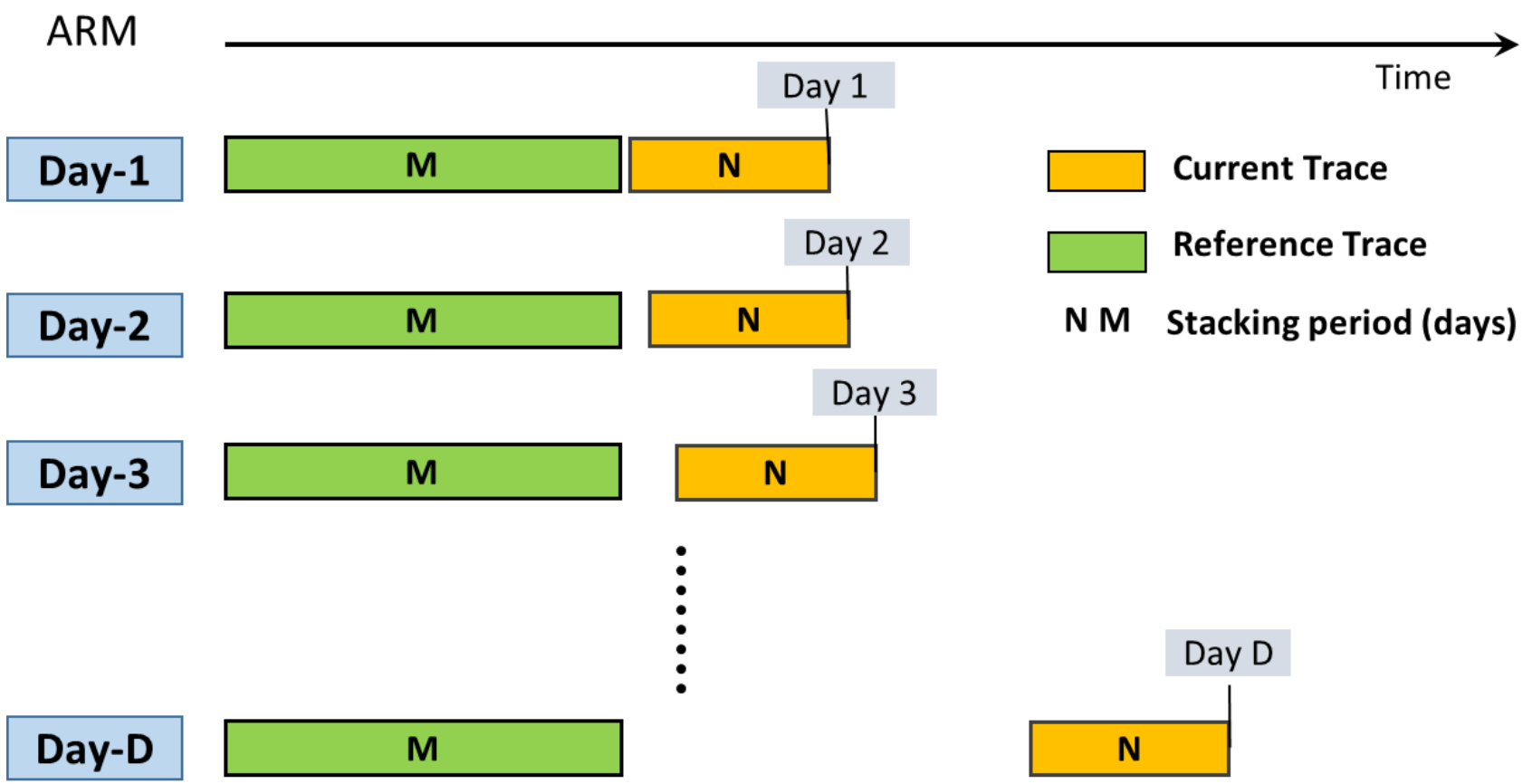

Figure 2.3 Schematic representation of estimating temporal velocity variation by ARM. N and M are the stacking periods, in days, for the current and reference trace, respectively. data period for the new current trace is moved by 1 day while the data period of the reference trace is fixed, then stretching interpolation is applied. No overlapping between current and reference trace.

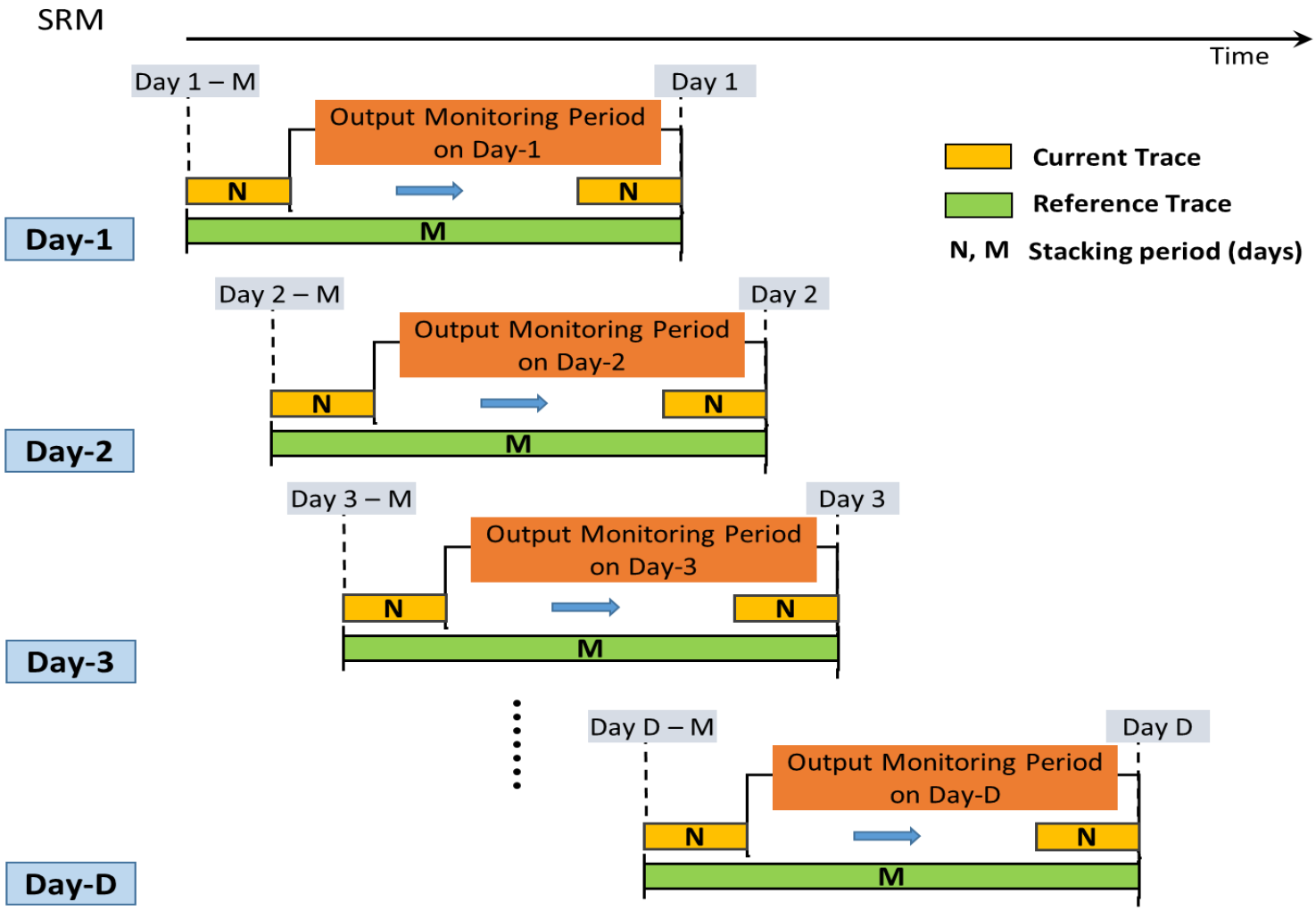

Figure 2.4 Schematic representation of estimating temporal velocity variation by SRM. N and M are the stacking periods, in days, for the current and reference trace, respectively. Data periods for the reference and current traces are both moved by 1 day, then stretching interpolation is applied for the entire length of the reference trace. The current trace data always overlap with the reference trace

2.2.3.2 Detect and remove outliers

When the value of $C(E)$ is low, the corresponding temporal velocity change is usually unstable. To remove these outliers, all temporal velocity changes that have $C(E)$ below a threshold value were marked. This study used 0.5 as the threshold because a cross-correlation coefficient smaller than 0.5 indicates a weak relationship between two variables, in this case the reference and current traces (Asuero et al. 2006; Mukaka 2012). This study also marked velocity changes if there were multiple peaks in $C(E)$. The outliers were removed using MAD algorithm and a median filter. The range of acceptable data was defined in the following equations:

$$MAD = Median_i(|s_i - Median_j(s_j)|), \quad (2.5)$$

$$Median_j(s_j) - TC * MAD < s_i < Median_j(s_j) + TC * MAD, \quad (2.6)$$

where s_i is the i-th velocity change in the time series of velocity changes at a station pair, and TC is the tolerance coefficient for MAD. After several trial an error, this study decided to use $TC = 3$. Velocity changes that were outside the MAD range were removed. Furthermore, a median filter was applied with a time window of 3 days. The procedure is diagrammed in Figure 2.5.

In the monitoring results derived from SRM (Figure 2.6), stable spatio-temporal velocity variations of the whole Japanese Islands can be observed, and identify velocity variations due to the earthquake, volcanic eruptions and weathering (and tiding), as discussed later. The next section describes how to optimized parameters to define high-temporal resolution (i.e., how many days of data to create current traces).

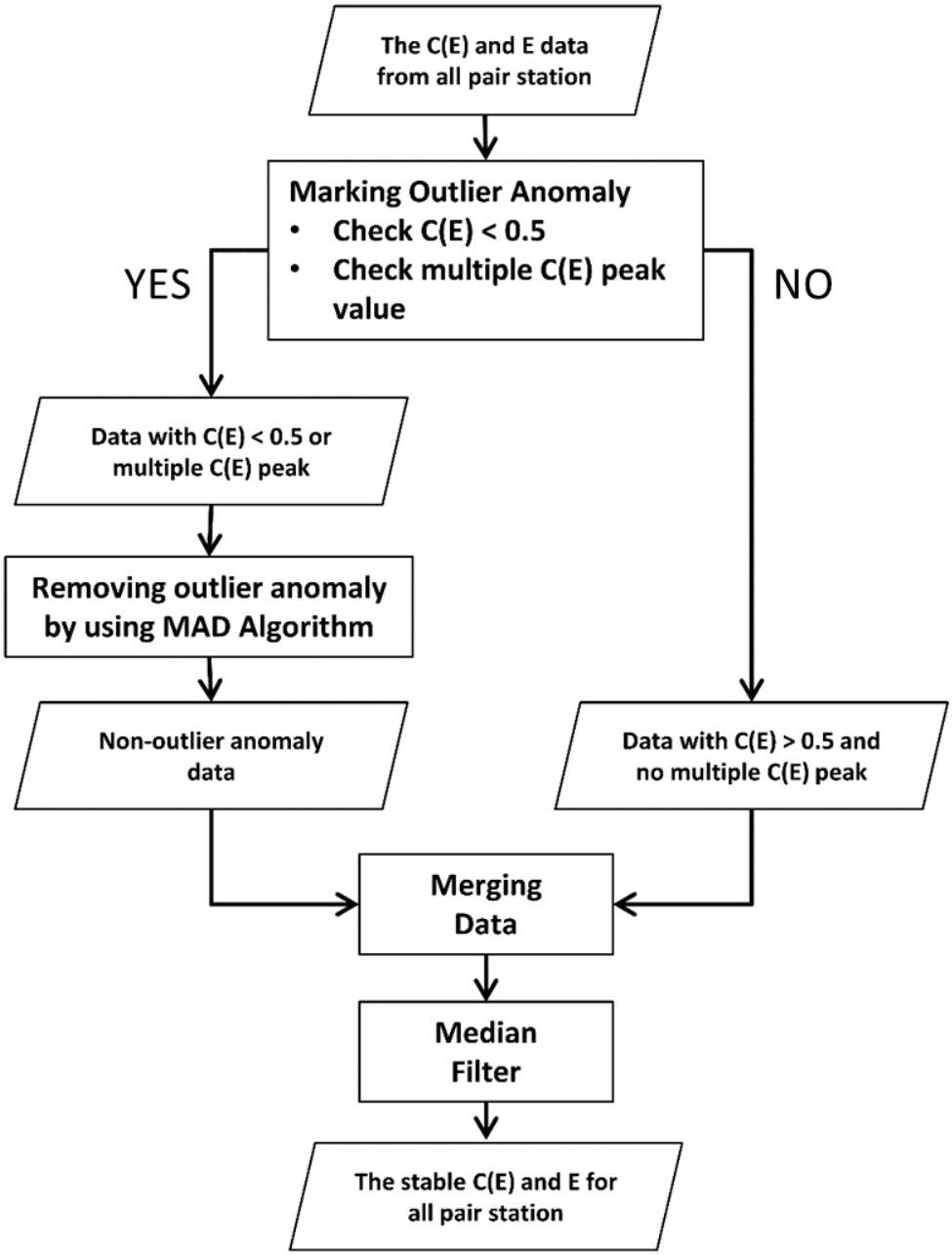

Figure 2.5 Flow chart for detection and removal of unstable temporal velocity variations (outliers). Input data consist of the stretching coefficient C(E) and the temporal velocity changes E for each pair of stations. Given the threshold of C(E) and the tolerance coefficient TC of MAD, this step can estimate the range of acceptable velocity variations by using Eqs. 2.5 and 2.6. If C(E) is outside that range, the velocity change is rejected. The final step is applying a median filter to obtain stable E values.

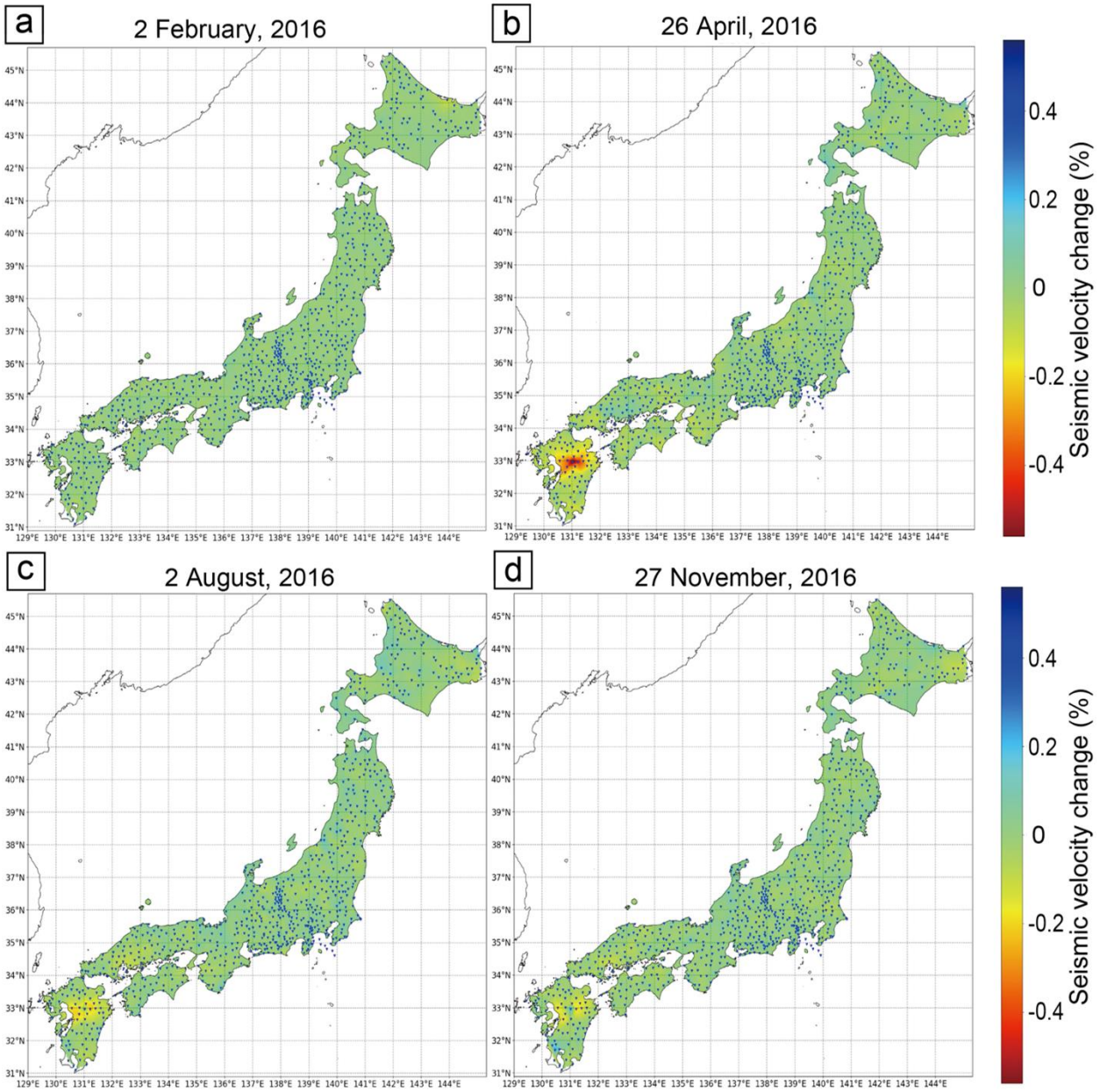

Figure 2.6 Spatio-temporal variation of seismic velocity of whole Japanese Islands. This velocity variation was derived by applying SRM to ambient-noise data recorded by 767 Hi-net stations (blue symbols). Each panel shows the latest date within the window of current trace: a) 2 February 2016, b) 26 April 2016, c) 2 August 2016, and d) 27 November 2016. Warm colors indicate regions where seismic velocity was decreased. The Kumamoto earthquakes (Mw. 7) occurred on 16 April, and the Mt Aso eruptions occurred on 7 and 8 October

2.3 Optimum parameters for Absolute Reference Method and Sliding Reference Method

There is a trade-off between stability and temporal resolution, depending on how many days of data are stacked for current traces and how reference traces are defined (i.e., ARM or SRM). To find optimum parameters for defining current traces, the stability of the stretching interpolation result was assessed by evaluating different time windows of current traces for ARM and SRM. *C(E)* for the estimated velocity change (Eq. 2.2) is commonly used for this stability evaluation (e.g., Budi-Santoso and Lesage 2016; Yukutake et al. 2016). The stability was evaluated according to the number of data points for which the correlation coefficients for the estimated velocity changes exceeded the threshold value (0.5 in this study). The input data for this evaluation consisted of the velocity variations before outlier removal. For the stability evaluation, this study used ambient-noise data (vertical component) recorded by Hi-net stations (~100 stations) in Kyushu from 2015 to 2016, a period including the 2016 Kumamoto earthquake, and tested different stacking periods to create the current trace. The results (Figure 2.7) showed that as expected, increasing the time window yielded more stable monitoring results for the ARM and SRM.

In the SRM, the incremental improvement from each added day was less than 5% when the current trace was stacked more than 11 days of data, therefore this study chose 11 days of stacking as the best compromise between stability and temporal resolution. Likewise, this study could choose 17 days of data to make the current trace for the ARM. The chosen time window depends on several parameters, such as the window size for reference traces (here 1 year),

seismometer density and station interval, ambient-noise characteristics, lithology, and the frequency range for analysis. Therefore, the stability evaluation (Figure 2.7) should be done for any application of the monitoring method. The result found that the seismic velocities clearly decreased (due to pore pressure changes and formation damage) around the hypocenter of the 2016 Kumamoto earthquake (Mw7) and Mt Aso eruption after the earthquake by using both the ARM (Figure 2.8a–e) and the SRM (Figure 2.8f–j). This study assumed that surface waves dominated the coda wave and that the surface-wave velocity was close to the S-wave velocity.

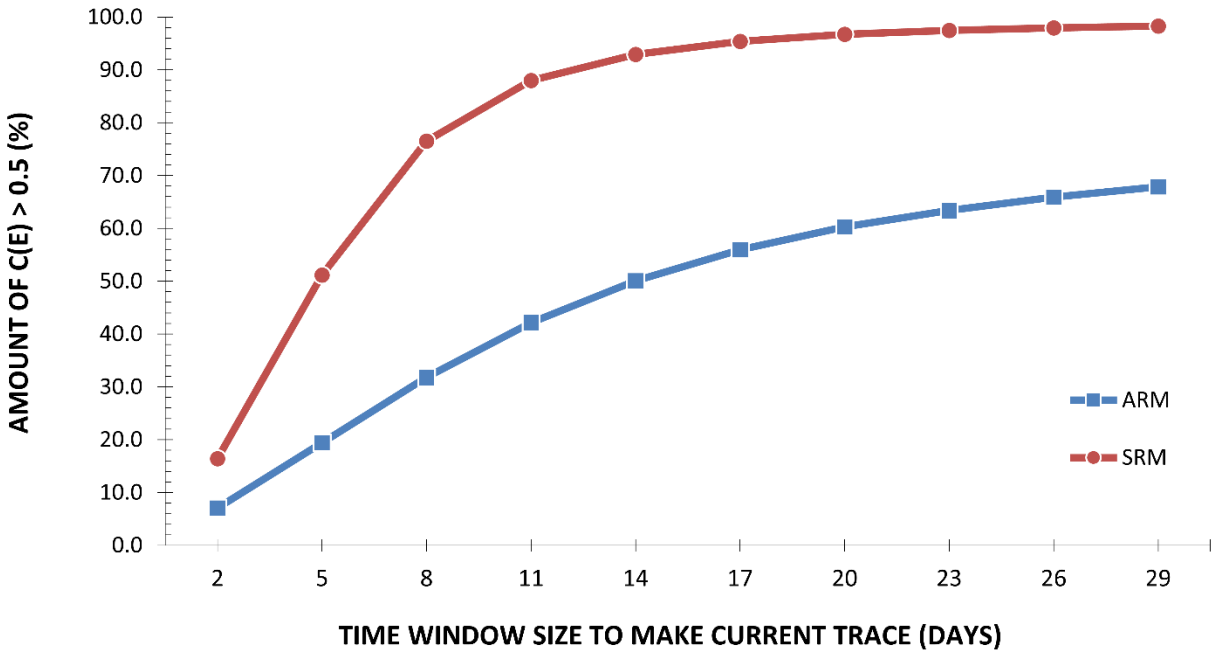

Figure 2.7 The result of stability evaluation for the ARM and SRM in Kyushu. The study period includes the 2016 Kumamoto earthquake. The time windows chosen to define the current trace are 17 and 11 days for the ARM and SRM, respectively

Because the frequency range was analyzed from 0.1 to 0.9 Hz hence the results were sensitive to S-wave velocity variations shallower than ~10 km depth (Nimiya et al. 2017). A

different choice of frequency range should reveal velocity variations for a different depth range. The characteristics of velocity variations differed in the results of the ARM and SRM (Figure 2.8). Thus, *C(E)* derived by both methods were mapped to evaluate the stability of the monitoring results (Figure 2.9). In the results using the ARM, *C(E)* suddenly decreased at the time of the earthquake, particularly around the hypocenter and Aso volcano (Figure 2.9a, b).

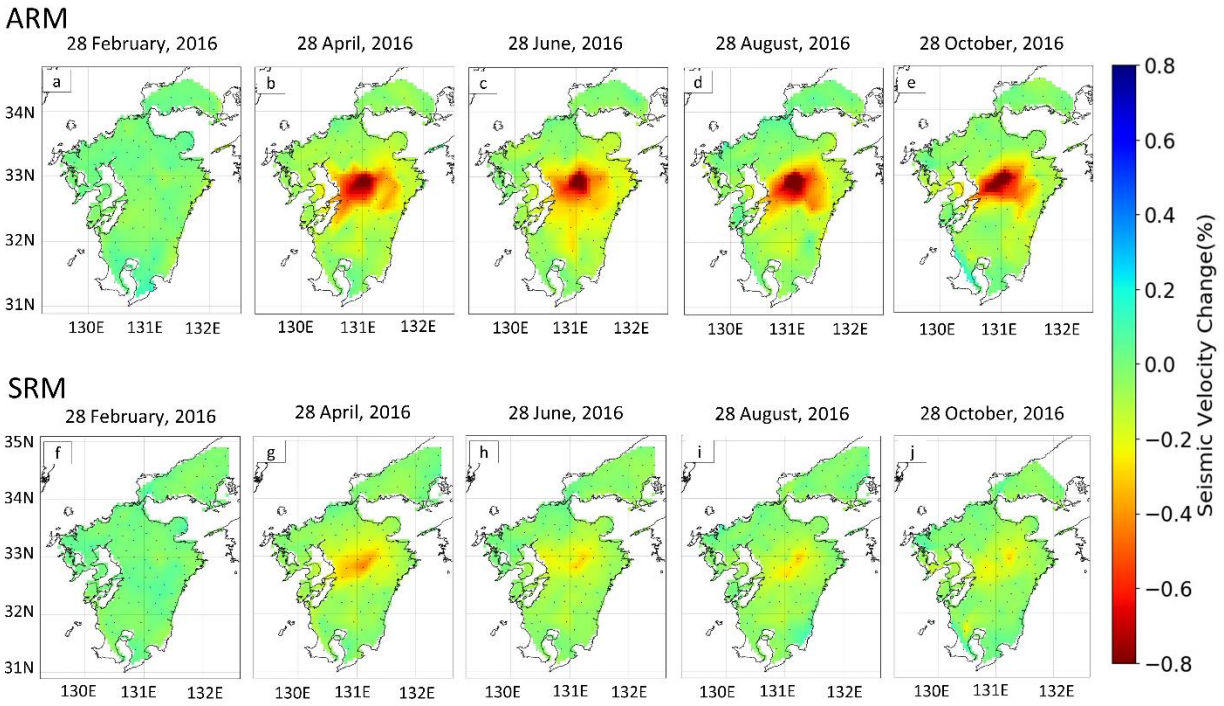

Figure 2.8 Spatio-temporal velocity maps determined by using a) to e) the ARM and f) to j) the SRM. The ARM and SRM have different temporal resolutions because the current trace was made with stacking periods of 17 and 11 days, respectively. Both methods identified spatio-temporal velocity changes caused by the Kumamoto earthquake. But the velocity changes derived from both methods are largely different

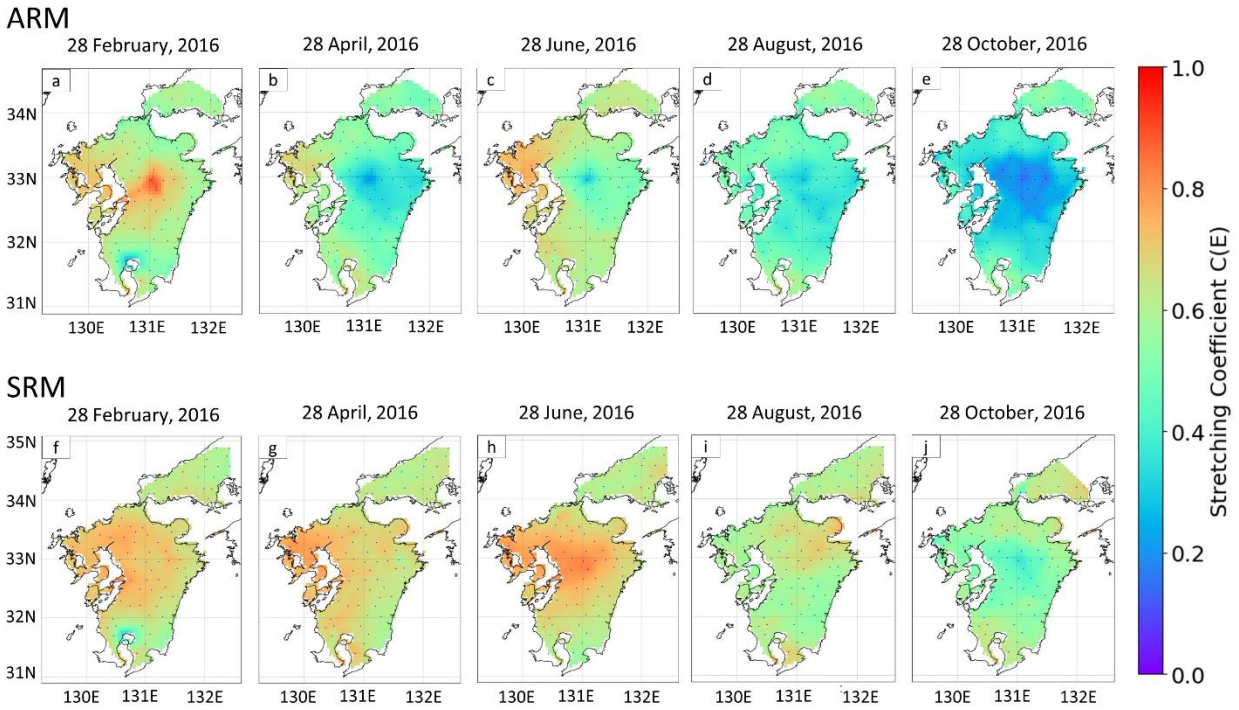

Figure 2.9 Stretching coefficient C(E) maps determined by using a) to e) the ARM and f) to j) the SRM. The SRM results show higher C(E) values than the ARM results because in the SRM the current trace always overlaps the reference trace. Panels d) and e) show that C(E) decreased because the current trace has continued to separate in time from the fixed reference trace and also is affected by multiple eruptions, such that the velocity variation could become unstable. The ARM appears to be useful to detect changes in the scatter wave, whereas the SRM determines velocity variations with high stability and high temporal resolution

This indicates that the characteristics of the coda wave changed after the earthquake, probably the effect of a change in scattering due to the earthquake (Obermann et al. 2013) and change in ambient-noise characteristics. The *C(E)* value for the ARM significantly decreased in October 2016 (Figure 2.9e), which could be related to multiple eruptions occurred at Aso volcano from mid-September 2016 to mid-November 2016. On the other hand, in the SRM results, *C(E)* remained high even after the earthquake and eruption (Figure 2.9f–j) because the sliding reference period included time after their occurrence. These results indicate that the velocity variations

derived from the ARM are less accurate when the current trace is affected by dynamic events (volcanic activity and a change in scattering) and when the current trace date is too far separated from the reference trace. The advantage of the SRM is that the reference trace always reflects geological conditions at times close to the current trace, and this method can detect velocity variations with greater stability (Figure 2.9) and higher temporal resolution (Figure 2.7). The weakness of the SRM is the limited period of velocity variation results (no more than *M - (N+ 1)* days; 1 year in present case) due to its intensive computation. Given these considerations, this study decided to use the SRM for monitoring daily velocity variations, using an 11-day window for the current trace and a 365-day window for the reference trace (Figure 2.6).

In order to analyze the error using the SRM, temporal velocity variations were calculated by using shorter time window for coda waves (50 s) and shifting the starting time of the window every 10 s interval (Meier et al. 2010; Nimiya et al. 2017). Using those velocity changes, the standard deviation was estimated for each station pair. Then, the error for each station was estimated using the standard deviations between all pairs that included a station within a distance of 40 km as follows:

$$\sigma = \sqrt{\frac{1}{N}\frac{\sum_{i=1} n_i \sigma_i^2}{\sum_{i=1} n_i}}, \tag{2.7}$$

where σ_i and n_i are the standard deviation and the number of measurements (maximum is 6) for the ith station pair, respectively, and N is the number of station pairs. When only one measurement data is accepted for a station pair after detection and removal of outlier data, the pair do not include in the error evaluation (Eq. (2.7)) because the standard deviation cannot define. In the error map using the conventional approach (without detection and removal of outlier data) (Figure 2.10 a-e), large values of errors were locally observed (~0.2 %), which are comparable to

velocity variation due to the earthquake. On the other hand, by considering outlier data using MAD and a median filter, the results did not observe such larger values of errors and most of the errors are smaller than 0.02 % (Figure 2.10 f-j), indicating high stability of the monitoring results.

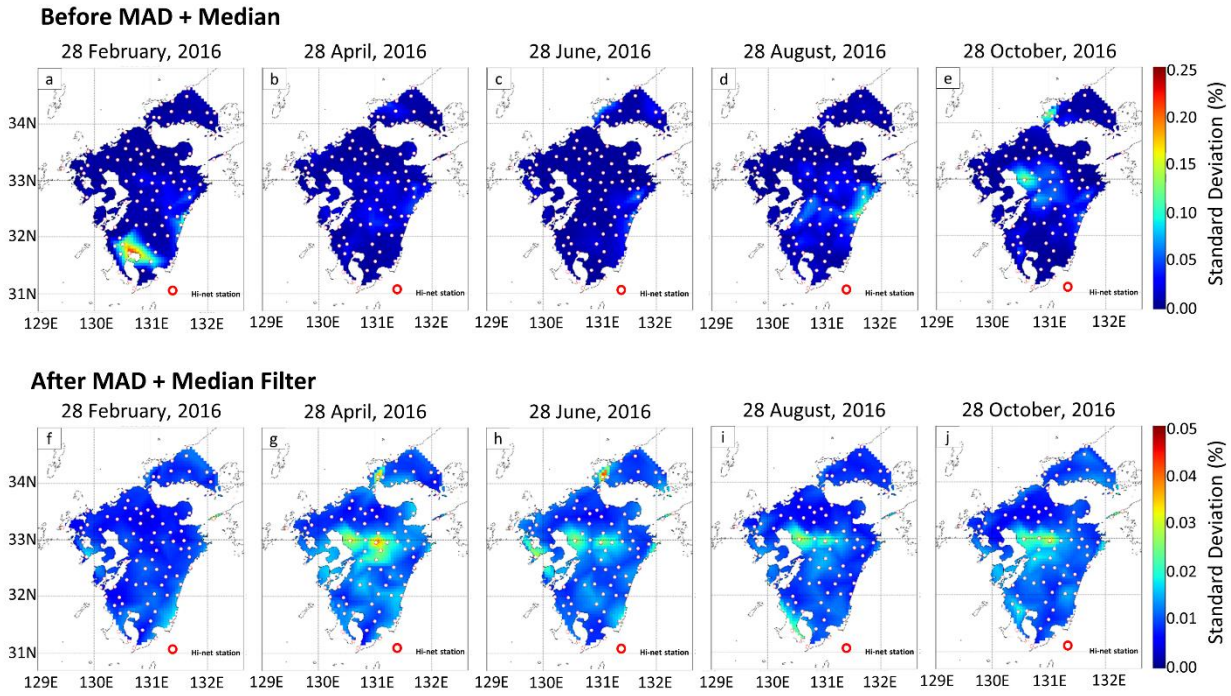

Figure 2.10 Error maps determined by using the SRM. a) to e) Without MAD and a median filter and f–j with MAD and a median filter. Note that color scales for panels a) to e) and f) to j) are different. By considering outlier data using the approach, the errors of velocity variations can be significantly suppressed

2.4 Developing an automated monitoring system

To realize daily monitoring using SRM, an effective system needs to be considered duo to the intensive computation. The continuous monitoring system was described in terms of its three

main tasks: downloading ambient-noise data from the NIED server, processing the ambient-noise data to estimate spatio-temporal velocity variations, and displaying the results in a web application on a virtual private server (Figure 2.11). The governing equations for the processing steps are presented in section 2. This system used CentOS 6.10-x64 and Red Hat 6.4-x64 as the main operating system platform to build and test the system performance. Python 3.6.4 from the Anaconda Distribution was used to ensure that this monitoring system works on various Linux platforms. NumPy library (Van der Walt et al. 2011) was used as the input-output data format for processing the ambient-noise data. The python codes for building this monitoring system is described on the Appendix section.

2.4.1 Update spatio-temporal velocity variation

To update spatio-temporal velocity changes automatically, the previous day of ambient-noise data (in win32 format) was downloaded from the NIED server every midnight. Before starting the processing step, this system checked whether the number of downloaded files is complete or not. The analysis of the ambient-noise data then begins (Figure 2.1 and 2.11). After obtaining both temporal and spatio-temporal velocity variations, the results are transformed into a tabular data format by NumPy. A map of velocity variations is compiled by linearly interpolating the spatio-temporal velocity variations at each station and then converting the results into a grid data format by using Netcdf4 (Unidata 2015). These files are used as input data to make objects for the web application. The web application is updated by a subroutine that checks the

modification time of the input files and restarts the application whenever the input files are updated. A crontab module in the Linux operating system automatically starts all of these processes daily.

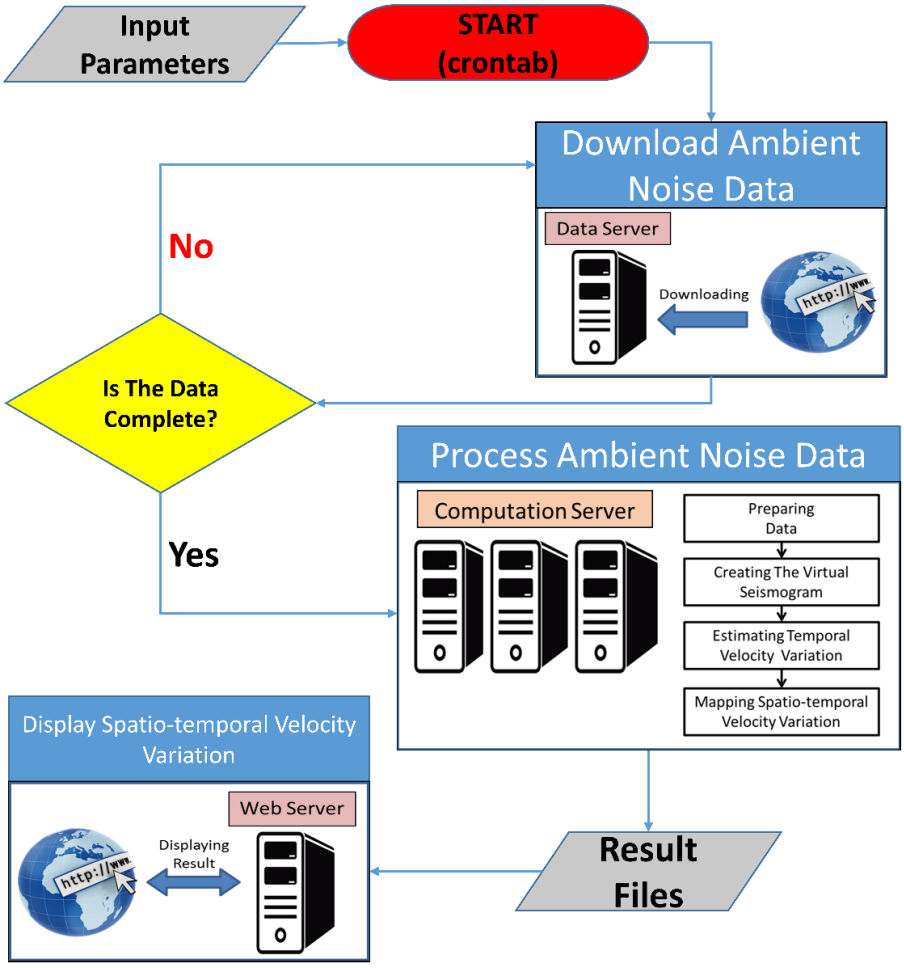

Figure 2.11 Schematic diagram of the ambient-noise monitoring system. The system comprises three operations: downloading ambient-noise data, processing the data, and displaying the spatio-temporal velocity variations on a website. A crontab module starts the update cycle every day at midnight. Presently the monitoring results has been updated every week on the website

2.4.2 Parallel computation design

The parallel computation architecture was adopted using a shared memory model and hybrid distributed memory (or cluster) model (Barney 2018). The shared memory model used multi-core central processing unit (CPU) computation inside a single node of workstations, and the hybrid distributed memory model used multi-core CPU and several nodes of workstations connected through a network. Python's Multiprocessing module (Python Software Foundation 2017) was used to handle the shared memory model and the combination of Multiprocessing and MPI4PY (Dalcin et al. 2011) modules to handle the hybrid distributed memory model. The ambient-noise processing system consisted of parent, child, and storage nodes. The storage for the parent and child nodes was connected to the storage node through the network file system. During computation in the hybrid distributed memory model, only the child node (rank $\neq 0$) ran the computation job and the input–output data were in the storage node. In terms of Flynn's taxonomy (Flynn 1972), the monitoring system can be categorized as single instruction stream, multiple data streams (SIMD) because ambient-noise data can be processed independently by every seismometer station or every station pair. Although this system could use both the shared memory and hybrid distributed memory models in all processing steps in Figure 2.1, this system did not use the latter in mapping velocity changes because this processing step is faster than the others. During the computation process, the volume of data loaded caused computation crashes when it exceeded the available RAM. To prevent this problem, there is a subroutine to divide the input data into portions smaller than 70% of the free RAM before starting the computation

To benchmark the computation performance, this system used a cluster system with eight server nodes (one parent and seven child nodes) and estimated the velocity variation using the SRM. this study recorded the total processing time for a system that used hybrid distributed memory parallelization (cluster parallelization) for preparing data, creating virtual seismograms, and estimating temporal velocity variations. The computation process ran only on the child nodes. In theory, this system would yield a sevenfold increase in processing speed, but because the hardware specification of every node was not uniform (the CPU clocks and memory sizes were not uniform), the level of performance could not be attained. Table 1 lists the computation performance for different numbers of child nodes. Using all seven child nodes led to the fastest computation time (33.3 minutes), whereas using the fastest single child node (with dual Intel Xeon E5-2680 CPUs and 96 GB RAM) required 202.4 minutes. The use of parallel computation thus sped up processing by 6.09 times and consumed almost 87% of the cluster nodes resource.

Table 2.1 Parallel computation performance by using hybrid-distribution memory model (cluster) parallelization.

Number of Child Node	Computation Time (s)	Speed-up
1	12143.92	1.00
2	6217.99	1.95
3	4296.75	2.83
4	3296.37	3.68
5	2781.27	4.37
6	2316.71	5.24
7	1995.14	6.09

Although the monitoring scheme using the SRM is computationally expensive, the high efficiency of the designed system enables us to update monitoring results daily with different processing parameters (e.g., different stacking windows). The monitoring system has flexibility in computation environments; thus, this system could be used in cloud computing.

2.4.3 Display results on website

To make daily monitoring results available to the public, a web application was developed by using Holoviews, Geoviews, Panes (Stevens et al. 2015), and Bokeh (Bokeh Development Team 2018). For the data structure, Pandas was used to manage the tabular data (McKinney 2010) and X-array (Hoyer and Hamman 2017) to manage the grid data file. The application produces three main products: a map of the spatio-temporal velocity variations derived from the SRM (Figure 2.12a), the temporal velocity variations for all station pairs (Figure 2.12b), and a graph of spatio-temporal velocity variations for each station (Figure 2.12c). Some interaction tools were included such as zoom, hover tools, pan, and selecting box (Figure 2.12). The newest update for spatio-temporal velocity variation for Kyushu area is shown on http://geo.mine.kyushu-u.ac.jp/tsuji/monitoring.html. There is a room for continued improvement in the design and content of the website.

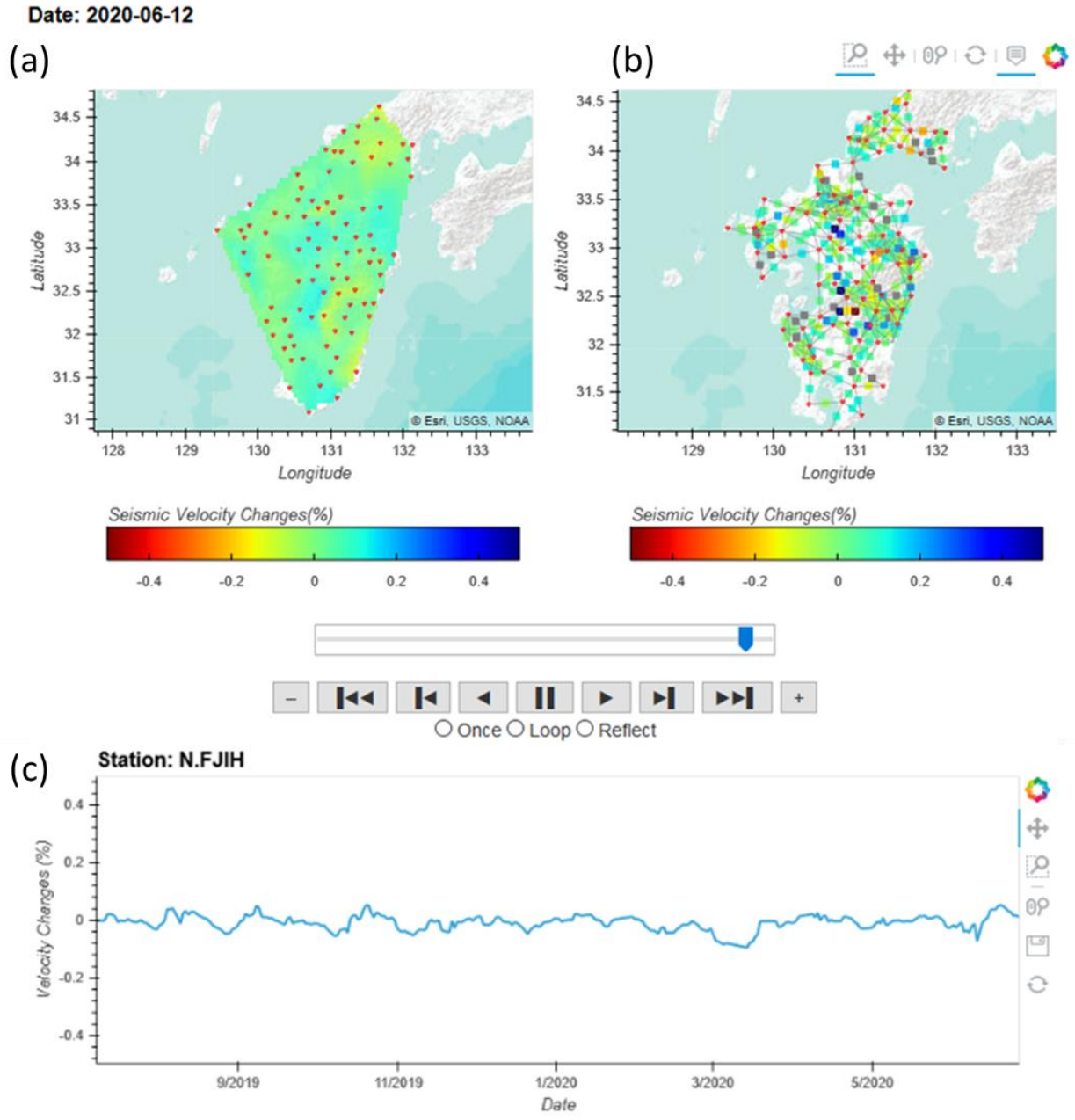

Figure 2.12 Template of the web application. a) Spatio-temporal velocity changes in Kyushu; b) temporal velocity variations for all station pairs; c) Spatio-temporal velocity variations for individual stations, and d attributes of the data used to calculate velocity variations. This template of web application design is not fixed, because there is a room to update the design in the future in order to improve the feature and performance

2.5 Discussion

This study developed a continuous spatio-temporal S-wave velocity monitoring system, based on cross-correlation of ambient-noise data from seismic stations, and performance during 2015 and 2016 was reported, a time period that included the Kumamoto earthquake (Mw 7.0) and eruption of Aso volcano (Figure. 2.6 and 2.8). Because the S-wave velocity reflects the shear modulus of the geological formation, the velocity decrease that detected near the earthquake hypocenter and Aso volcano could have been caused by formation damage, increased pore pressure, or pressurized volcanic fluids (Nimiya et al. 2017). In areas far from the hypocenter, velocity variations could be caused by pore pressure changes due to earthquake vibrations. Increased pore pressure could reduce effective stress as well as the friction on the seismogenic fault (Tsuji et al. 2014), thus the velocity variation could be related to the aftershock sequence. Moreover, before volcanic eruptions, increased pore pressure associated with the intrusion of magma (e.g., Budi-Santoso and Lesage 2016; Obermann et al. 2013) could decrease seismic velocity. Indeed, a decrease in velocity was detected before the Aso eruption and a velocity increase after the eruption, consistent with these pore pressure changes (Figure. 2.6 and 2.8).In addition, the monitoring result (Figure 2.6) contains signals of other processes that may affect earthquake generation. Using ambient-noise monitoring, the result can identify the velocity variation due to rain precipitation and snowfall (Wang et al. 2017), because the overburden or diffusion change the pressure conditions of the crust. The tides and snowfall are known to trigger small earthquakes (e.g., Heki 2003; Ide et al. 2016) through their effect on pressure and stress

conditions around seismogenic faults. In Kyushu Island, the influence of snowing can be neglected due to little snow-fall. However, this result also identified the velocity variation caused by rain precipitation and ocean tiding (Wang et al. 2017; Figure 2.6). Since our monitoring is related to S-wave, the maximum depth of monitoring result can be approximated. In S-wave analysis, there is a rule of thumb that said the length of receiver pair at least equal to the maximum desired wavelength, which corresponds to more than three times of the desired investigation depth (Foti et al., 2018). In this monitoring result, the pair distance is less than 40 km, so the depth monitoring result is less than ~13 km. Moreover, the depth of velocity changes can be estimated by using the frequency-dependent depth sensitivity of S-wave (Mainsant et al., 2012; Rivet et al., 2011). Our results using the frequency range of 0.1 to 0.9 Hz were sensitive to S-wave velocity variations from the surface to a depth of ~10 km (Nimiya et al., 2017). Therefore, the monitoring system can detect dynamic crustal behaviors associated with earthquake triggering processes on a daily basis that may inform disaster warnings and related applications.

2.6 Summary

The research summary of crustal real-time monitoring with high temporal resolution based on spatio-temporal monitoring system are described bellows:

1. The high temporal continuous monitoring system of the whole Japanese Islands was developed. Geodetical approaches such as Global Navigation Satellite System (GNSS) or Interferometric Synthetic Aperture Radar (InSAR) are typical to identify dynamic crustal behavior, but the monitoring system uses the wave propagating within the crust.

2. The developed monitoring system includes a module to automatically remove unstable velocity results by using the Median Absolute Deviation algorithm and a median filter.
3. This study compared the Absolute Reference Method and Sliding Reference Method for treatment of reference and current traces for stretching interpolation. Because the Sliding Reference Method generated a reference trace that reflects geological conditions close to the current trace, the Sliding Reference Method produced results with greater stability and higher temporal resolution due to high *C(E)* result.
4. To find optimum high temporal resolution parameters for defining current traces, the stability of the stretching interpolation results was assessed by trying different time windows for current traces. On the basis of cross-correlation coefficient *C*(*E*), this study found that an 11-day stacking window offered the best compromise between stability and temporal resolution in the Sliding Reference Method.
5. The system used parallel computation on seven server nodes to calculate spatio-temporal velocity variations, achieving a greater than six-fold gain in computation speed. Therefore, this system can be used to continuously monitor seismic velocity using ambient-noise data.
6. A web application was developed to enable public access to spatio-temporal velocity variations detected by the monitoring network. The monitoring results can be seen from the following site; http://geo.mine.kyushu-u.ac.jp/tsuji/monitoring.html.

2.7 References

Asuero, A.G., Sayago, A., González, A.G., 2006. The correlation coefficient: An overview. Crit. Rev. Anal. Chem. https://doi.org/10.1080/10408340500526766

Barney, B., 2018. Introduction to Parallel Computing [WWW Document]. Lawrence Livermore Natl. Lab. URL https://computing.llnl.gov/tutorials/parallel_comp/

Bensen, G.D., Ritzwoller, M.H., Barmin, M.P., Levshin, A.L., Lin, F., Moschetti, M.P., Shapiro, N.M., Yang, Y., 2007. Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements. Geophys. J. Int. 169, 1239–1260. https://doi.org/10.1111/j.1365-246X.2007.03374.x

Bokeh Development Team, 2018. Bokeh: Python Library for Interactive Visualization [WWW Document]. URL https://bokeh.pydata.org/en/latest

Brenguier, F., Campillo, M., Takeda, T., Aoki, Y., Shapiro, N.M., Briand, X., Emoto, K., Miyake, H., 2014. Mapping pressurized volcanic fluids from induced crustal seismic velocity drops. Science (80-.). https://doi.org/10.1126/science.1254073

Brenguier, F., Shapiro, N.M., Campillo, M., Ferrazzini, V., Duputel, Z., Coutant, O., Nercessian, A., 2008. Towards forecasting volcanic eruptions using seismic noise. Nat. Geosci. 1, 126–130. https://doi.org/10.1038/ngeo104

Budi-Santoso, A., Lesage, P., 2016. Velocity variations associated with the large 2010 eruption ofMerapi volcano, Java, retrieved from seismic multiplets and ambient noise cross-

correlation. Geophys. J. Int. https://doi.org/10.1093/gji/ggw145

Chadwick, A., Williams, G., Delepine, N., Clochard, V., Labat, K., Sturton, S., Buddensiek, M.L., Dillen, M., Nickel, M., Lima, A.L., Arts, R., Neele, F., Rossi, G., 2010. Quantitative analysis of time-lapse seismic monitoring data at the Sleipner CO2 storage operation. Lead. Edge (Tulsa, OK). https://doi.org/10.1190/1.3304820

Chaves, E.J., Schwartz, S.Y., 2016. Monitoring transient changes within overpressured regions of subduction zones using ambient seismic noise. Sci. Adv. 2, 1–7. https://doi.org/10.1126/sciadv.1501289

Clarke, D., Zaccarelli, L., Shapiro, N.M., Brenguier, F., 2011. Assessment of resolution and accuracy of the Moving Window Cross Spectral technique for monitoring crustal temporal variations using ambient seismic noise. Geophys. J. Int. 186, 867–882. https://doi.org/10.1111/j.1365-246X.2011.05074.x

Dalcin, L.D., Paz, R.R., Kler, P.A., Cosimo, A., 2011. Parallel distributed computing using Python. Adv. Water Resour. https://doi.org/10.1016/j.advwatres.2011.04.013

DeVries, P.M.R., Viégas, F., Wattenberg, M., Meade, B.J., 2018. Deep learning of aftershock patterns following large earthquakes. Nature. https://doi.org/10.1038/s41586-018-0438-y

Flynn, M.J., 1972. Some computer organizations and their effectiveness. IEEE Trans. Comput. https://doi.org/10.1109/TC.1972.5009071

Foti, S., Hollender, F., Garofalo, F., Albarello, D., Asten, M., Bard, P.Y., Comina, C., Cornou, C., Cox, B., Di Giulio, G., Forbriger, T., Hayashi, K., Lunedei, E., Martin, A., Mercerat, D., Ohrnberger, M., Poggi, V., Renalier, F., Sicilia, D., Socco, V., 2018. Guidelines for the good

practice of surface wave analysis: a product of the InterPACIFIC project, Bulletin of Earthquake Engineering. https://doi.org/10.1007/s10518-017-0206-7

Heki, K., 2003. Snow load and seasonal variation of earthquake occurrence in Japan. Earth Planet. Sci. Lett. https://doi.org/10.1016/S0012-821X(02)01148-2

Hendriyana, A., Tsuji, T., 2019. Migration of Very Long Period Seismicity at Aso Volcano, Japan, Associated With the 2016 Kumamoto Earthquake. Geophys. Res. Lett. https://doi.org/10.1029/2019GL082645

Hobiger, M., Wegler, U., Shiomi, K., Nakahara, H., 2016. Coseismic and post-seismic velocity changes detected by passive image interferometry: Comparison of one great and five strong earthquakes in Japan. Geophys. J. Int. 205, 1053–1073. https://doi.org/10.1093/gji/ggw066

Hoyer, S., Hamman, J.J., 2017. xarray: N-D labeled Arrays and Datasets in Python. J. Open Res. Softw. https://doi.org/10.5334/jors.148

Hutapea, F.L., Nimiya, H., Ikeda, T., Tsuji, T., 2019. Evaluation of Optimal Processing Parameters for Automatic Continuous Monitoring Using Ambient Noise. https://doi.org/10.1190/segj2018-075.1

Ide, S., Yabe, S., Tanaka, Y., 2016. Earthquake potential revealed by tidal influence on earthquake size-frequency statistics. Nat. Geosci. https://doi.org/10.1038/ngeo2796

Ikeda, T., Tsuji, T., 2018. Temporal change in seismic velocity associated with an offshore MW 5.9 Off-Mie earthquake in the Nankai subduction zone from ambient noise cross-correlation. Prog. Earth Planet. Sci. 5. https://doi.org/10.1186/s40645-018-0211-8

Ikeda, T., Tsuji, T., Takanashi, M., Kurosawa, I., Nakatsukasa, M., Kato, A., Worth, K., White, D., Roberts, B., 2017. Temporal variation of the shallow subsurface at the Aquistore CO2 storage site associated with environmental influences using a continuous and controlled seismic source. J. Geophys. Res. Solid Earth. https://doi.org/10.1002/2016JB013691

Ikuta, R., Yamaoka, K., 2004. Temporal variation in the shear wave anisotropy detected using the Accurately Controlled Routinely Operated Signal System (ACROSS). J. Geophys. Res. Solid Earth. https://doi.org/10.1029/2003JB002901

Ikuta, R., Yamaoka, K., Miyakawa, K., Kunitomo, T., Kumazawa, M., 2002. Continuous monitoring of propagation velocity of seismic wave using ACROSS. Geophys. Res. Lett. 29, 3–6. https://doi.org/10.1029/2001GL013974

Kaneda, Y., Kawaguchi, K., Araki, E., Matsumoto, H., Nakamura, T., Kamiya, S., Ariyoshi, K., Hori, T., Baba, T., Takahashi, N., 2015. Development and application of an advanced ocean floor network system for megathrust earthquakes and tsunamis, in: Seafloor Observatories: A New Vision of the Earth from the Abyss. https://doi.org/10.1007/978-3-642-11374-1_25

Kawaguchi, K., Kaneko, S., Nishida, T., Komine, T., 2015. Construction of the DONET real-time seafloor observatory for earthquakes and tsunami monitoring, in: Seafloor Observatories: A New Vision of the Earth from the Abyss. https://doi.org/10.1007/978-3-642-11374-1_10

Kawakatsu, H., Kaneshima, S., Matsubayashi, H., Ohminato, T., Sudo, Y., Tsutsui, T., Uhira, K., Yamasato, H., Ito, H., Legrand, D., 2000. Aso94: Aso seismic observation with broadband instruments. J. Volcanol. Geotherm. Res. https://doi.org/10.1016/S0377-0273(00)00166-

9

Kumazawa, M., Takei, Y., 1994. No Active method of monitoring underground structures by means of accurately controlled rotary seismic source (ACROSS). 1. Purpose and principle. Title, in: Fall Meeting of the Seismological Society of Japan. p. 158.

Leys, C., Ley, C., Klein, O., Bernard, P., Licata, L., 2013. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. J. Exp. Soc. Psychol. 49, 764–766. https://doi.org/10.1016/j.jesp.2013.03.013

Maeda, Y., Yamaoka, K., Miyamachi, H., Watanabe, T., Kunitomo, T., Ikuta, R., Yakiwara, H., Iguchi, M., 2015. A subsurface structure change associated with the eruptive activity at Sakurajima Volcano, Japan, inferred from an accurately controlled source. Geophys. Res. Lett. https://doi.org/10.1002/2015GL064351

Mainsant, G., Larose, E., Brnnimann, C., Jongmans, D., Michoud, C., Jaboyedoff, M., 2012. Ambient seismic noise monitoring of a clay landslide: Toward failure prediction. J. Geophys. Res. Earth Surf. https://doi.org/10.1029/2011JF002159

McKinney, W., 2010. Data Structures for Statistical Computing in Python. Proc. 9th Python Sci. Conf.

Meier, U., Shapiro, N.M., Brenguier, F., 2010. Detecting seasonal variations in seismic velocities within Los Angeles basin from correlations of ambient seismic noise. Geophys. J. Int. 181, 985–996. https://doi.org/10.1111/j.1365-246X.2010.04550.x

Minato, S., Tsuji, T., Ohmi, S., Matsuoka, T., 2012. Monitoring seismic velocity change caused by the 2011 Tohoku-oki earthquake using ambient noise records. Geophys. Res. Lett. 39,

n/a-n/a. https://doi.org/10.1029/2012GL051405

Mukaka, M.M., 2012. Statistics corner: A guide to appropriate use of correlation coefficient in medical research. Malawi Med. J.

Nagaoka, Y., Nishida, K., Aoki, Y., Takeo, M., 2010. Temporal change of phase velocity beneath Mt. Asama, Japan, inferred from coda wave interferometry. Geophys. Res. Lett. 37, 1–5. https://doi.org/10.1029/2010GL045289

Nakata, N., Snieder, R., Tsuji, T., Larner, K., Matsuoka, T., 2011. Shear wave imaging from traffic noise using seismic interferometry by cross-coherence. Geophysics. https://doi.org/10.1190/geo2010-0188.1

NIED, 2019. NIED Hi-net. https://doi.org/https://doi.org/10.17598/NIED.0003

Nimiya, H., Ikeda, T., Tsuji, T., 2017. Spatial and temporal seismic velocity changes on Kyushu Island during the 2016 Kumamoto earthquake. Sci. Adv. 3. https://doi.org/10.1126/sciadv.1700813

Obermann, A., Planès, T., Larose, E., Campillo, M., 2013. Imaging preeruptive and coeruptive structural and mechanical changes of a volcano with ambient seismic noise. J. Geophys. Res. Solid Earth 118, 6285–6294. https://doi.org/10.1002/2013JB010399

Okada, Y., Kasahara, K., Hori, S., Obara, K., Sekiguchi, S., Fujiwara, H., Yamamoto, A., 2004. Recent progress of seismic observation networks in Japan - Hi-net, F-net, K-net and KiK-net -. Earth, Planets Sp. https://doi.org/10.1186/BF03353076

Olivier, G., Brenguier, F., De Wit, T., Lynch, R., 2017. Monitoring the stability of tailings dam walls with ambient seismic noise. Lead. Edge 36, 350a1-350a6.

https://doi.org/10.1190/tle36040350a1.1

Python Software Fondation, 2017. The Python Language Reference version 3.6 [WWW Document]. URL https://docs.python.org/3.6 (accessed 10.6.19).

Ratdomopurbo, A., Poupinet, G., 1995. Monitoring a temporal change of seismic velocity in a volcano: Application to the 1992 eruption of Mt. Merapi (Indonesia). Geophys. Res. Lett. https://doi.org/10.1029/95GL00302

Rivet, D., Campillo, M., Shapiro, N.M., Cruz-Atienza, V., Radiguet, M., Cotte, N., Kostoglodov, V., 2011. Seismic evidence of nonlinear crustal deformation during a large slow slip event in Mexico. Geophys. Res. Lett. 38, 3–7. https://doi.org/10.1029/2011GL047151

Stevens, J.-L., Rudiger, P., Bednar, J., 2015. HoloViews: Building Complex Visualizations Easily for Reproducible Science, in: Proceedings of the 14th Python in Science Conference. https://doi.org/10.25080/majora-7b98e3ed-00a

Taira, T., Avinash, N., Brenguier, F., Manga, M., 2018. Monitoring reservoir response to earthquakes and fluid extraction, salton sea geothermal field, California. Sci. Adv. https://doi.org/10.1126/sciadv.1701536

Taira, T., Brenguier, F., 2016. Response of hydrothermal system to stress transients at Lassen Volcanic Center, California, inferred from seismic interferometry with ambient noise 4. Seismology. Earth, Planets Sp. https://doi.org/10.1186/s40623-016-0538-6

Tsuji, S., Yamaoka, K., Ikuta, R., Kunitomo, T., Watanabe, T., Yoshida, Y., Katsumata, A., 2018. Secular and coseismic changes in S-wave velocity detected using ACROSS in the Tokai region. Earth, Planets Sp. https://doi.org/10.1186/s40623-018-0917-2

Tsuji, T., Ikeda, T., Johansen, T.A., Ole Ruud, B., 2016. Using seismic noise derived from fluid injection well for continuous reservoir monitoring. Interpretation. https://doi.org/10.1190/INT-2016-0019.1

Tsuji, T., Kamei, R., Pratt, R.G., 2014. Pore pressure distribution of a mega-splay fault system in the Nankai trough subduction zone: Insight into up-dip extent of the seismogenic zone. Earth Planet. Sci. Lett. https://doi.org/10.1016/j.epsl.2014.04.011

Unidata, 2015. Network Common Data Form (netCDF) version 4.3.3.1 [software]. https://doi.org/http://doi.org/10.5065/D6H70CW6

Van Brummelen, G., 2012. Heavenly mathematics: The forgotten art of spherical trigonometry, Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry. https://doi.org/10.1080/10848770.2015.1028011

Van der Walt, S., Colbert, S.C., Varoquaux, G., 2011. The NumPy Array: A Structure for Efficient Numerical Computation. Comput. Sci. Eng. 13, 22–30. https://doi.org/10.1109/MCSE.2011.37

Wang, Q.Y., Brenguier, F., Campillo, M., Lecointre, A., Takeda, T., Aoki, Y., 2017. Seasonal Crustal Seismic Velocity Changes Throughout Japan. J. Geophys. Res. Solid Earth. https://doi.org/10.1002/2017JB014307

Wapenaar, K., Draganov, D., Snieder, R., Campman, X., Verdel, A., 2010. Tutorial on seismic interferometry: Part 1 — Basic principles and applications. Geophysics 75, 75A195-75A209. https://doi.org/10.1190/1.3457445

Wegler, U., Nakahara, H., Sens-Schönfelder, C., Korn, M., Shiomi, K., 2009. Sudden drop of

seismic velocity after the 2004 Mw 6.6 mid-Niigata earthquake, Japan, observed with Passive Image Interferometry B06305. J. Geophys. Res. Solid Earth 114, 1–11. https://doi.org/10.1029/2008JB005869

Yamaoka, K., Kunitomo, T., Miyakawa, K., Kobayashi, K., Kumazawa, M., 2001. A trial for monitoring temporal variation of seismic velocity using an ACROSS system. Isl. Arc. https://doi.org/10.1111/j.1440-1738.2001.00332.x

Yukutake, Y., Ueno, T., Miyaoka, K., 2016. Determination of temporal changes in seismic velocity caused by volcanic activity in and around Hakone volcano, central Japan, using ambient seismic noise records. Prog. Earth Planet. Sci. https://doi.org/10.1186/s40645-016-0106-5

Zhan, Z., Tsai, V.C., Clayton, R.W., 2013. Spurious velocity changes caused by temporal variations in ambient noise frequency content. Geophys. J. Int. 194, 1574–1581. https://doi.org/10.1093/gji/ggt170

Chapter 3

Data processing and interpretation schemes for deeply-towed Autonomous Cable Seismic (ACS) for gas and hydrate exploration

Abstract

The deep-towed Autonomous Cable Seismic (ACS) system is a high-resolution acoustic survey system designed for use in deep-water environments. This system uses a high-frequency acoustic source and a multichannel receiver cable. A common problem in the analysis of deep-towed ACS data is the unstable positioning of the source and receivers due to ocean currents and seafloor bathymetry. Since high-frequency data acquisition with unstable source–receiver positions can cause destructive interference on final stack profile, correction of the unstable

source-receiver is a crucial issue. This study proposed a method to solve the unstable source–receiver position problem and thus to construct an accurate final stack profile. This study used deep-towed ACS data acquired in the Joetsu Basin in Niigata, Japan, where hydrocarbon features in the form of gas chimneys, gas hydrate, and free gas have been observed. Because sidelobes in the ACS source signature defocus the source wavelet and decrease the bandwidth frequency content, this study designed a filter to focus the source signature. Our approach considerably improved the quality of the final stacked profile. Even though depth information was not available for all receivers, the velocity semblance was well focused. Furthermore, focusing the source wavelet considerably increased the bandwidth frequency of the pre-stack data. Seismic attribute analysis was applied to the final stacked profile to identify the distributions of free gas and hydrate. The seismic attribute analysis of the ACS data demonstrated that free gas accumulations are characterized by low reflection amplitude and an unstable frequency component, and that hydrate close to the seafloor can be identified by its high reflection amplitude.

Keywords: deep-towed acoustic survey, seismic attribute, seismic data processing, source–receiver depth correction, gas and hydrate exploration

3.1 Introduction

High-frequency acoustic survey in marine area is used to obtain high-resolution image of shallow structures. The high-resolution shallow profiles can be used in several applications, such as evaluation of geohazard potential (Marsset et al. 2010; Riboulot et al. 2013), exploration for natural gas (Ross Chapman et al. 2002; Wood et al. 2008; Riboulot et al. 2013; Marsset et al. 2014, 2018), characterizing marine volcanic/hydrothermal systems (Bischoff et al. 2017; Angkasa et al. 2017), and finding deep-sea ore deposits (Romagnoli et al. 2018; Sacchi et al. 2019). A high-frequency source signal is more strongly attenuated (Sato et al., 2012) than those of conventional marine seismic acquisition systems that use air-gun arrays (i.e., low-frequency source). Deep-towed acoustic survey is a new data acquisition system that have been considered for deep-sea environments to reduce attenuation of the high-frequency source signal by locating both source and receivers close to the seafloor or targeted reflector. Recently, several examples of deep-towed acoustic survey systems have been developed such as the Deep Towed Acoustic Geophysical System (DTAGS) (Wood et al. 2008), Pasisar (Savoye et al. 1995) and SYstème SIsmique Fond (SYSIF) (Marsset et al. 2010, 2014, 2018). In Japan, a deep-towed acoustic survey system named the Autonomous Cable Seismic (ACS) system has been developed by the Research and Development Partnership for Next Generation Technology of Marine Resources Survey under JGI/J-MARES (Asakawa et al. 2017). The ACS system not only uses a high-frequency source, but also uses a multichannel hydrophone receiver cable (Figure 3.1a); hence, conventional seismic reflection processing methods can be applied. In this study, the ACS data acquired in the Joetsu Basin in Niigata, Japan, where a high potential of gas hydrate has been recognized (Joshima et al. 2009; Freire et al. 2011; JAMSTEC 2012a, b; Hachikubo et al. 2015; Yoneda et al. 2019). Based

on a Japanese government report (Ministriy Economic Trade and Industry (METI) 2016), there are numerous gas chimneys in the Sea of Japan that are associated with accumulations of gas hydrates and free gas. As future energy demand increases, these gases could be considered as alternative energy sources. Moreover, gas hydrate deposits have the potential to contribute to global climate change. Therefore, high-resolution geophysical exploration is important for identifying the locations of gas chimneys, free gas, and gas hydrates.

The main objective of multichannel seismic reflection data processing is to obtain a zero-offset seismic stack section where the receiver and source points coincide (Yilmaz, 2001). This study wants to achieve the same goal for processing the deep-towed ACS data. However, because of strong ocean currents and rough seafloor topography during data acquisition, it was difficult to control the positions of the source and receivers. High-frequency ACS data require more accurate source–receiver positioning than low-frequency conventional seismic data. Despite instability of the depths of source and receivers, the method used for measurement of their positions was basically valid, so the instability of their depths could be counteracted during data processing. Furthermore, unstable depth is an important issue (Ross Chapman et al. 2002; Tokarev et al. 2008b; Marsset et al. 2014) because it may distort the hyperbolic curve of the reflection signal in pre-stack data. Distortion of the hyperbolic curve causes defocusing of velocity semblance during the velocity analysis process and producing ambiguous stack section. In seismic data processing, the static correction method is commonly applied to adjust the error from the hyperbolic curve associated with irregular locations of the sources and receivers. Unfortunately, the static correction method is not sufficient to resolve problem of the distorted hyperbolic curve in ACS data because an accurate static correction result needs precise depth information for the source and each receiver. In addition to correction of the geometrical error, the source signature of the ACS system contains

strong sidelobes (ripples) due to autocorrelation noise during the process of source signature extraction. It is crucial to reduce the sidelobes as much as possible because they can decrease the bandwidth frequency of pre-stack data.

In this study, ACS data was processed using a method to adjust unstable receiver depths in pre-stack data. Through this approach, and the hyperbolic curve can be adjusted, and it can minimize the scattering of the obtained velocity semblance. The unstable source effect from post-the stack data was attempted to remove. In addition to the geometrical correction scheme, the sidelobes was suppressed, thus widening the frequency content of the ACS data. This study applied seismic attribute analysis to the final stack section to characterize free gas, gas chimneys, and gas hydrates and identify their locations.

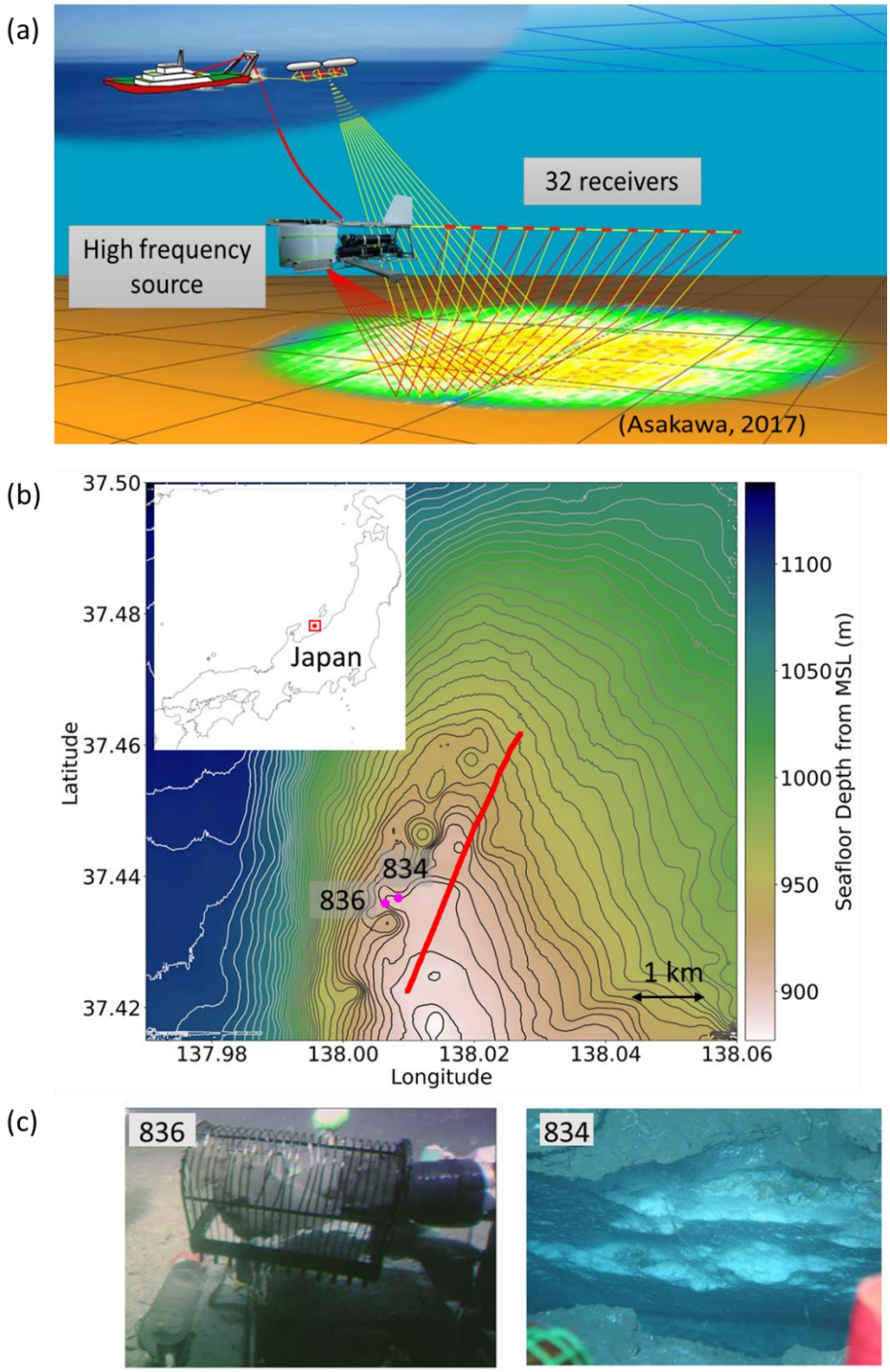

Figure 3.1. Basemap of ACS operation in the Joetsu Basin, Japan. (a) Schematic diagram of an ACS in operation (Asakawa, 2017). (b) Regional location map (inset) and bathymetric map showing the location of the ACS profile of this study (red line). The magenta points are the locations of hydrate outcrops. (c) Photographs of hydrates on the seafloor (JAMSTEC, 2012a, 2012b).

3.2 Geologic Setting and Data

In our study area, the process of basin creation began ~25 m.y. ago during rifting of the Eurasian continent's eastern margin. During the mid-Pliocene, the extensional tectonic regime in this region changed to a compressional regime (Tamaki 1985). The compressional forces not only created geological structures for hydrocarbon traps, but also a hydrocarbon source area (Okui et al. 2008). The Joetsu Basin is southwest of Sado Island at the eastern margin of the Sea of Japan (Figure 3.1b). The basin began to form during the Miocene (Okui et al. 2008; Freire et al. 2011). From the mid-Miocene to the early Pliocene, the sedimentation rate in the Joetsu Basin increased, producing formations later discovered to be reservoirs for hydrocarbons, such as the Teradomari and Shiiya (5.5–3.5 Ma) formations (Okui et al. 2008; Freire et al. 2011). Hydrocarbons in the Joetsu Basin are known to exist as free gas and gas hydrates (Aoyama and Matsumoto 2009; Hiruta et al. 2009; Saeki et al. 2009; Freire et al. 2011; Yoneda et al. 2019; Santos et al. 2020).

J-MARES acquired high-resolution acoustic data using a deep-towed ACS acoustic survey system in the Joetsu Basin, where gas hydrates have been identified (Figure 3.1). This acquisition used a streamer cable that has 32 hydrophones at 5 m intervals. The nearest and farthest offsets were 35 and 190 m, respectively. The sampling rate and record length were 0.1 and 1600 ms, respectively. The source of the ACS data was a iXBlue Echos 1500 sub-bottom profiler with signal frequency ranging from 700 to 2250 Hz (Asakawa et al. 2017). During data acquisition, the positions of the source and receivers varied between 614 and 720 m below sea level. The vessel speed was ~3 knots and the shot intervals were from 2 to 5 m. Details of the acquisition parameters are presented in Table 3.1.

Table 3.1 ACS acquisition parameters for the Joetsu Basin survey

Average Streamer Depth (m)	Average Seawater Velocity (m/s)	Number of Shot Points	Acquisition Direction	Recording Delay (ms)
630	1474.0	1532	N to S	4.10

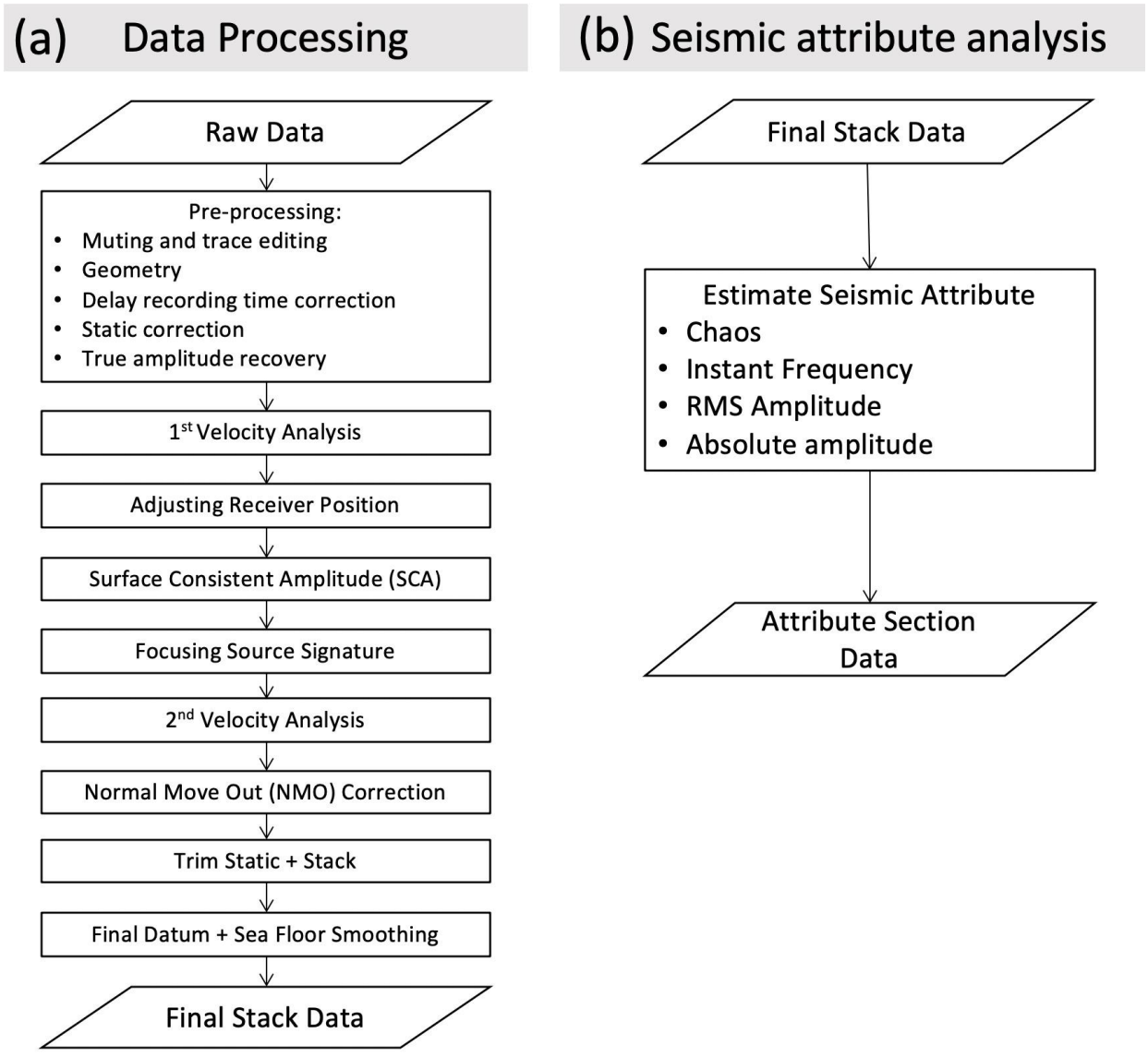

Figure 3.2. Workflow used for processing ACS data in this study. (a) Data processing flow. (b) Seismic attribute processing flow.

3.3 Method

This section describes each step of the data processing (Figure 3.2).

3.3.1 Preprocessing

The two-way travel time (TWT) for the seafloor reflection signals in our ACS pre-stack data were seen after 300 ms. To focus on the signal between 300 and 700 ms (our range of interest signal; red box in Figure 3.3), all signals before 300 ms were muted (green box in Figure 3.3) and after 700 ms (blue box in Figure 3.3). Because the depths of the source and receivers did not remain constant during data acquisition (Figure 3.4), a delay-recording time correction, static correction, and datum shift were needed in the pre-processing step. These steps are known to repair the distorted hyperbolic curve due to the unstable source–receiver geometry (Yilmaz 2001). The source and receiver depth information were used as input for static corrections. Because the shallowest receiver depth was 614 m (Figure 3.4), and to prevent a huge increase in the size of the pre-stack dataset (due to trace shifting during static corrections), a floating datum of 600 m was used. After the static corrections, this preprocessing applied true amplitude recovery (TAR) to compensate for signal amplitude loss due to wave front propagation and inelastic attenuation (Gadallah and Fisher 2005). TAR process used the time raised power correction method with a power correction constant of 2 (Gavotti and Lawton 2013).

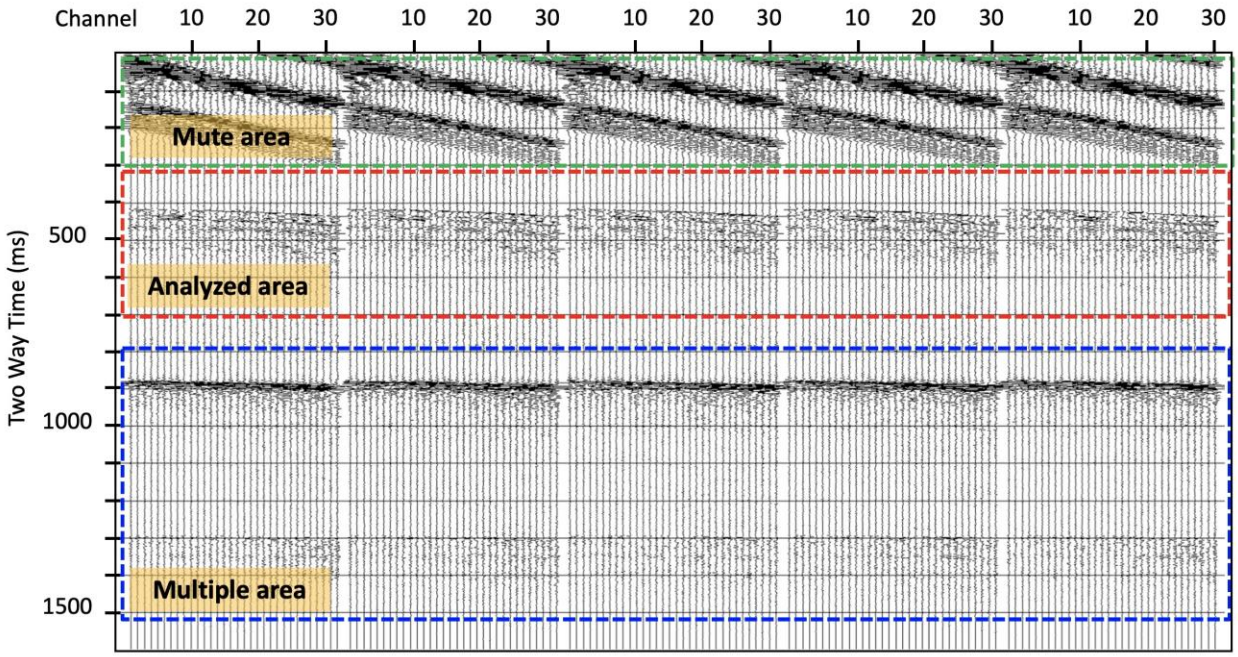

Figure 3.3 Examples of shot gathers from the ACS data. The red box indicates the reflection signals from the target lithology, the green box depicts the noise muted during processing, and the blue box depicts the multiple seismic reflections from the seafloor and sea surface. This study use signal on the red box.

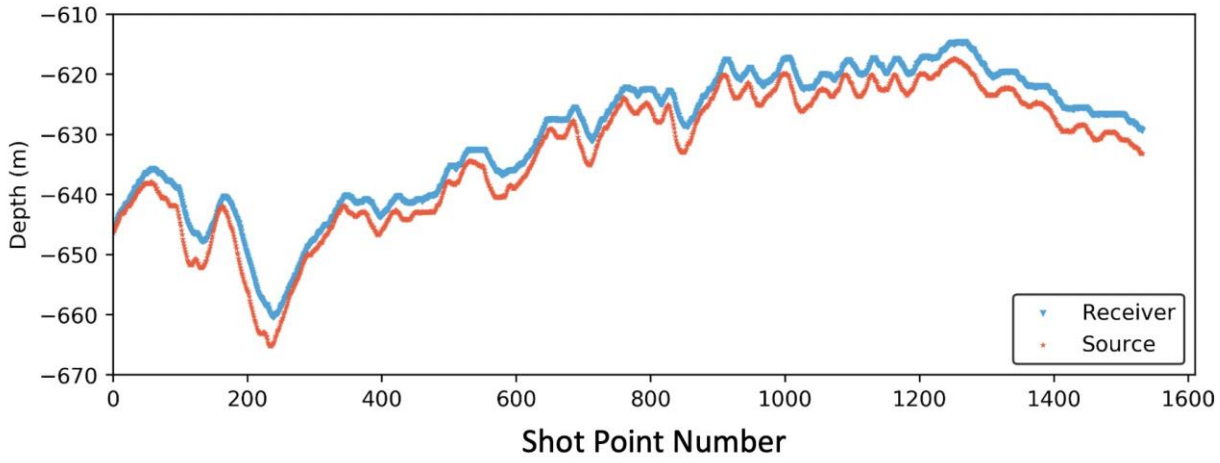

Figure 3.4 Depths of the source and receivers along our ACS profile. The unstable depths distorted the hyperbolic curve of reflection signal.

3.3.2 Adjusting for unstable receiver depths

In high-frequency data processing, inaccurate static corrections may lead to destructive interference during the stacking process (Gutowski et al. 2002; Kluesner et al. 2019) due to distortion of the hyperbolic curve of the reflection signal (Figure 3.5). Inaccurate static corrections caused by inaccurate information about the depths of the source and receivers. During the deep-towed ACS survey, it was difficult to measure the depth accurately for every receiver and source at each shot point because of the effect of strong ocean currents, rough seafloor topography, and the short recording time. Therefore, this study developed a method to adjust the receiver position for a better hyperbolic curve based on the Normal Move Out (NMO) equation (Yilmaz 2001), which this method used to adjust receiver positions in the common depth-point (CDP) gather:

$$T_{(i)}{}^2 = T_0{}^2 + \frac{x_{(i)}^2}{v^2} \tag{3.1}$$

$$dt_{(i)} = T_{(i)} - T_{obs(i)} \tag{3.2}$$

where i is the trace sequence number, x_i is the distance between the source and receiver (offset) for trace i, v is the velocity above the reflector (the speed of sound in seawater; v = 1474 m/s), $T_{(i)}$ is the estimated reflection signal travel time for trace i, T_0 is the zero offset travel time, $T_{obs(i)}$ is the picked seafloor travel time for trace i, and $dt_{(i)}$ is the time correction for trace i. Because this study did not have zero offset data (T_0), so the near offset (first channel) travel time

was used as T_0 data, hence the x_i on near offset was effectively 0 m for this process. The process to repair the hyperbolic curve follows these steps:

- Picking the seafloor reflector for each ACS receiver trace, and identifying its travel time as T_{obs}
- Sorting the ACS traces into the CDP domain
- Calculating the travel time $T_{(i)}$ for every receiver using equation (3.1)
- Calculating the time difference dt_x for every receiver using equation (3.2)
- Shifting the trace data for every receiver by dt_x

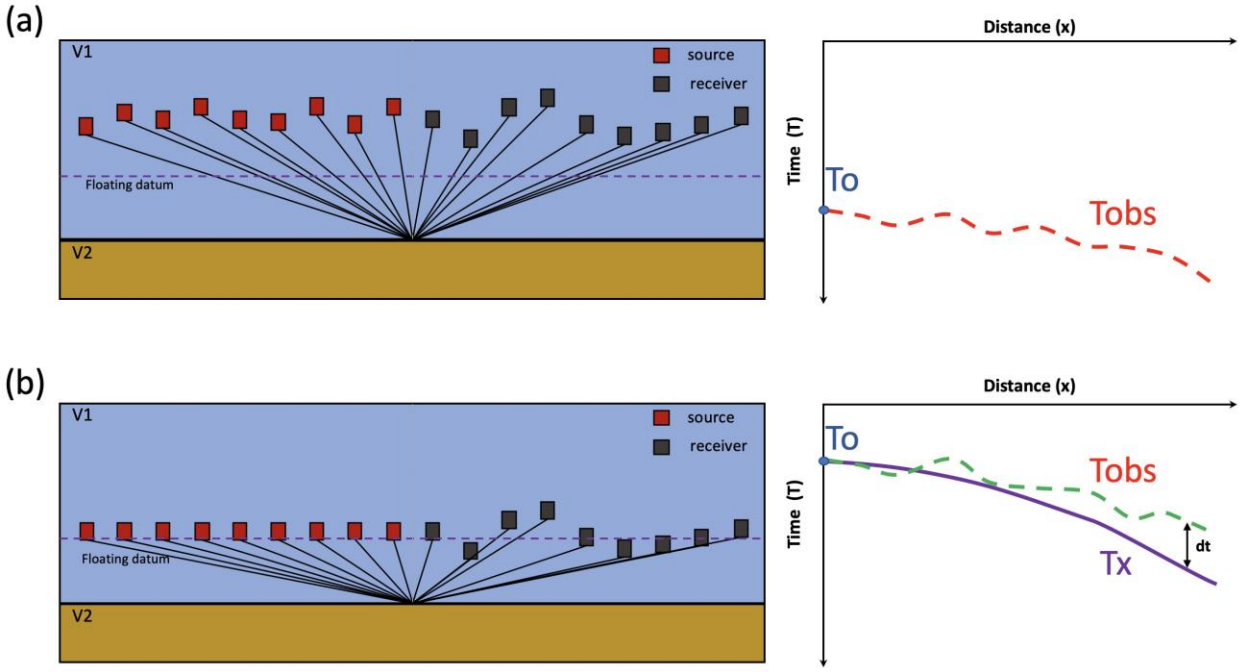

Figure 3.5 Schematic diagram illustrating the adjustment for unstable receiver depths. (a) Source–receiver relationships for a CDP gather with unstable depths of the source and receivers (left panel), distorting the seismic reflection signal (red dashed line in right panel). (b) After applying inaccurate static corrections based on inaccurate receiver depths, the curve of the CDP gather is still distorted (green dashed line). The correct hyperbolic shape of the reflection signal (purple line) was obtained by using equations (1) and (2).

3.3.3 Surface Consistent Amplitude

In seismic data acquisition, spikes or unstable amplitudes in the obtained data can be caused by inconsistency of the energy source, the response and condition of receivers, and the distance between the source and receiver. To reduce the influence of these factors on the acquired data, this study applied the Surface Consistent Amplitude (SCA) method (Levin 1989; Van Vossen et al. 2006), which estimates a scaling factor for every trace by using the gain factor in each domain such as: the common shot, common channel, common offset, and CDP. Then, each sample trace is multiplied by the following scaling factor:

$$S_{(i)} = \frac{1}{(G_{shot(i)} + G_{receiver(i)} + G_{offset(i)} + G_{cdp(i)})} \tag{3.3}$$

$$G_{x(i)} = \frac{RMS_{x(i)}}{\frac{1}{N}\sum_{n=1}^{N} RMS_{x(n)}} \tag{3.4}$$

where $S_{(i)}$ is the scaling factor for SCA at sequence *i*, G_{shot} is the gain factor for the common shot domain, $G_{receiver}$ is the gain factor for the common receiver domain, G_{offset} is the gain factor for the common offset domain, G_{cdp} is the gain factor for the CDP domain, *x* is the type of data domain (i.e., shot, receiver, offset, and CDP), $G_{x(i)}$ is the gain factor for the specific domain *x* (i.e., $G_{shot,}\ G_{receiver},\ G_{offset}, G_{cdp}$) at sequence *i*, $RMS_{x(i)}$ is the root mean square of sequence *i* for domain *x*, and *N* is the total sequence of the specific domain.

3.3.4 Focusing the Source Signature

The source signature of the ACS wavelet was obtained by autocorrelation of the chirp signal. There are some ripples in the signals around the center of the wavelet (Figure 3.6); these are known as sidelobes (Koefoed 1981; Huang et al. 2007; Karsli and Dondurur 2013). Sidelobes are considered to be noise in the correlation process (Nelson 1989). Significant sidelobes in the ACS source wavelet were generated by the narrow bandwidth of the chirp signal (i.e., 700 - 2250 Hz). It is essential to minimize the sidelobes in the source wavelet because they can decrease frequency bandwidth and cause destructive interference in the stacking process (Huang et al. 2007; Karsli and Dondurur 2013). To minimize the effect of sidelobes, this study designed a filter using a Hanning window (Horita and Kanasewich 1983) to make the source wavelet more focused and then applied the filter to the pre-stack data.

3.3.5 Velocity Analysis

Velocity analysis is the process for determining the seismic velocity functions (Yilmaz 2001). The 1st velocity analysis was estimated every 50 CDP intervals to assess whether the pre-processing had created appropriate hyperbolic reflection signals and clear velocity semblance. After making adjustments for unstable receivers, performing SCA, and focusing the source signature, the 2nd velocity analysis were estimated every 25 CDP intervals to obtain a high-resolution velocity model for accurate NMO correction of CDP gathers.

3.3.6 Trim Static and Stacking

Normal moveout correction (NMO) is a stretching of the time axis to make all seismic gather become zero-offset data. Imperfect NMO correction can cause slight tilting of the reflection signal in CDP gathers, especially for the longer offset traces. The difference between the slightly tilted signal and the flat signal after NMO correction is called the Residual Move Out (RMO). The trim static correction method (Gutowski et al. 2002) was used to remove the RMO from the pre-stack data and then stacked the corrected data.

3.3.7 Final Datum Correction and Smoothing

For the final datum correction, mean sea level (MSL) was used as our reference level. MSL can be a global datum for estimating the height of the reflection signal. This process did not directly shift the signal from the floating datum to the MSL datum because the result after trim static and stacking (section 3.6, Figure 3.2) had not considered the unstable source depth. First, because the coordinates of the source and CDP were different, the depth of the source close to each CDP was estimated by linear interpolation. Then, the trace was shifted from the floating datum (600 m in this case) to the estimated source depth. For the final datum correction, the trace was shifted from the estimated source depth to the MSL datum. The equation used to estimate the time correction of final datum process is as follows:

$$t_{final(p)} = t_{float(p)} + \frac{2d_{(p)}}{v} \tag{3.5}$$

where p is the CDP number sequence, d is the interpolated depth of the source near to point p , v is the sea-water velocity (1474 m/s), t_{float} is the time for shifting the floating datum to estimated source depth in point-p, and t_{final} is the correction time to do final datum correction.

3.4 Results

During ACS data acquisition, if the receiver streamer was tilted down (toward the seafloor) (Figure 3.7a), then the recorded travel time of the seafloor reflection was shorter (left panel of Figure 3.7b). Conversely, if the receiver streamer was tilted up (away from the seafloor) (Figure 3.7a), then the recorded travel time of the seafloor reflection was longer (right panel of Figure 3.7b). The greatest effect of tilting was for the receiver with the farthest offset. The maximum absolute receiver correction time of ~10 ms (Figure 3.7c) indicates that the far-offset receiver had been shifted by up to ± 7.5 m or the receiver streamer was tilted by up to 4.7° relative to the near-offset receiver.

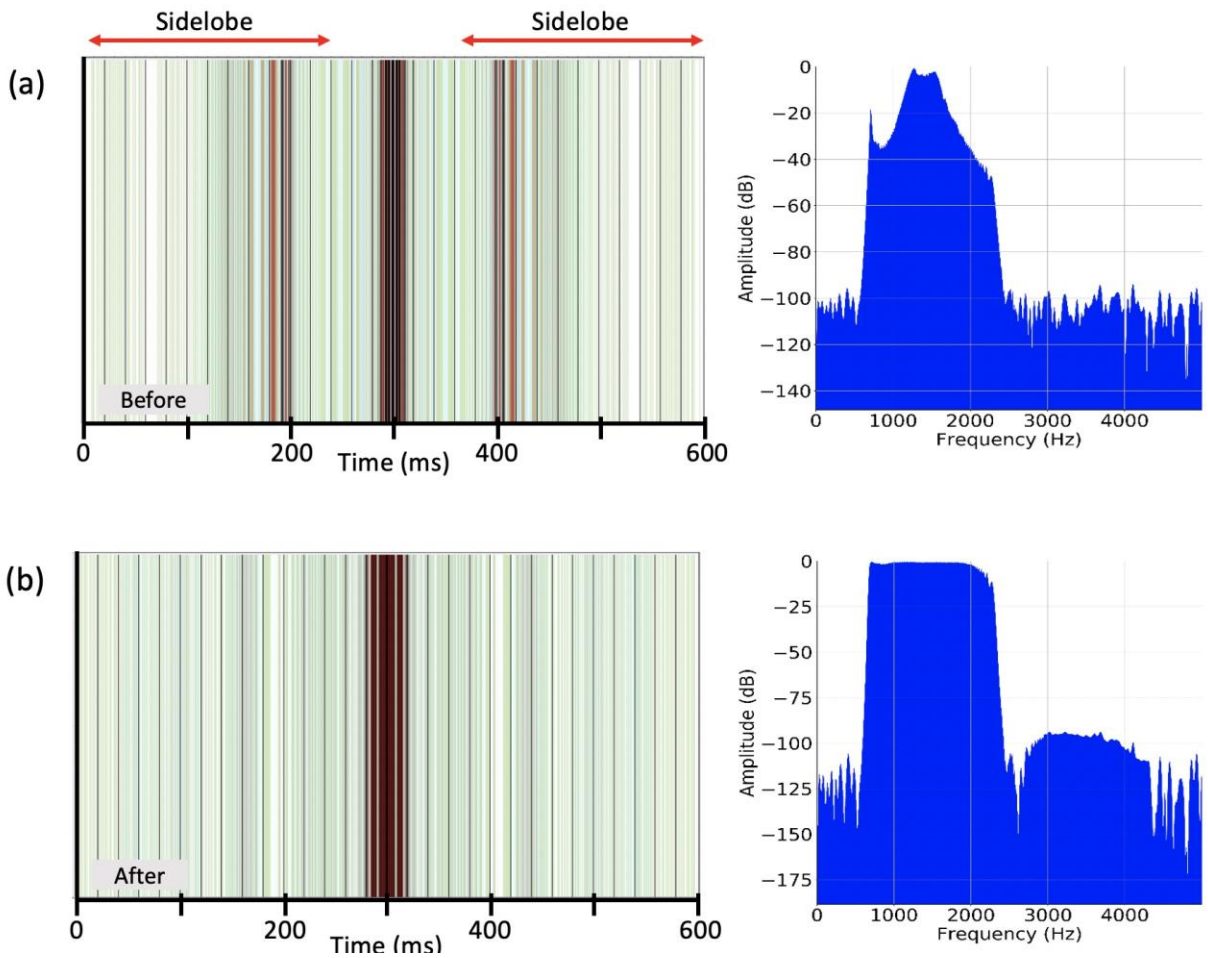

Figure 3.6 Results of focusing the source signature. (a) Original source wavelet (left panel) and frequency spectrum (right panel). (b) Source wavelet (left panel) and frequency spectrum (right panel) after focusing the source signature. The source wavelet in (b) is better focused, the sidelobes have been minimized, and the frequency spectrum is wider.

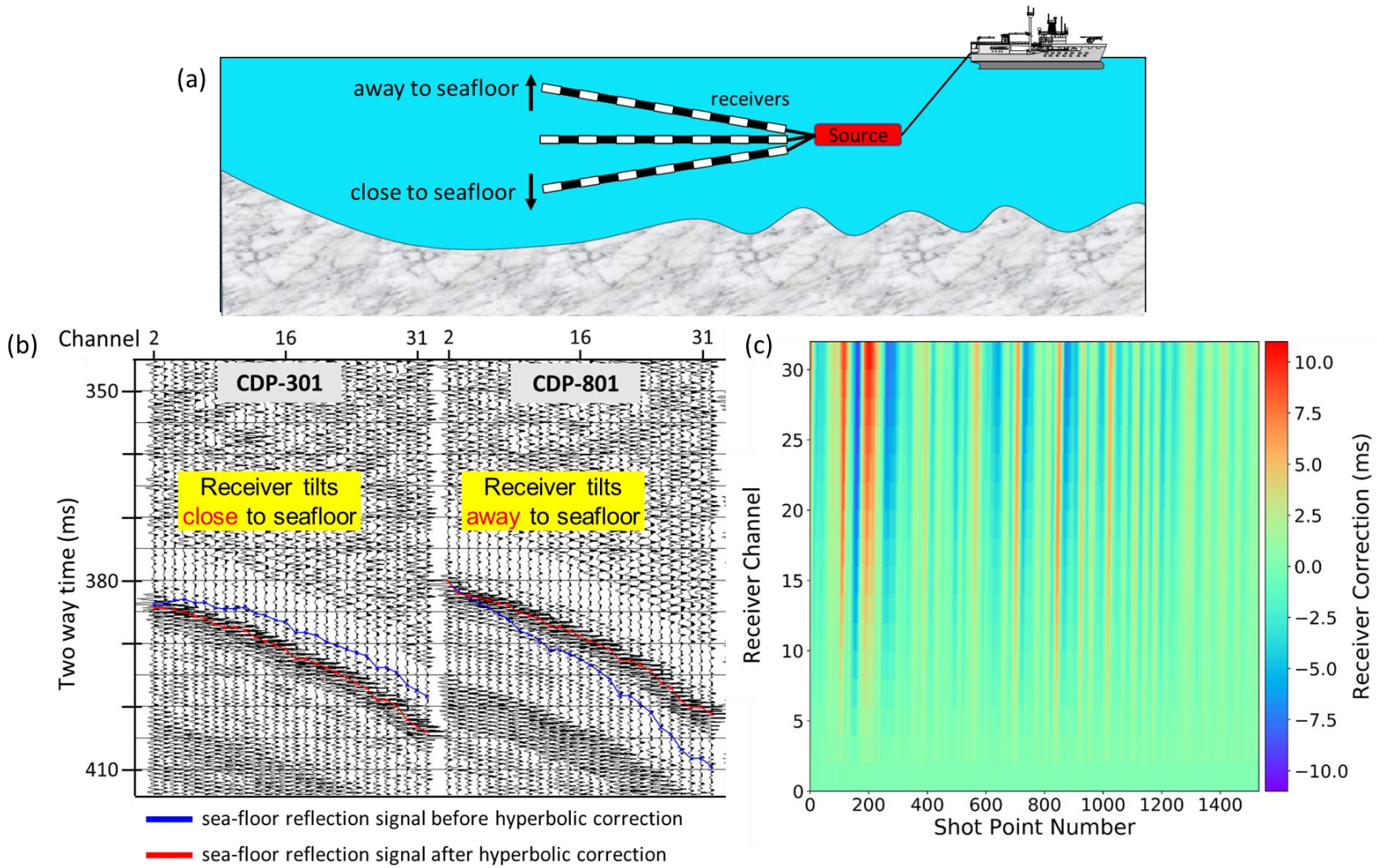

Figure 3.7 Results of adjusting for unstable receiver depths. (a) Schematic illustration of tilted receivers. (b) The blue and red lines are seafloor reflectors in the CDP domain before and after adjustment, respectively. (c) Time corrections required to adjust for unstable depths for all receivers.

The results of focusing the source signature are shown in Figure 3.6b. The sidelobe signals of the source wavelet were notably reduced (left panels in Figure 3.6b). Furthermore, the quality of the source spectrum was improved by as much as 25 dB, especially in the frequency ranges 700–1200 Hz and 1700–2200 Hz (right panel in Figure 3.6). Since the designed filter that used in the focusing step demonstrated good performance, this filter also applied to the pre-stack ACS data.

After pre-processing, the quality of the reflection signals was poor, as shown by the unfocused velocity semblances in the first run of velocity analysis (left panels in Figures 3.8a, 3.8c, 3.8e). These semblances were unfocused because the conventional static correction and delay-recording time correction were not sufficiently effective in correcting the distortion of the recorded reflection signal because of unstable streamer position. However, after application of the adjustments for unstable receivers, SCA, and focusing the source signature, the quality of the reflection signal markedly improved. The improvements are demonstrated by the better focused velocity semblances produced by the second run of velocity analysis (left panels in Figures 3.8b, 3.8d, 3.8f). Moreover, by applying the filter that designed for focusing the source signature to the pre-stack ACS data, the average frequency spectrum in the CDP domain became wider. In particular, the average spectrum of ACS data in the frequency ranges of 700–1000 Hz and 1700–2200 Hz increased by as much as 20 dB (right panels in Figures 3.8b, 3.8d, 3.8f) in comparison to the frequencies after pre-processing (right panels in Figures 3.8a, 3.8c, 3.8e). These improvements at low and high frequencies in the pre-stack data are essential for achieving high resolution in the stack section (Yilmaz 2001; Karsli and Dondurur 2013).

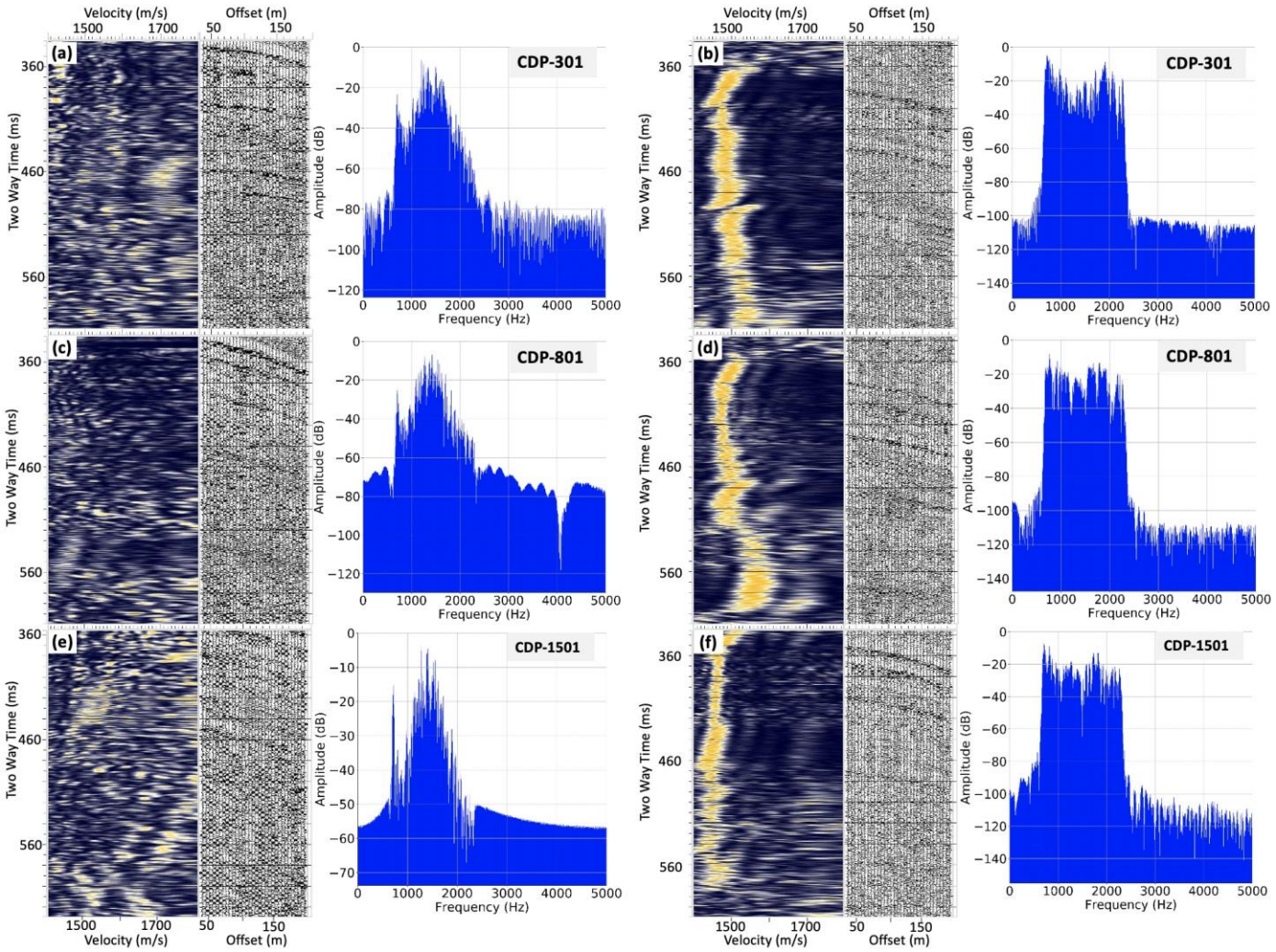

Figure 3.8 Examples of velocity analyses (left panels) and averaged frequency spectra (right panels) for CDP gathers (center panels) from three positions along the profile. (a), (c), and (e) re-stack data after pre-processing steps. (b), (e), and (f) Pre-stack data after adjustment for unstable receiver depths, SCA, and focusing of source signature.

Source depth (Figure 3.9a) has a strong influence on both the near-offset gather (Figure 3.9b) and the stack section (Figure 3.9c). The stack section (Figure 3.9c) shows significant improvements of signal-to-noise ratio at almost all CDP locations in comparison to the near channel gather (Figure 3.9b). After final datum correction and smoothing, the effect of source undulation was removed, and the final stacked profile was obtained (Figure 3.9d). The final stack section shows several shallow faults in Northern and Southern side with high spatial resolution (thickness less than 100 m) (Figure 10). It is hard for conventional seismic reflection to obtain this high resolution geological structure because subsurface with 100 m thickness is shown by single reflector signal (low spatial resolution). In order to identify hydrocarbon features (i.e., gas hydrates, free gas, and gas chimneys), this study calculated several seismic attributes from the final stacked data (Figures 11 and 12). Chimneys have vertical geometries that disrupt the seismic response of the sedimentary sequence and are often associated with low-amplitude blanking (Nourollah et al. 2010; Singh et al. 2016; Nourollah and Urosevic 2019). These features can be emphasized by post-stack seismic attributes (Tsuji et al. 2012).

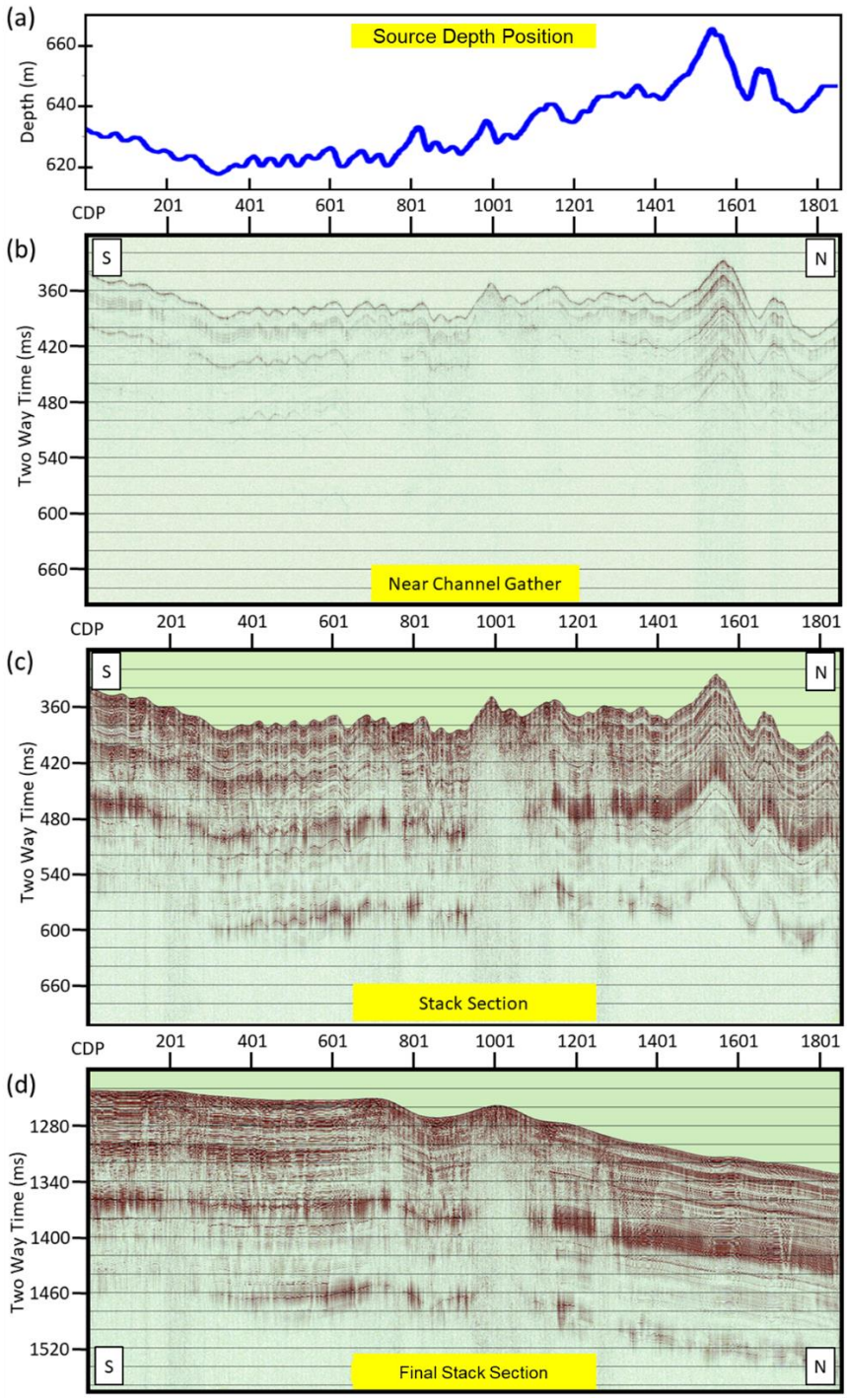

Figure 3.9 (a) Depths of the source near each CDP along the profile. (b) Near-offset gather (seismic trace from 1st channel only) after pre-processing. (c) Stack section after adjustment for unstable receiver depths, SCA, and focusing of source signature) without final datum correction. (d) Final stack section after final datum correction. The seafloor reflections in (b) and (c) show correlations with the depth profile in (a).

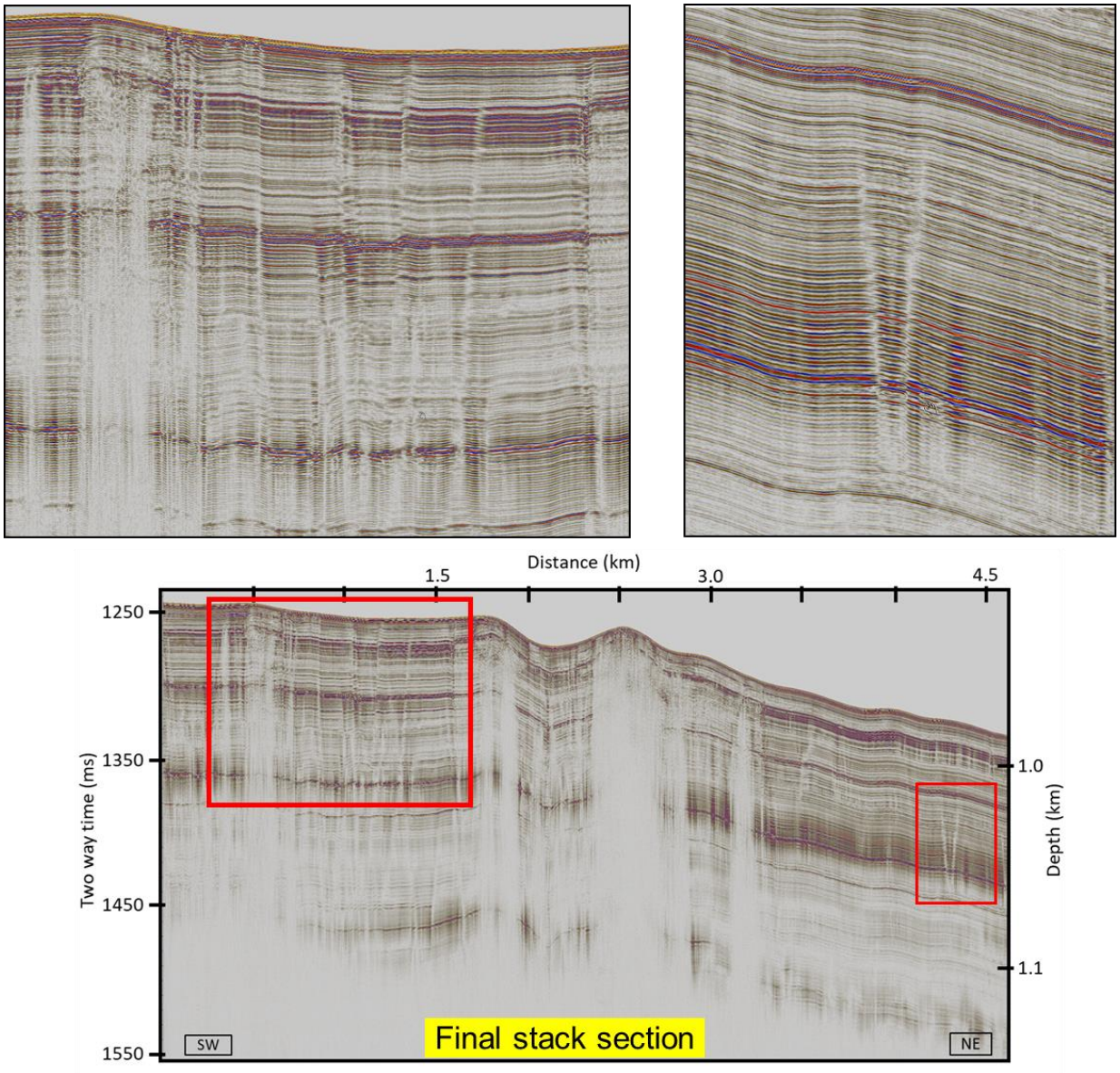

Figure 3.10 Final stack section shows several shallow faults with high spatial resolution. Thickness of geological structures are less than 100 m. Top panels are the examples of faults in shallow depth, and bottom panel is final stack section.

3.5 Interpretation

Gas chimneys have features that are related to fluids or gases such as pockmarks, mud volcanoes, and carbonate formations (Heggland 1998; Nourollah et al. 2010; Kioka et al. 2019). Methane hydrate is commonly formed at the low-temperature and high-pressure conditions of deep-sea environments (Kvenvolden and McMenamin, 1980; Pandey et al., 2019; Sain and Gupta, 2012; Sloan, 2003), and its presence can be identified by Bottom Simulating Reflectors (BSR) on reflection seismic profiles (Ross Chapman et al. 2002; Ojha and Sain 2009; Sain and Gupta 2012; Eng and Tsuji 2019; Ye et al. 2019). To identify hydrocarbon gas and hydrates from the ACS derived profile, seismic attributes were used, including absolute amplitude, Root Mean Square (RMS) amplitude, chaos attribute, and instantaneous frequency (Chopra and Marfurt, 2007; Ismail et al., 2019; Kim et al., 2015; Kumar et al., 2019; Nourollah et al., 2010; Singh et al., 2016; Thakur and Rajput, 2011; Tsuji et al., 2005).

Tracking gas chimneys is useful for understanding the migration pathways of hydrocarbons from source rock to reservoir (Saeki et al. 2009; Freire et al. 2011; Eng and Tsuji 2019; Kret et al. 2020). The presence of gas chimneys causes amplitude blanking in stack section. The chaos attribute and reflection amplitude attributes (e.g., RMS amplitude) can be used to identify amplitude blanking (Figure 3.11; Nourollah et al., 2010; Nourollah and Urosevic, 2019; Singh et al., 2016). These attributes identified four dominant gas chimneys in our stack profile and showed them to be under localized anticline structures. Indeed, the hydrate deposits near the seafloor observed by JAMSTEC (Figures 1c, 1d) are within a similar ridge structure. The upward migration of methane could be important to make high concentration of hydrate deposits in the shallow subsurface, including the seafloor (Kvenvolden and McMenamin 1980; Judd and Hovland 2007;

Kret et al. 2020). In addition, many minor discontinuities were observed, such as small chimneys (area that has seismic blanking) and faults (discontinuity layer).

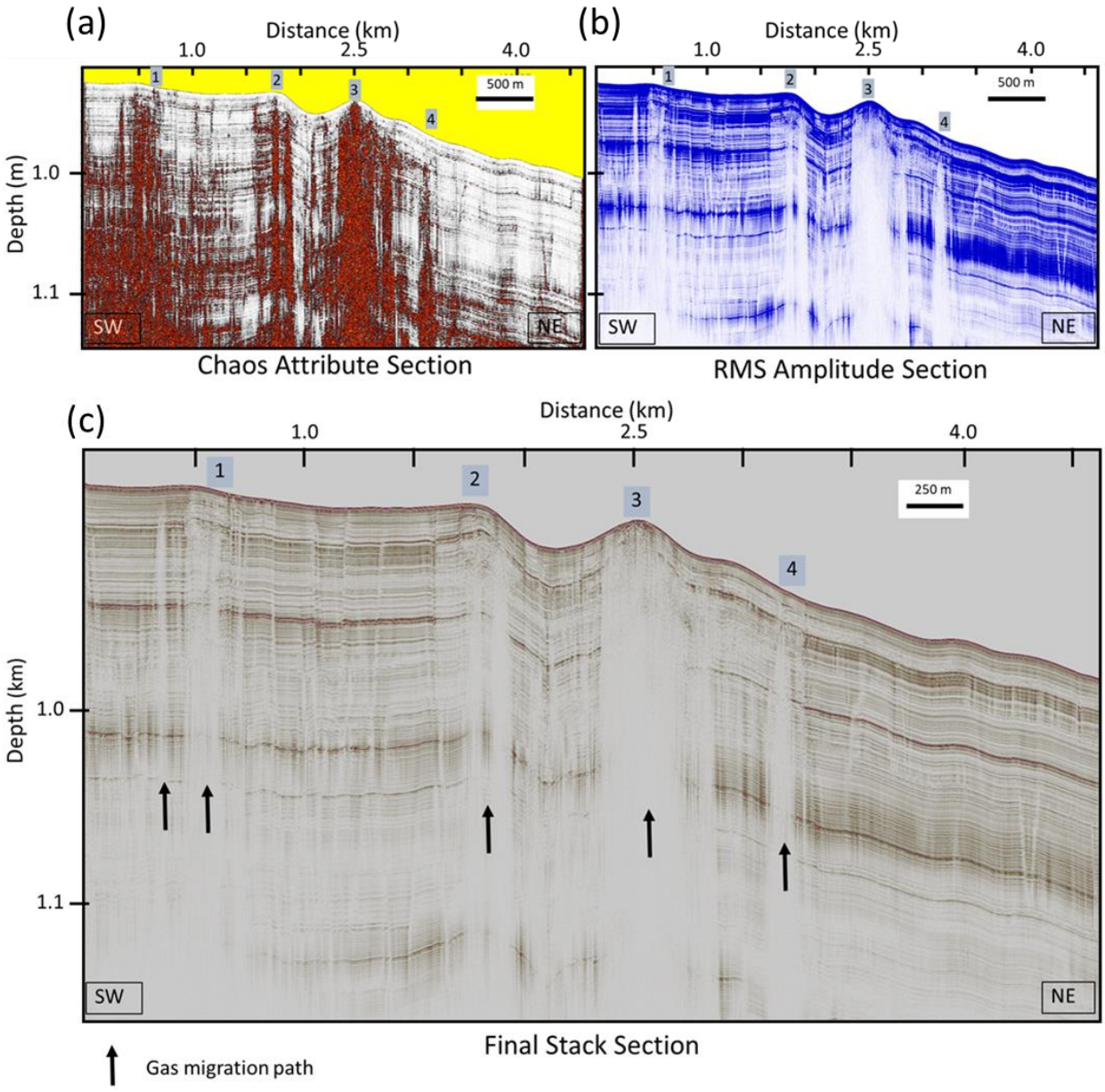

Figure 3.11 Identification of gas chimneys from seismic attribute analysis of the ACS final stack section. (a) Chaos attribute section. (b) RMS amplitude section. (c) Final stack section depicts gas migration paths.

Gas hydrate close to the seafloor generates amplitude variations in the seafloor reflector because the acoustic impedance of gas hydrate is different from that of water-saturated marine sediment (right panel in Figures 3.12 and 3.13). A localized area of strong amplitude was observed on the seafloor, suggesting the presence of concentrated hydrate. Because the high-amplitude area of the seafloor reflector lies above minor faults (e.g., right panel in Figure 3.12a), upward migration of fluid through the minor faults might be important for deposition of hydrate close to the seafloor. This attribute could be useful to identify the localized hydrate close to the seafloor.

Mixing of gas hydrate with free gas causes frequency fluctuations and amplitude blanking below the seafloor (left panels in Figure 3.12; Figure 3.13; Kim et al., 2015; Lee et al., 2017; Ojha and Sain, 2009; Sain and Gupta, 2012). Furthermore, the presence of gas hydrate and free gas together appears to have caused frequency fluctuation at the same location where amplitude variations are visible(middle panels in Figure 3.12; Ojha and Sain, 2009). From these observations, the accumulation of gas hydrate and free gas is seen to be between CDPs 221 and 331 (Figure 3.12a). Interestingly, a localized hydrate accumulation appears to be not only at the top of the anticline structure (Figures 3.12b, 3.13a), but also on its flank (Figures 3.12a, 3.13b).

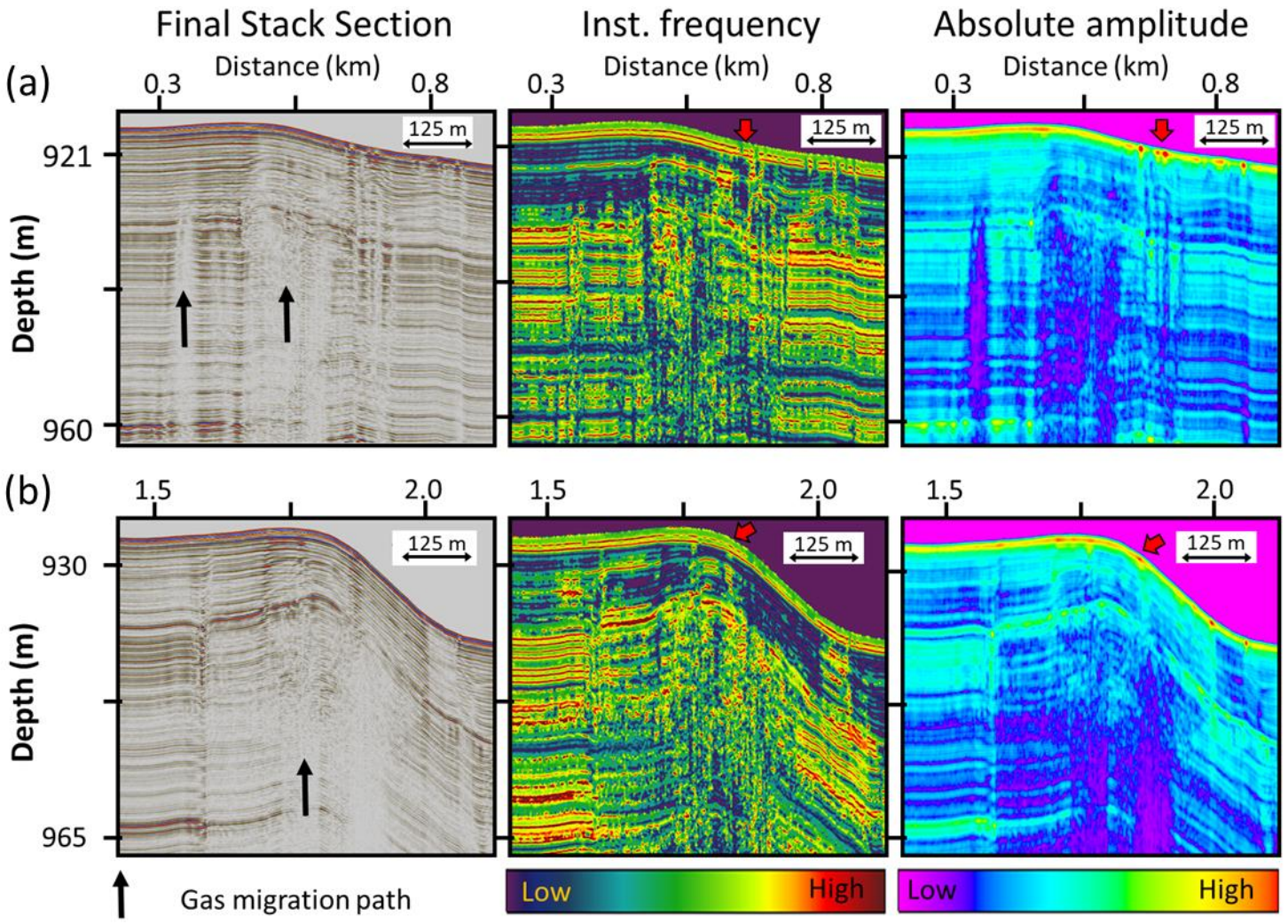

Figure 3.12 Potential areas for gas and hydrate accumulations in our ACS final stack section. (a) and (b) Gas potentials areas at locations 1 and 2 in Figure 3.11c, respectively. The left panels are the stack section, the center panels are instantaneous frequency, and the right panels are absolute amplitude. Red arrows indicate the localized hydrate close to the seafloor.

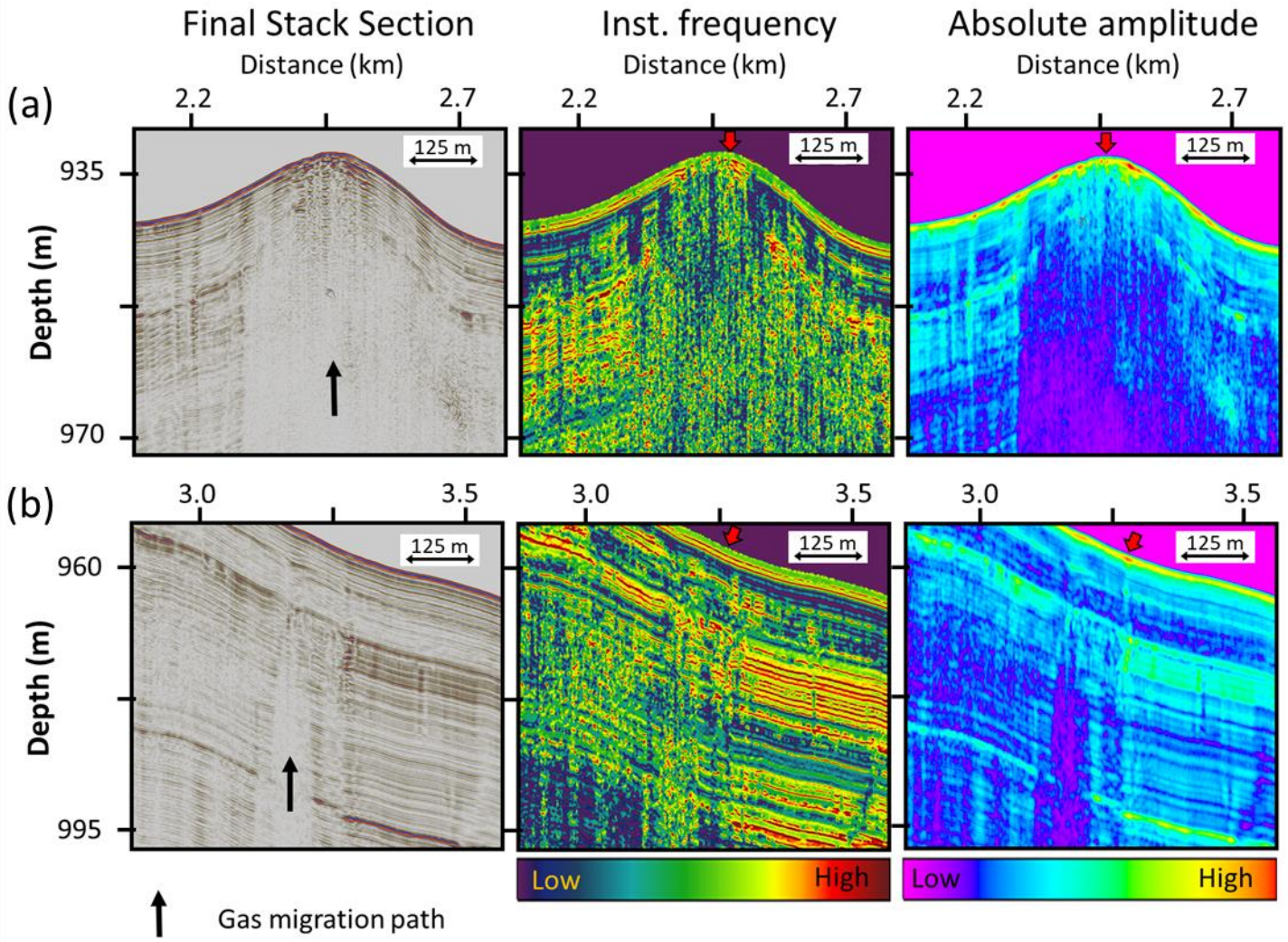

Figure 3.13 *Potential areas for gas and hydrate accumulations in our ACS final stack section. (a) and (b) Gas potentials areas at locations 3 and 4 in Figure 3.10c, respectively. The left panels are the stack section, the center panels are instantaneous frequency, and the right panels are absolute amplitude. Red arrows indicate the localized hydrate close to the seafloor.*

3.6 Conclusions

The unstable depths of sources and receivers is a common problem that occurs in deep-towed acoustic surveys. In terms of data processing, the conventional static correction method is insufficient to correct the timing error of the reflection signal due to the unstable depths of source and receivers. Our pre-stack data processing scheme provides a solution for this problem for deep-towed acoustic survey data. Even if complete receiver depth information is not available for all receivers, our processing scheme can recover the reflection signal if offset information is available, and it notably improves the focusing of velocity semblances. In addition, our data processing scheme increases effective frequency bandwidth as a result of improvement in the focusing of the source signature.

In our seismic attribute analysis, chaos and RMS amplitude attributes were used to successfully identify four gas chimneys as pathways for upward migration of gas. Furthermore, this study used absolute amplitude and instantaneous frequency attributes to identify localized hydrate accumulations close to the seafloor.

3.7 References

Angkasa, S.S., Jerram, D.A., Millett, J.M., Svensen, H.H., Planke, S., Taylor, R.A., Schofield, N., Howell, J., 2017. Mafic intrusions, hydrothermal venting, and the basalt-sediment transition: Linking onshore and offshore examples from the North Atlantic igneous province. Interpretation 5, SK83–SK101. https://doi.org/10.1190/INT-2016-0162.1

Aoyama, C., Matsumoto, R., 2009. Acoustic Surveys of Methane Plumes by Quantitative Echo Sounder in Japan Sea and the Estimate of the Seeping Amount of the Methane Hydrate Bubbles. Chigaku Zasshi (Jounal Geogr. 118, 156–174. https://doi.org/10.5026/jgeography.118.156

Asakawa, E., Murakami, F., Saito, S., Tara, K., Kato, M., Tsukahara, H., Lee, S., Sumi, T., Yamakawa, T., 2017. New seismic survey techniques for seafloor massive sulphide (SMS) Exploration. 23rd Eur. Meet. Environ. Eng. Geophys. https://doi.org/10.3997/2214-4609.201702091

Bischoff, A.P., Nicol, A., Beggs, M., 2017. Stratigraphy of architectural elements in a buried volcanic system and implications for hydrocarbon exploration. Interpretation. https://doi.org/10.1190/int-2016-0201.1

Chopra, S., Marfurt, K.J., 2007. Seismic Attributes for Prospect Identification and Reservoir Characterization. Society of Exploration Geophysicists and European Association of Geoscientists and Engineers. https://doi.org/10.1190/1.9781560801900

Eng, C., Tsuji, T., 2019. Influence of faults and slumping on hydrocarbon migration inferred from

3D seismic attributes: Sanriku-Oki forearc basin, northeast Japan. Mar. Pet. Geol. https://doi.org/10.1016/j.marpetgeo.2018.10.013

Freire, A.F.M., Matsumoto, R., Santos, L.A., 2011. Structural-stratigraphic control on the Umitaka Spur gas hydrates of Joetsu Basin in the eastern margin of Japan Sea. Mar. Pet. Geol. 28, 1967–1978. https://doi.org/10.1016/j.marpetgeo.2010.10.004

Gadallah, M.R., Fisher, R.L., 2005. Applied seismology : a comprehensive guide to seismic theory and application. PennWell, Tulsa, Okla.

Gavotti, P.E., Lawton, D.C., 2013. Seismic processing workflow for supressing coherent noise while retaining low-frequency signal, CREWES Research Report.

Gutowski, M., Breitzke, M., Spei, V., 2002. Fast static correction methods for high-frequency multichannel marine seismic reflection data: A high-resolution seismic study of channel-levee systems on the Bengal Fan. Mar. Geophys. Res. https://doi.org/10.1023/A:1021240415963

Hachikubo, A., Yanagawa, K., Tomaru, H., Lu, H., Matsumoto, R., 2015. Molecular and Isotopic Composition of Volatiles in Gas Hydrates and in Sediment from the Joetsu Basin, Eastern Margin of the Japan Sea. Energies 8, 4647–4666. https://doi.org/10.3390/en8064647

Heggland, R., 1998. Gas seepage as an indicator of deeper prospective reservoirs. A study based on exploration 3D seismic data. Mar. Pet. Geol. https://doi.org/10.1016/S0264-8172(97)00060-3

Hiruta, A., Snyder, G.T., Tomaru, H., Matsumoto, R., 2009. Geochemical constraints for the formation and dissociation of gas hydrate in an area of high methane flux, eastern margin

of the Japan Sea. Earth Planet. Sci. Lett. 279, 326–339. https://doi.org/10.1016/j.epsl.2009.01.015

Horita, R.E., Kanasewich, E.R., 1983. Time Sequence Analysis in Geophysics. Can. J. Stat. / La Rev. Can. Stat. https://doi.org/10.2307/3314985

Huang, J., Gao, L., Gao, Y., 2007. Side lobes of wavelets impact identification of thin sand bodies. Appl. Geophys. https://doi.org/10.1007/s11770-007-0016-9

Ismail, A., Ewida, H.F., Al-Ibiary, M.G., Gammaldi, S., Zollo, A., 2019. Identification of gas zones and chimneys using seismic attributes analysis at the Scarab field, offshore, Nile Delta, Egypt. Pet. Res. 5, 59–69. https://doi.org/10.1016/j.ptlrs.2019.09.002

JAMSTEC, 2012a. NATSUSHIMA NT08-09 Leg2 Cruise Data. Japan Agency for Marine-Earth Science and Technology (JAMSTEC). https://doi.org/10.17596/0000488

JAMSTEC, 2012b. NATSUSHIMA NT08-09 Leg1 Cruise Data. Japan Agency for Marine-Earth Science and Technology (JAMSTEC). https://doi.org/10.17596/0000487

Joshima, M., Nishimura, K., Kisimoto, K., Hiromatsu, M., Sato, M., Matsumoto, R., 2009. Micro-topography and Shallow Sub-bottom Structure on the Gas Hydrate Area off Joetsu City, Niigata, Japan: DAI-PACK Mapping Study. Chigaku Zasshi (Jounal Geogr. 118, 72–92. https://doi.org/10.5026/jgeography.118.72

Judd, A., Hovland, M., 2007. Seabed Fluid Flow, Seabed Fluid Flow: The Impact on Geology, Biology, and the Marine Environment. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511535918

Karsli, H., Dondurur, D., 2013. A procedure to reduce side lobes of reflection wavelets: A

contribution to low frequency information. J. Appl. Geophys. 96, 107–118. https://doi.org/10.1016/j.jappgeo.2013.07.002

Kim, K.J., Yi, B.Y., Kang, N.K., Yoo, D.G., 2015. Seismic Attribute Analysis of the Indicator for Gas Hydrate Occurrence in the Northwest Ulleung Basin, East Sea. Energy Procedia 76, 463–469. https://doi.org/10.1016/j.egypro.2015.07.882

Kioka, A., Tsuji, T., Otsuka, H., Ashi, J., 2019. Methane Concentration in Mud Conduits of Submarine Mud Volcanoes: A Coupled Geochemical and Geophysical Approach. Geochemistry, Geophys. Geosystems. https://doi.org/10.1029/2018GC007890

Kluesner, J., Brothers, D., Hart, P., Miller, N., Hatcher, G., 2019. Practical approaches to maximizing the resolution of sparker seismic reflection data. Mar. Geophys. Res. 40, 279–301. https://doi.org/10.1007/s11001-018-9367-2

Koefoed, O., 1981. ASPECTS OF VERTICAL SEISMIC RESOLUTION. Geophys. Prospect. https://doi.org/10.1111/j.1365-2478.1981.tb01008.x

Kret, K., Tsuji, T., Chhun, C., Takano, O., 2020. Distributions of gas hydrate and free gas accumulations associated with upward fluid flow in the Sanriku-Oki forearc basin, northeast Japan. Mar. Pet. Geol. 116, 104305. https://doi.org/10.1016/j.marpetgeo.2020.104305

Kumar, J., Sain, K., Arun, K.P., 2019. Seismic attributes for characterizing gas hydrates: a study from the Mahanadi offshore, India. Mar. Geophys. Res. 40, 73–86. https://doi.org/10.1007/s11001-018-9357-4

Kvenvolden, K.A., McMenamin, M.A., 1980. Hydrates Of Natural Gas: A Review of Their

Geologic Occurence. Geol. Surv. Circ. (United States). https://doi.org/10.3133/cir825

Lee, J., Byun, J., Kim, B., Yoo, D.G., 2017. Delineation of gas hydrate reservoirs in the Ulleung Basin using unsupervised multi-attribute clustering without well log data. J. Nat. Gas Sci. Eng. https://doi.org/10.1016/j.jngse.2017.08.007

Levin, S.A., 1989. Surface-consistent deconvolution. Geophysics. https://doi.org/10.1190/1.1442747

Marsset, B., Ker, S., Thomas, Y., Colin, F., 2018. Deep-towed high resolution seismic imaging II: Determination of P-wave velocity distribution. Deep Sea Res. Part I Oceanogr. Res. Pap. 132, 29–36. https://doi.org/10.1016/j.dsr.2017.12.005

Marsset, B., Menut, E., Ker, S., Thomas, Y., Regnault, J.P., Leon, P., Martinossi, H., Artzner, L., Chenot, D., Dentrecolas, S., Spychalski, B., Mellier, G., Sultan, N., 2014. Deep-towed high resolution multichannel seismic imaging. Deep. Res. Part I Oceanogr. Res. Pap. 93, 83–90. https://doi.org/10.1016/j.dsr.2014.07.013

Marsset, T., Marsset, B., Ker, S., Thomas, Y., Le Gall, Y., 2010. High and very high resolution deep-towed seismic system: Performance and examples from deep water Geohazard studies. Deep. Res. Part I Oceanogr. Res. Pap. https://doi.org/10.1016/j.dsr.2010.01.001

Ministriy Economic Trade and Industry (METI), 2016. ANRE Estimated the Amount of Shallow Methane Hydrate Resources and Verified the Estimation Results [WWW Document]. URL https://www.meti.go.jp/english/press/2016/0916_03.html

Nelson, D.E., 1989. Methods and apparatus for reducing correlation sidelobe interference in seismic profiling systems. J. Acoust. Soc. Am. https://doi.org/10.1121/1.398650

Nourollah, H., Keetley, J., O'Brien, G., 2010. Gas chimney identification through seismic attribute analysis in the Gippsland Basin, Australia. Lead. Edge (Tulsa, OK). https://doi.org/10.1190/1.3479999

Nourollah, H., Urosevic, M., 2019. Evaluation of sealing potential near gas chimneys–the Gippsland Basin, Australia. J. Appl. Geophys. 160, 254–263. https://doi.org/10.1016/j.jappgeo.2018.11.019

Ojha, M., Sain, K., 2009. Seismic attributes for identifying gas-hydrates and free-gas zones: application to the Makran accretionary prism. Episodes 32, 264–270. https://doi.org/10.18814/epiiugs/2009/v32i4/003

Okui, A., Kaneko, M., Nakanishi, S., Monzawa, N., Yamamoto, H., 2008. An integrated approach to understanding the petroleum system of a frontier deep-water area, offshore Japan. Pet. Geosci. https://doi.org/10.1144/1354-079308-765

Pandey, L., Sain, K., Joshi, A.K., 2019. Estimate of gas hydrate saturations in the Krishna-Godavari basin, eastern continental margin of India, results of expedition NGHP-02. Mar. Pet. Geol. 108, 581–594. https://doi.org/10.1016/j.marpetgeo.2018.12.009

Riboulot, V., Cattaneo, A., Sultan, N., Garziglia, S., Ker, S., Imbert, P., Voisset, M., 2013. Sea-level change and free gas occurrence influencing a submarine landslide and pockmark formation and distribution in deepwater Nigeria. Earth Planet. Sci. Lett. 375, 78–91. https://doi.org/10.1016/j.epsl.2013.05.013

Romagnoli, C., Casalbore, D., Ricchi, A., Lucchi, F., Quartau, R., Bosman, A., Tranne, C.A., Chiocci, F.L., 2018. Morpho-bathymetric and seismo-stratigraphic analysis of the insular

shelf of Salina (Aeolian archipelago) to unveil its Late-Quaternary geological evolution. Mar. Geol. 395, 133–151. https://doi.org/10.1016/j.margeo.2017.10.003

Ross Chapman, N., Gettrust, J.F., Walia, R., Hannay, D., Spence, G.D., Wood, W.T., Hyndman, R.D., 2002. High-resolution, deep-towed, multichannel seismic survey of deep-sea gas hydrates off western Canada. Geophysics 67, 1038–1047. https://doi.org/10.1190/1.1500364

Sacchi, M., Caccavale, M., Corradino, M., Esposito, G., Ferranti, L., Hámori, Z., Horváth †, F., Insinga, D., Marino, C., Matano, F., Molisso, F., Natale, J., Passaro, S., Pepe, F., Tóth, T., 2019. The use and beauty of ultra-high-resolution seismic reflection imaging in Late Quaternary marine volcaniclastic settings, Napoli Bay, Italy. Földtani Közlöny 149, 371. https://doi.org/10.23928/foldt.kozl.2019.149.4.371

Saeki, T., Inamori, T., Nagakubo, S., Ward, P., Asakawa, E., 2009. 3D Seismic Velocity Structure below Mounds and Pockmarks in the Deep Water Southwest of the Sado Island. Chigaku Zasshi (Jounal Geogr. 118, 93–110. https://doi.org/10.5026/jgeography.118.93

Sain, K., Gupta, H., 2012. Gas hydrates in India: Potential and development. Gondwana Res. https://doi.org/10.1016/j.gr.2012.01.007

Santos, L.A., Neves, E.H.P., Freire, A.F.M., Santos, M.A.C., Matsumoto, R., Ajus, C.M.I., 2020. Diffraction velocity analysis in a single-channel seismic survey in the Joetsu Basin. GEOPHYSICS 85, U47–U53. https://doi.org/10.1190/geo2019-0011.1

Savoye, B., Leon, P., De Roeck, Y.H., Marsset, B., Lopes, L., Herveou, J., 1995. PASISAR: a new tool for near-bottom very high-resolution profiling in deep water. First Break.

https://doi.org/10.3997/1365-2397.1995013

Singh, D., Kumar, P.C., Sain, K., 2016. Interpretation of gas chimney from seismic data using artificial neural network: A study from Maari 3D prospect in the Taranaki basin, New Zealand. J. Nat. Gas Sci. Eng. 36, 339–357. https://doi.org/10.1016/j.jngse.2016.10.039

Sloan, E.D., 2003. Fundamental principles and applications of natural gas hydrates. Nature. https://doi.org/10.1038/nature02135

Tamaki, K., 1985. Two modes of back-arc spreading. Geology 13, 475. https://doi.org/10.1130/0091-7613(1985)13<475:TMOBS>2.0.CO;2

Thakur, N.K., Rajput, S., 2011. Exploration of gas hydrates: Geophysical techniques, Exploration of Gas Hydrates: Geophysical Techniques. https://doi.org/10.1007/978-3-642-14234-5

Tokarev, M., Kuzub, N., Pevzner, R., Kalmykov, D., Bouriak, S., 2008. System for Shallow Water Investigation 26, 77–85.

Tsuji, T., Takai, K., Oiwane, H., Nakamura, Y., Masaki, Y., Kumagai, H., Kinoshita, M., Yamamoto, F., Okano, T., Kuramoto, S., 2012. Hydrothermal fluid flow system around the Iheya North Knoll in the mid-Okinawa trough based on seismic reflection data. J. Volcanol. Geotherm. Res. 213–214, 41–50. https://doi.org/10.1016/j.jvolgeores.2011.11.007

Van Vossen, R., Curtis, A., Trampert, J., 2006. Surface-consistent amplitude corrections for single or multicomponent sources and receivers using reciprocity and waveform inversion. Geophys. J. Int. 165, 311–322. https://doi.org/10.1111/j.1365-246X.2006.02908.x

Wood, W.T., Hart, P.E., Hutchinson, D.R., Dutta, N., Snyder, F., Coffin, R.B., Gettrust, J.F., 2008.

Gas and gas hydrate distribution around seafloor seeps in Mississippi Canyon, Northern Gulf of Mexico, using multi-resolution seismic imagery. Mar. Pet. Geol. https://doi.org/10.1016/j.marpetgeo.2008.01.015

Ye, J., Wei, J., Liang, J., Lu, J., Lu, H., Zhang, W., 2019. Complex gas hydrate system in a gas chimney, South China Sea. Mar. Pet. Geol. 104, 29–39. https://doi.org/10.1016/j.marpetgeo.2019.03.023

Yilmaz, Ö., 2001. Seismic Data Analysis: Processing, Inversion, and Interpretation of Seismic Data, Investigations in geophysics. https://doi.org/10.1190/1.9781560801580

Yoneda, J., Kida, M., Konno, Y., Jin, Y., Morita, S., Tenma, N., 2019. In Situ Mechanical Properties of Shallow Gas Hydrate Deposits in the Deep Seabed. Geophys. Res. Lett. 46, 14459–14468. https://doi.org/10.1029/2019GL084668

Chapter 4

Conclusions

4.1 Conclusions

The combination of several advanced data processing methods and computation technology have given significant results on the utilization of passive and active source in geophysical research. The ambient noise is best compatible for monitoring velocity variation for long term duration with high temporal resolution. The proposed data processing schemes on deep-towed data are suitable for obtaining high spatial resolution shallow subsurface structure. The key findings on dissertation below are summarized below:

Chapter 2 provides a comprehensive monitoring system that can monitor the spatio-temporal velocity changes in the Japanese Islands. This monitoring system is like geodetic monitoring which can be beneficial for monitoring crustal behavior. The proposed method, named

as “Sliding Reference Method”, is suitable for monitoring the temporal velocity variation with high stability, high temporal resolution, and close to the newest geological condition in comparison with the “Absolute Reference Method” (conventional). Moreover, the proposed method was given higher *C(E)* compare to conventional method. The high performance computation implemented in this monitoring system, has given breakthrough for making fast computation results, so this monitoring system can continuously monitor velocity variation for long periods. There is no hardware limitation in this system, hence it can be implemented for monitoring velocity variations in other areas. Moreover, this system has been designed so the computation speed can be easily to improve by adding the computation resource. In the future, the area that can record the ambient noise data, through the seismometers or the fiber optic sensors, may be broaden with denser sampling points. Therefore, a global scale monitoring system (worldwide scale) can be developed by using the monitoring system that was described previously. As the result, the worldwide geological crustal phenomena can be further observed with detailed resolution in future.

Chapter 3 shows that the integration of the proposed data processing scheme and seismic attribute on deep-towed acquisition data has given a significant result for imaging shallow subsurface in high spatial resolution. The common problem related with the unstable source-receiver depth of deep-towed acquisition has been counteracted during data processing. Minimizing the sidelobes on source signature also become an important factor for increasing the frequency content of pre-stack data. For gas exploration, the seismic attribute succeeded in identifying the presence of gas chimney, gas hydrate, and free-gas near the seafloor. Many locations have enormous potential for containing large amount of gas hydrate. Moreover, the energy demand will increase in the future, hence gas hydrate can be an important asset to fulfill this high energy demand. The deep-towed acquisition can be used for exploring the gas hydrate on

other location areas. These processing schemes on deep-towed acquisition data are not only useful for hydrocarbon exploration, but are also useful for another cases, such as: shallow submarine geo-hazard potential, exploring the volcano system, finding seafloor massive sulfides deposit, etc. The high spatial resolution subsurface image is important for understanding the geological subsurface in more detail, hence the deep-towed acquisition can be another geophysical solution for achieving that purpose.

Acknowledgments

The journey as PhD student in Kyushu University has given me so much wonderful experience as a human being. During this PhD period, some many supports and kindness that peoples had given to me. I cannot describe all those peoples one by one, but I will like to express my gratitude to the following people.

Firstly, I would like to express my sincere gratitude to my very supportive supervisor Prof. Takeshi Tsuji for the continuous support of my PhD study and related research. In my opinion, not so many geophysicists as smart as him who can understand the earth phenomena from nanoscale to geologic scale. His knowledge about research is so wide, sometimes I wonder “can I borrow his brain for one month?”. He also very humble, kind, full energy, cheerful, and polite person. His guidance helped me in all the time of research and writing of this dissertation. I always surprise that he always has a time for student who want to discuss about anything. I am fell so grateful and lucky to know him during my lifetime journey.

I also want to say big thank you for Assistant Prof. Dr. Tatsunori Ikeda. Because of him, I know the “beauty of the surface wave”. In my previous study, the surface wave always become my big enemies. But, after so many positive discussions with him, he has changed my paradigm

about the surface wave so much. He also very patient, detail, and logic person. I always enjoy discussing about research and drinking coffee with him. Many of his technical guidance has solved my research problem and made me obtain a good research result.

I also want to say thank you for Kyushu University officer staff, Tomomi Yamaguchi and Yasura Oiwa. They have given many supports related to administrative problem during my life in Japan. They made my daily Japan life become easier.

I also want to say thank you very much also Japan Ministry of Education, Culture, Sports, Science and Technology (MEXT) for providing me the scholarship during this 3 year period. I want to say thank you also for Dr. rer. nat Rachmat Sule, Prof. Toshifumi Matsuoka, and Takao Aizawa PhD. At beginning, they had introduced and convinced me to continue my study under Prof. Takeshi Tsuji.

For Chapter 2 and 3, I thank you Prof. Martha Savage (Victoria University of Wellington), Dr. Masafumi Katou (JOGMEC) and Dr. Eichii Asakawa (JGI, Inc /J-MARES) for constructive comments and discussions for this study. In Chapter 2 research, I thank the National Research Institute for Earth Science and Disaster Resilience (NIED) for providing waveform of Hi-net data. The Chapter 2 research was supported by the Japan Society for the Promotion of Science (JSPS) through Grants-in-Aid for Scientific Research in Innovative Areas (JP17H05318), and by Education and Research Center for Mathematical and Data Science, Kyushu University. Also, I gratefully acknowledge support of I2CNER sponsored by the MEXT through the World Premier International Research Center Initiative. In the Chapter 3 research, I also thank J-GI, Inc /J-MARES and Japan Agency for Marine-Earth Science and Technology (JAMSTEC) for providing all the data. The Chapter 3 was supported by Council for Science, Technology and Innovation

(CSTI), Cross-ministerial Strategic Innovation Promotion Program (SIP), “Next generation technology for ocean resources exploration” (Lead agency: JAMSTEC), and JSPS KAKENHI grants 20H01997.

I want to say big special thanks for Felicia Agatha Satriya and Delfina Hanna Chrisyandra, because they were become my personal proofreader. They help me a lot when I was writing manuscript. Moreover, they always respond to my inquires very quickly and without any payment. I owe both of them so much.

These acknowledgements would not complete without mentioning all my Exploration Geophysics Lab colleagues (Japanese and non-Japanese) and all Indonesia community (Susenji Crews, Takata Cius Gengs, and West-4 Brotherhood). We had so much fun together in Japan!

In the end, I am grateful to my parents, siblings, and girlfriend (Ezra Lenny) who remembered me in their prayers for the ultimate success. I consider myself nothing without them. They gave me enough moral support, encouragement and motivation to accomplish the PhD journey goals.

Fukuoka, Summer Season 2020

Fernando Lawrens Hutapea

Appendix

This appendix contents the python script for building the ambient noise monitoring system (my research focus and originality). In this script several processing steps have been developed, such as: stretching interpolation, ternary search, creating reference and current trace by using ARM or SRM, and removing outlier velocity changes. Several parameters should be defined such as: “root_out_folder; root_data_folder, cross_corr_folder_name; output_starttime; output_endtime; ref_stack_starttime, ref_window_size,cur_window_size, stack_method, stack_folder_name, full_stack_window_output, plot, e, e_delta, parallel, wave_duration, stretching_folder_name, data_cc_merge, log_folder, mad_tolerance, mad_thresh_cc,mad_filename, median_filter_enable, E0_correction, output_folder.

```
####################################
#                                  #
# 04_estimating_temporal_velocity.py#
#                                  #
####################################
import sys, os, shutil, glob, time, multiprocessing
from scipy.signal import argrelextrema, medfilt
import numpy as np

def calc_correction_E0(data_in, num_day=30):
    ''' E_corr = E - E0'''
    names = []
    for descr in data_in.dtype.descr:
        if len(descr) == 2:
            names.append(descr[0])
    dates = np.unique(data_in["starttime"])
    min_date = np.min(dates)
    max_date = min_date + np.timedelta64(num_day, 'D')
    idx = (data_in["starttime"] >= min_date) & (data_in["starttime"] <= max_date)
    ### Make Data E0 correction (mean of num_day) ###
    data_E0 = data_in[idx]
    df_E0 = pd.DataFrame({name : data_E0[name] for name in names})
    df_E0_mean = df_E0.groupby(["station", "pair_name"]).mean(). reset_index()
```

```
    data_E0_mean = df_E0_mean.to_records(index=False)
    data_E_corr = []
    print("#### Process: optimize_velocity_v1 (correction_with_e0) \n#### Status: Start\n\n")
    for data in data_in:
        pair_name = data["pair_name"]
        station = data["station"]
        idx = (data_E0_mean["station"] == station) & (data_E0_mean["pair_name"] == pair_name)
        buff = data["e_max"]
        if np.sum(idx) > 0:
            try:
                data["e_max"] = data["e_max"] - data_E0_mean["e_max"][idx]
            except:
                print("ERROR")
            data_E_corr.append(data)
#         print(buff-data["e_max"])
    print("#### Process: optimize_velocity_v1 (correction_with_e0) \n#### Status: Finish \n\n")
    result = np.concatenate([data_E_corr])
    return(result)

def calc_median_filter(data_in):
    data = np.copy(data_in)
    kernel_size = 3
    half_kernel_size = int(kernel_size / 2)
    e_max_pad = np.pad(data["e_max"], (half_kernel_size, half_kernel_size), 'edge')
    cc_max_pad = np.pad(data["cc_max"], (half_kernel_size, half_kernel_size), 'edge')
    for start, emax in enumerate(data["e_max"]):
        end = start + kernel_size
        buff_e_max = e_max_pad[start:end]
        buff_cc_max = cc_max_pad[start:end]
        if np.isnan(emax) == False:
            e_max_med = np.nanmedian(buff_e_max)
            data["e_max"][start] = e_max_med
            try:
                data["cc_max"][start] = np.max(buff_cc_max[buff_e_max == e_max_med])
            except:
                pass
        e_max_pad = np.pad(data["e_max"], (half_kernel_size, half_kernel_size), 'edge')
        cc_max_pad = np.pad(data["cc_max"], (half_kernel_size, half_kernel_size), 'edge')
    return(data)


def calc_mad(data_in, thresh_cc = "None", mad_tolerance = "None", nan_outlier=1, concatenate_output = 1, l
og_folder=None, median_filter=1): #"Make remove outlier, if not pass MAD then 1"
    '''
    Input : merge of all data channel (np.ndarray)
    '''
    if type(data_in) == list:
        buff = []
        for bf in data_in:
            if type(bf) == np.ndarray:
                buff.append(bf)
        if len(buff) != 0:
            data = np.concatenate(buff)
    elif type(data_in) == np.ndarray:
        data = data_in
    data = stretching_interp_fluctuation_rate(data)
    print("#### Process: optimize_velocity_v1(calc_mad) \nStatus: Start!")
    if (thresh_cc == "None"): thresh_cc = 0.5
    if (mad_tolerance == "None"): mad_tolerance = 3
    stations = np.unique(data["station"])
    new_dtype = data.dtype.descr + [("fr_mad", "f")]
    names = list(data.dtype.names)
    result = []
    for stn in stations:
        slice_stn = data[data["station"]==stn]
        pairs = np.unique(slice_stn["pair_name"])
        for pair in pairs:
            count = 0
            slice_pair = slice_stn[slice_stn["pair_name"]==pair]
```

```
            buff = np.ones_like(slice_pair, dtype=new_dtype)
            buff[names] = slice_pair[names]
            if "cc_max" not in buff.dtype.names:
                new_dtype = buff.dtype.descr + [("cc_max", "f")]
                buff = np.ones_like(buff, dtype=new_dtype)
                buff[names] = slice_pair[names]
                buff["cc_max"] = np.max(buff["CC"], axis=1)
            #### Find low CC ####
            cc_low = np.where(buff["cc_max"] < thresh_cc)[0]
            #### Find CC has min max local ###
            for i, CC in enumerate(buff["CC"]):
                maxInd = argrelextrema(CC, np.greater)[0]
                if(len(maxInd) > 1):
#                 if(len(maxInd) > 1) and np.max(CC) < 0.7:
                    cc_low = np.append(cc_low, i)
            ### remove duplicated
            cc_low = np.unique(cc_low)
            #### CALC MAD FOR E_MAX #####
            print(stn, pair)
            arr = slice_pair["e_max"]
            med = np.nanmedian(slice_pair["e_max"])
            mad = np.nanmedian(np.abs(arr - med))
            thresh_min = med - mad_tolerance * mad
            thresh_max = med + mad_tolerance * mad
            idx = (buff["e_max"] > thresh_min) & (buff["e_max"] < thresh_max)
            for id in cc_low:
                if ((buff["e_max"][id] > thresh_min) & (buff["e_max"][id] < thresh_max)) == False:
                    print("#### Process: optimize_velocity_v1(calc_mad) \nStatus: MAD Applied! \n{} {} {}\
n".format(buff["e_max"][id], thresh_min, thresh_max))
                    if (nan_outlier == 1):
                        buff["e_max"][id] = np.nan
                    else:
                        buff["e_max"][id] = 0
                    count +=1
            ### Apply Median Filter ###
            if median_filter != 0:
                print("#### Process: optimize_velocity_v1(calc_median_filter) \n#### Status: Median Filter
 START!! \n{}\n".format([stn, pair[:6]]))
                buff = calc_median_filter(buff)
                print("#### Process: optimize_velocity_v1(calc_median_filter) \n#### Status: Median Filter
 FINISH!!\n{}\n".format([stn, pair[:6]]))
            print(stn, pair, np.sum(idx), len(cc_low))
            ### remove if %count
            if (count / len(buff["e_max"])) < 0.15:
                result.append(buff)
    if(concatenate_output == 1): result = np.concatenate(result)
    print("#### Process: optimize_velocity_v1(calc_mad) \n#### Status: FINISH!\n\n")
    if (type(log_folder) != type(None)):
        if not os.path.isdir(log_folder): os.makedirs(log_folder, exist_ok=True)
        log_file = os.path.join(log_folder, "mad.log")
        f = open(log_file, "w")
        f.write("Removed = {}".format(count))
        f.close()
    return(result)

def optimize_velocity(data_in, mad_tolerance=3, thresh_cc= 0.5, median_filter=1, E0_correction=1,
                      output_folder=None, filename="all_mad.npy"):
    script_name = os.path.basename(__file__)
    mad_folder_path = output_folder
    ###calculate mad and median filter
    data = calc_mad(data_in=data_in, mad_tolerance=mad_tolerance, thresh_cc= thresh_cc, median_filter=medi
an_filter)
    ###correction with E0
    if E0_correction != 0:
        data = calc_correction_E0(data_in=data)
    if mad_folder_path != None:
        mad_file_path = os.path.join(mad_folder_path, filename)
        if os.path.isdir(mad_folder_path):
            shutil.rmtree(mad_folder_path)
```

```
        os.makedirs(mad_folder_path, exist_ok=True)
        np.save(mad_file_path, data)
        print("#### Process: {} (optimize_velocity)\n#### Status: FINISH! Optimize Velocity Process!!\n {}
\n".format(script_name, mad_file_path))
    else:
        return(data)


def strecthing_interpolation_function(cur_files, ref_files, e, e_delta, plot, wave_duration,  tt = 0, coda
_side = "positive"):
    if wave_duration == None:
    checkpar_name = list(('sampling_rate', 'npts', 'station', 'longitude', 'latitude', 'fmin', 'fmax', 'pa
ir_name', 'starttime'))
    vel  = 1
    e_start = -1 *e
    e_end = e
    stn = np.unique(cur_files['station'])
    date = np.unique(cur_files["starttime"])
    result_files = []
    print("#### Process: stretching_interpolation_v9 \nStatus: Start Stretching Interpolation! \n{} {}\n".
format(stn, date))
    for cur, ref in zip(cur_files, ref_files):
        if (cur[checkpar_name] == ref[checkpar_name]) & (type(cur) == type(ref)):
            M = int(cur['npts']/2)
            sampling_rate = cur['sampling_rate']
            f_cur_ = np.fft.irfft(cur['freq'], cur['npts']) ## Return to time domain
            buff1 = np.copy(f_cur_) ###Flip The Array IFFT
            f_cur_[:M] = buff1[M:]
            f_cur_[M:] = buff1[:M]
            f_ref_= np.fft.irfft(ref['freq'], ref['npts'])
            buff1 = np.copy(f_ref_) ###Flip The Array IFFT
            f_ref_[:M] = buff1[M:]
            f_ref_[M:] = buff1[:M]
            t_all = np.linspace(-M/sampling_rate, M/sampling_rate, num=cur['npts']) ### create  time array
            t1 = cur["pair_distance"] / vel
            window = wave_duration
            t_start = t1 + tt
            t_end = t_start + window
            if(coda_side == "negative"):
                t_start = -1 * t_start
                t_end = -1 * t_end
            E = np.arange(e_start, e_end, e_delta)
            CC = np.zeros_like(E)
            idx = (t_all >= t_start) & (t_all <= t_end)
            t = t_all[(t_all >= t_start) & (t_all <= t_end)]
            f_cur = f_cur_[(t_all >= t_start) & (t_all <= t_end)]
            f_ref = f_ref_[(t_all >= t_start) & (t_all <= t_end)]
            ## Apply Stretching Interpolation ##
            for i, e in enumerate(E):
                te_cur = np.multiply(1+e, t)
                fe_cur = np.interp(t, te_cur, f_cur)
                num = np.sum(np.multiply(fe_cur, f_ref))
                pow1 = np.power(fe_cur,2)
                pow2 = np.power(f_ref,2)
                denom = np.power(np.multiply(np.sum(pow1), np.sum(pow2) ), 0.5)
                CC[i] = np.divide(num, denom, out=np.zeros_like(num), where=denom!=0)
            cc_max = np.nanmax(CC)
            e_max = np.nanmax(E[CC == cc_max])
            ## Apply Tenery Search ##
            N_tenery = 10
            e_max_tenery = e_max
            cc_max_tenery = cc_max
            e_delta_tenery = e_delta/2
            for i in range (N_tenery):
                E_tenery = np.array([e_max_tenery, e_max_tenery - e_delta_tenery, e_max_tenery + e_delta_t
enery])
                CC_tenery = np.zeros_like(E_tenery)
                CC_tenery = cc_max_tenery
                for e in (E_tenery[1:]):
```

```
                    te_cur = np.multiply(1+e, t)
                    fe_cur = np.interp(t, te_cur, f_cur)
                    num = np.sum(np.multiply(fe_cur, f_ref))
                    pow1 = np.power(fe_cur, 2)
                    pow2 = np.power(f_ref, 2)
                    denom = np.power(np.multiply(np.sum(pow1), np.sum(pow2) ), 0.5)
                    cc = np.divide(num, denom, out=np.zeros_like(num), where=denom!=0)
                    CC_tenery = np.append(CC_tenery, cc)
                e_delta_tenery = e_delta_tenery / 2
                cc_max_tenery = np.nanmax(CC_tenery)
                e_max_tenery = np.nanmax(E_tenery[CC_tenery == cc_max_tenery])
            if (cc_max_tenery > cc_max) :
                print("#### Process: stretching_interpolation_v9 \nStatus: CC-
Coef Increase !!! After tenery search \n cc_max {}\n e_max {}\n\n".format([cc_max, cc_max_tenery], [e_max,
 e_max_tenery]))
            elif (cc_max_tenery < cc_max) :
                print("#### Process: stretching_interpolation_v9 \nStatus: CC-
Coef Decrease !!! After tenery search \n cc_max {}\n e_max {}\n\n".format([cc_max, cc_max_tenery], [e_max,
 e_max_tenery]))
            id_emax=np.argwhere(E < e_max_tenery)
            if len(id_emax) > 0:
                id_emax = id_emax[-1].item() + 1
            elif len(id_emax) == 0:
                id_emax = 0
            CC = np.insert(CC, id_emax , cc_max_tenery)
            E = np.insert(E, id_emax, e_max_tenery)
            ## Make Result ##
            cc_max = np.nanmax(CC)
            e_max = np.nanmax(E[CC == cc_max])
            if cc_max == 0: e_max = 0
        else:
            print("#### Process: stretching_interpolation_v9 \nStatus: Header not same! {} {} {}\n".format
(cur['station'], cur['pair_name'], cur['starttime']))
        result_dtype = [('sampling_rate', '<i4'), ('npts', '<i4'), ('starttime', '<M8[ms]'), ('station', '
<U10'), ('motion', '<U10'),
                        ('longitude', '<f4'), ('latitude', '<f4'), ('fmin', '<f8'), ('fmax', '<f8'), ('pai
r_name', '<U20'), ('pair_distance', '<f4'),
                        ('pair_lon', '<f4'), ('pair_lat', '<f4'), ('stack_count', '<i4'), ("coda_side", "U
20"), ('wave_duration','<f4'), ('t_start','<f4'),
                        ('t_end','<f4'), ("E", 'f', len(CC)), ("CC", 'f', len(CC)), ('e_delta', "<f4"), ('
cc_max','<f4'), ('e_max','<f4'), ('e_rms','<f4'),
                        ("cur_window_size", 'i'), ("ref_window_size", 'i'), ("cur_window_type", 'U20'), ("
ref_window_type", 'U20'), ('pair_code', '<U20'),
                        ('stack_method', '<U20'), ("cur_starttime", '<M8[D]'), ("ref_starttime", '<M8[D]')
, ("cur_endtime", '<M8[D]'), ("ref_endtime", '<M8[D]')]
        copycur_name = ['sampling_rate', 'npts', 'station', 'starttime', 'motion', 'longitude', 'latitude'
, 'fmin', 'fmax', 'pair_name',
                            'pair_distance', 'pair_lon', 'pair_lat']
        names = ["coda_side", 'wave_duration', 't_start', 't_end', "E", "CC", 'e_delta', 'cc_max', 'e_max'
,  "cur_window_size", "ref_window_size",
                 "cur_window_type", "ref_window_type", "e_rms", "pair_code", "cur_starttime", "ref_starttim
e", "cur_endtime", "ref_endtime"]
        cur_window_size = cur["window_size"]
        ref_window_size = ref["window_size"]
        cur_window_type = cur["window_type"]
        ref_window_type = ref["window_type"]
        cur_starttime = cur["stack_starttime"]
        ref_starttime = ref["stack_starttime"]
        cur_endtime = cur["stack_endtime"]
        ref_endtime = ref["stack_endtime"]
        pair_code = "{}--{}".format(cur['station'], cur['pair_name'][:6])
        e_rms = calc_stretching_rms_error(data=f_ref, cc_max=cc_max, t_start=t_start, t_end=t_end, fmin=re
f["fmin"], fmax=ref["fmax"], sps = ref["sampling_rate"])
        result = np.zeros(1, dtype=result_dtype)
        result[copycur_name] = cur[copycur_name]
        try:
            for name in names : result[name] = eval(name)
        except:
```

```
            errorfile = os.path.join(os.getcwd() + "stretching.{}.error".format(MPI.COMM_WORLD.Get_size())
)
            f = open(errorfile, "w")
            f.write("{} {} {}".format(stn, date, pair_name))
        result_files.append(result)
#     output = np.concatenate(result_files)
    print("#### Process: stretching_interpolation_v9 \nStatus: *** Finish Stretching Interpolation! *** \n
{} {}\n\n".format(cur['station'], cur['starttime']))
    if(len(result_files) != 0):
        return(np.concatenate(result_files))

def strecthing_interpolation(data_in, plot=0, e=0.025, e_delta = 0.0005, parallel=0, wave_duration=None, t
t=0 , output_starttime = None, output_endtime=None):
    '''
    Input  =  array of full path of filename (str) each channel
    Output = array of numpy struct array
    '''
    print("##### Process: stretching_interpolation_v9 (stretching_interpolation) \nStatus: Starting \nTota
l input: {}".format(len(data_in)))
    ### Stretching Interpolation ###
    result = []
    if (len(data_in) != 0):
        for file in data_in:
            if (len(file) > 0 and file != [] and type(file) != type(None)):
                #### check data input
                if (type(file) == str) or (type(file) == np.str_): stacks_file = np.load(file)
                elif(type(file) == np.ndarray): stacks_file = file
                else: sys.exit("#### Process: stretching_interpolation_v9 (stretching_interpolation)\nStat
us: Not support Data Input Type {}\n".format(type(file)))
                cur_files = stacks_file[stacks_file['trace_type'] == 'current'] ### select all current fil
e
                ref_files = stacks_file[stacks_file['trace_type'] == 'reference'] ### select all reference
 file
                if type(output_starttime) != type(None) and type(output_endtime) != type(None):
                    dates = np.arange(np.datetime64(output_starttime, 'D'), np.datetime64(output_endtime,
'D') + np.timedelta64(1,"D"), dtype="datetime64[D]")
                else:
                    dates = np.unique(cur_files["starttime"])
                params = []
                for date in dates:
                    one_day_cur_files = cur_files[cur_files["starttime"] == date]
                    one_day_ref_files = ref_files[ref_files["starttime"] == date]
                    one_day_cur_files = np.sort(one_day_cur_files, order=["station", "pair_name"])
                    one_day_ref_files = np.sort(one_day_ref_files, order=["station", "pair_name"])
                    if len(one_day_ref_files) != 0 and len(one_day_cur_files) != 0:
                        params.append([one_day_cur_files, one_day_ref_files, e, e_delta, plot, wave_durati
on, tt])
                if len(params) == 0:
                    sys.exit("##### Process: stretching_interpolation_v9 (stretching_interpolation)\nStatu
s: No params input")
                if(parallel== 0): n_worker = 1
                else: n_worker = multiprocessing.cpu_count()
                pool = multiprocessing.Pool(processes = n_worker)
                buff = list(pool.starmap(strecthing_interpolation_function, params))
                pool.close()
                pool.join()
                result.append(np.concatenate(buff))
    if (len(result) != 0):
        print("##### Process: stretching_interpolation_v9 (stretching_interpolation)\nStatus: Finish!! Tot
al output = {}".format(len(result)))
        print(type(result))
        return(result)
    else:
        print("##### Process: stretching_interpolation_v9 (stretching_interpolation)\nStatus: Finish!! No
output = {}".format(len(result)))

def run_velocity_estimation_mpi(root_out_folder, root_data_folder, cross_corr_folder_name,
        log_folder, data_cc_merge, output_starttime, output_endtime, ref_stack_starttime,
        ref_window_size, cur_window_size, stack_method, stack_folder_name, full_stack_window_output,
```

```
        cur_window_type, ref_window_type, plot, e, e_delta, parallel, wave_duration, stretching_folder_nam
e,
        mad_tolerance, mad_thresh_cc, mad_filename, median_filter_enable, E0_correction, output_folder):
    script_name = os.path.basename(__file__)
    comm = MPI.COMM_WORLD
    rank = comm.Get_rank()
    size = comm.Get_size()
    name = platform.node()
    #### setup log ###
    if rank > 0:
        log_filename  = open("/tmp/running.log",'w')
        sys.stdout =  log_filename
        sys.stderr =  log_filename
    if size == 0:
        sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: STOP ({})!!! \n Number of p
rocess != 1 \n\n".format(script_name, name))
    #Setup log folder
    if log_folder == None:
       log_folder = os.path.join(root_data_folder, "log")
    if os.path.isdir(log_folder) == False:
        os.makedirs(log_folder)
    time_folder = log_folder
    date = output_endtime
    process_check = os.path.join(log_folder, "{}.03.success".format(date))
    ########################## Check None Parameters ##############################
    if data_cc_merge == None:
        sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: STOP ({})!!! \nPlease give
data_cc_merge full path \n\n".format(script_name, name))
    if stack_folder_name == None:
        sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: STOP ({})!!! \nPlease give
stack_folder_name \n\n".format(script_name, name))
    if stretching_folder_name == None:
        sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: STOP ({})!!! \nPlease give
stretching_folder_name \n\n".format(script_name, name))
    if output_folder == None:
        sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: STOP ({})!!! \nPlease give
output_folder \n\n".format(script_name, name))
    ###### Setup folder ######
    stack_folder_path = os.path.join(root_out_folder, stack_folder_name)
    stretching_folder_path = os.path.join(root_out_folder, stretching_folder_name)
    ############ Load Daily Cross-correlation Data and Merge Into File Before Stacking ##########
    timing = [time.time()]
    proc_start = time.time()
    print("\n#### Process: {} (run_velocity_estimation_mpi) \n#### Status: Start!! Merge Cross-
correlation Data {}\n\n".format(script_name, name))
    dates = np.array([output_starttime, output_endtime, ref_stack_starttime], dtype="datetime64[D]")
    dates = np.arange(np.min(dates), np.max(dates)+np.timedelta64(1, 'D'), dtype='datetime64[D]')
    if rank == 1 or size == 1:
        if os.path.isfile(data_cc_merge):
            old_data_cc_merge = np.load(data_cc_merge)
            old_dates = np.sort(np.unique(old_data_cc_merge["starttime"]))
        else:
            old_dates = []
            old_data_cc_merge = None
        new_data_cc_merge = []
        for d in dates:
            if (d not in old_dates):
                dd = d.astype(np.str).replace("-", "")
                cc_folder_path = os.path.join(root_data_folder, dd, cross_corr_folder_name)
                if os.path.isdir(cc_folder_path):
                    print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Loading Daily Cros
s-correlation Data! {}\n\n".format(script_name, d))
                    buff = ddi(folderIn = cc_folder_path, file_pattern = "*cc.npy",  method="single-
folder", out_type="array-list", parallel = parallel)
                    new_data_cc_merge.append(np.concatenate(buff))
            else:
                print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: ({}) Data exist {}\n\n
".format(script_name, name, d))
        if new_data_cc_merge != []:
```

```
            new_data_cc_merge = np.concatenate(new_data_cc_merge)
        if type(old_data_cc_merge) == np.ndarray:
            if type(new_data_cc_merge) == np.ndarray:
                new_data_cc_merge = np.concatenate([old_data_cc_merge, new_data_cc_merge])
                np.save(data_cc_merge, new_data_cc_merge)
            elif new_data_cc_merge == []:
                new_data_cc_merge = old_data_cc_merge
        elif type(data_cc_merge) == str:
            print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: ({}) Saving The Merged CC
Data!! \n{}\n\n".format(script_name, name, data_cc_merge))
            np.save(data_cc_merge, new_data_cc_merge)
        if type(new_data_cc_merge) == np.ndarray:
            data_cc_merge = new_data_cc_merge
    ############################### Stacking Data ############################
        print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: {} Start Stacking Data! \n\n".
format(script_name, name))
        if stack_method == "ARM":
            stack_cur_ref_arm(data_in=data_cc_merge, out_dir=stack_folder_path,npy_list=0,
                              output_starttime = output_starttime, output_endtime = output_endtime,
                              ref_window_size = ref_window_size, ref_stack_starttime = ref_stack_startti
me, ref_window_type=ref_window_type,
                              cur_window_size = cur_window_size, full_stack_window_output=full_stack_win
dow_output, cur_window_type=cur_window_type)
        elif stack_method == "SRM":
            stack_cur_ref_srm(data_in=data_cc_merge, out_dir=stack_folder_path, npy_list=0,
                              output_starttime = output_starttime, output_endtime = output_endtime,
                              ref_window_size = ref_window_size, ref_stack_starttime = ref_stack_startti
me, ref_window_type=ref_window_type,
                              cur_window_size = cur_window_size, full_stack_window_output = full_stack_w
indow_output,  cur_window_type=cur_window_type)
        else:
            sys.exit("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Please define stack met
hod ('ARM' or 'SRM' \n".format(script_name))
    else:
        print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Start Stacking Data!! \nWaitin
g to finish\n\n".format(script_name))
    comm.barrier()
    timing.append(time.time())


    ########################### Stretching interpolation ######################
    print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Start Velocity Estimation! ({})\n\
n".format(script_name, name))
#         print("\n\n#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Finish Velocity Estimati
on! ({})\n\n".format(script_name, name))
    if rank != 0:
        data_in = ddi(folderIn = stack_folder_path, file_pattern = "*stack*", method="single-
folder", out_type="string-list", parallel = 1)
        buff = sorted(data_in)
        buff = np.array_split(buff, size-1)
        data_in = sorted(buff[rank-1])
        data_stretching = strecthing_interpolation(data_in=data_in, plot=plot, e=e, e_delta = e_delta, par
allel=1, wave_duration=wave_duration,
                                                   output_starttime = output_starttime, output_endtime = o
utput_endtime)
        ddo(data_in = data_stretching, proc_name="stretching", out_dir = stretching_folder_path, method="s
plit")
        print("\n\n#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Finish Velocity Estimation
! ({})\n\n".format(script_name, name))
    elif rank == 0:
        print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Start Velocity Estimation! ({}
)\n\n".format(script_name, name))
    comm.barrier()
    timing.append(time.time())
    ########################### Optimize Velocity (Removing Outlier Anomaly, E0 Correction) ##############
#########
    output_file_path = os.path.join(output_folder, mad_filename)
    if rank == 0:
```

```
        print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: START! Optimize Velocity! \n {
}\n".format(script_name, output_file_path))
        data = ddi(folderIn = stretching_folder_path, file_pattern = "*stretching.npy", method="single-
folder", out_type="array-list", parallel = 0)
        data = np.concatenate(data)
        data = optimize_velocity(data_in = data, mad_tolerance=mad_tolerance, thresh_cc = mad_thresh_cc,
                                 median_filter=median_filter_enable,  E0_correction=E0_correction)
        if os.path.isdir(output_folder):
            shutil.rmtree(output_folder)
        os.makedirs(output_folder, exist_ok=True)
        np.save(output_file_path, data)
        print("#### Process: {} (run_velocity_estimation_mpi)\n#### Status: FINISH! Optimize Velocity Proc
ess!!\n {}\n".format(script_name, output_file_path))
        timing.append(time.time())
    proc_time = proc_start - time.time()
    ###Save log file status
    if rank == 0:
        print(process_check)
        process_check_file = open(process_check, "w").write("{} {} {} {}\n".format(date, timing, proc_time
, size))
    print("\n\n#### Process: {} (run_velocity_estimation_mpi)\n#### Status: Finish ALL Process! ({})\nFold
er : {}\n".format(script_name, name, output_file_path))

if __name__ == "__main__":
    today_date = sys.argv[1]
    ######################### Input Paramameters ##########################
    # Folder Configuration
    root_out_folder = "/home/fernando/hinet/data/kyushu"
    root_data_folder = "/home/fernando/hinet/data/kyushu"
    log_folder = "/home/fernando/hinet/log"
    time_folder  = log_folder
    cross_corr_folder_name = "cross_corr"
    # Stack Parameters
    output_starttime = None
    output_endtime = today_date
    ref_stack_starttime = None
    ref_window_size = 365
    cur_window_size = 11
    stack_method = "SRM"
    full_stack_window_output = 1
    stack_folder_name = "SRM_{}_stack_{}".format(cur_window_size, today_date)
    data_cc_merge = "/home/fernando/hinet/data/kyushu/merge_data_cc.npy"
    # Velocity Estimation Parameters
    plot = 0
    e = 0.025
    e_delta = 0.0005
    parallel=1
    wave_duration=100
    stretching_folder_name  = "SRM_{}_stretch_{}".format(cur_window_size, today_date)
    #Velocity Optimize Parameters
    mad_tolerance = 3
    mad_thresh_cc = 0.5
    mad_filename = "all_mad.npy"
    median_filter_enable = 1
    E0_correction=1
    optimize_vel_folder = "/home/fernando/hinet/data/kyushu/temporal_vel/{}".format(today_date)
    ######################### Stacking, And velocity estimation ######################### _
    run_velocity_estimation_mpi(root_out_folder = root_out_folder, root_data_folder = root_data_folder,
            cross_corr_folder_name = cross_corr_folder_name,
            output_starttime = output_starttime, output_endtime = output_endtime,
            ref_stack_starttime = ref_stack_starttime, ref_window_size = ref_window_size,
            cur_window_size = cur_window_size, stack_method = stack_method,
            stack_folder_name = stack_folder_name, full_stack_window_output = full_stack_window_output,
            plot = plot, e = e, e_delta =e_delta, parallel=parallel, wave_duration=wave_duration,
            stretching_folder_name = stretching_folder_name, data_cc_merge = data_cc_merge,
            log_folder =log_folder, mad_tolerance = mad_tolerance, mad_thresh_cc = mad_thresh_cc,
            mad_filename = mad_filename, median_filter_enable = median_filter_enable,
            E0_correction = E0_correction, output_folder = optimize_vel_folder)
```