Comparison of Stacking Methods Regarding Processing and Computing of Geoscientific Depth Data

Claus-Peter Rückemann
Leibniz Universität Hannover,
Westfälische Wilhelms-Universität Münster (WWU),
North-German Supercomputing Alliance (HLRN), Germany
Email: ruckema@uni-muenster.de

***Abstract*—This paper presents a comparison of different stacking methods available for processing and computing of geoscientific depth data for use with various high level applications and computing architectures. These methods are used with seismics and comparable geophysical techniques for example. Todays resources enable to use these methods for more than batch processing. For these cases it is important to analyse the algorithms regarding strengths and implementation benefits. The algorithms presented have been successfully implemented and evaluated for their purpose with application scenarios using High End Computing (HEC) resources. The focus is on integrating stacking algorithms in information and computing systems, utilising Distributed Computing and High Performance Computing (HPC) from integrated systems for use in natural sciences disciplines and scientific information systems.**

***Keywords*–Processing; Scientific Computing; Stacking; Comparison; Seismics; Geosciences; Depth Data.**

I. INTRODUCTION

Processing of geoscientific depth data does have a long and successful history and evolution. Decades of methods development, numerical algorithms, and implementation of processing and visualisation systems have been needed to understand how to reveal and analyse some essential information of depth sequences and profiles, from the work built by the complex geological and geophysical processes of millions of years. Processing of depth data has always been very data and computing intensive, so there is no focus on geo-data processing itself in this paper. Computing architectures have been available for decades but these have been quite limited regarding computing power and therefore resulted in long processing time intervals, in many cases to weeks and months, even for single profiles. This has restricted applications and algorithms to batch processing and rarely interactive applications have been reasonable. Classical use and applications are known from published use cases [1]. With the modern parallel architectures many interactive and dynamical applications, Active Source, and InfoPoints get into the focus [2]. In the early 1990s the advanced superstack algorithm has been developed [3]. Even the full vectorisation of the algorithms has resulted in days of processing time on VAX mainframe and Unix machines, even for small parts of profiles, and with increased resolution and data size computing times have not been reduced today. With parallel architectures several processing algorithms are undergoing parallelisation efforts, meaning parallelisation regarding data sets, algorithms, iterations, and so on. These algorithms are for example part of stacking methods, migration methods, Fresnel Section [4] calculation as well as elementary algorithms like Fast Fourier Transformations (FFT) and many more, for seismic software [5], [6] as well as for dynamical information system components [7], [8].

This paper is organised as follows. Section two presents motivation and complexity with the implementation. Section three introduces stacking, basic terms, and challenges. Section four shows the different stacking methods and algorithms. Section five and six discuss and evaluate the methods regarding application and Sections seven and eight summarise the lessons learned, conclusions, and future work.

II. MOTIVATION

In most cases geoscientific and geophysical algorithms are used in conventional batch applications. None of these have been integrated into interactive information system components so far. The reason is that computing power is limited for up-to-date applications and resolutions and that parallelisation would not be possible for local or standalone computing systems. Parallelisation, loosely and embarrassingly parallel, of geoscientific algorithms will help to support new application scenarios, for example dynamical interactive information and computing systems for geosciences and environmental sciences. With the implementation use cases for Information Systems the suitability of Distributed and High Performance Computing resources supporting processing and computing of geoscientific data have been studied. These use cases have focus on event triggered communication, dynamical cartography, compute and storage resources. The goal has been, to bring together the features and the experiences for an integrated information and computing system. An example that has been implemented is a spatial information system with hundreds of thousands of ad-hoc simulations of interest. Within these interactive systems depth information may play an important role as “next

 ISBN: 978-1-61208-178-6

informations of interest", being dynamically calculated in parallel. Due to the complexity of integrated information and computing systems, we have applied meta-instructions and signatures for algorithms and interfaces. For these cases, envelopes and IPC has been used to provide a unique event and process triggered interface for event, computing, and storage access.

III. Stacking, Terms, Goals, and Challenges

Stacking is an essential part of seismic data processing. The primary goal of seismic stacking methods is the enhancement of the Signal-to-Noise Ratio (SNR) of the data material. Increasing demands for high resolution and true amplitudes, and allowing interpretation of amplitude ratios have led to seismic stacking methods [9]. The basic stacking methods and references have been collected and described [3]. Stacking and migration are the central methods for discovering and defining slanted crustal boundary layers [10]. Stacking is done in order to to reduce Common Mid-Point (CMP) gathers into one trace. The appropriate corrections for statics and Normal-Moveout (NMO) should be done in all cases where advanced precision is necessary. The description of standard NMO correction and CMP methods can be found in all common textbooks. In some cases stacking methods are used in order to combine groups of traces other than CMP groups. For example in vertical stacking from repeated shots traces of sequences of depth points are combined. Nearly all stacking methods commonly used are phase stacks. The term Maximum Coherency Stack is sometimes used to point out explicitely that maximum coherence of different traces is achieved by moving them in direction of the time axis, for example by NMO correction. Only in rare case stacking in the frequency domain is used, for example the envelope stacking. All stacking methods obtain their significance by experience, not primarily by theoretical deliberations [9]. Along with improving the SNR, stacking reduces disturbing events and energy in the data. These disturbances can be called "unwanted energy" [9]. Although this term depends on the situation, in most cases it means the following effects:

- Multiples energy,
- Refracted energy,
- Uncorrelated noise,
- 'Noise bursts' with large amplitude,
- Cable noise.

In many cases randomly distributed energy is the target. Suitable assumptions for noise regarding stacking strongly depend from the preprocessing and much less from the post-stack processing. For that, stacking methods are most desirable, that make a distinction between signal and coherent noise. It is also desirable to retain the signal form and amplitude with the stacking algorithm as these contain physical and geological information but in many cases the focus is on recognition of primary signals especially with a very small SNR. Besides the goals there are various challenges making use of stacking algorithms with different application scenarios. The consequences of the properties and strengths of different stacking methods as well as the very different processing and computing requirements has made it neccessary considering operational areas with complex systems, suitable for scientific and educational purposes. Which methods can be applied for the purposes of dynamical processing of depth data or only used for precomputation of depth data? Different application scenarios need different stacking methods, especially this holds true for informations system and expert system components for which individual processing decisions and dependencies are not feasible in most cases. With this we need to know the architecture of the algorithms, specialisation, and strengths.

IV. Comparison of different stacking methods

The following sections give a short comparison of available stacking methods and show their strengths and possible field of operation in the context of complex application scenarios. Although each principle is characteristic, variations of the methods are applicable.

A. Straight Stack, Mean Stack

This stacking method is the most simple one [11]. This method is a special, simple case of more common stacking methods like the Superstack or the Trimmed Mean Stack. Nevertheless, many higher level considerations are based on it [12]. The Straight Stack sums up the sample amplitude values at the isochrone locations and divides by the number of values, for all channels to be processed:

$$a_t^{StraightStack} = \frac{1}{N} \sum_{i=1}^{N} S_i \tag{1}$$

N is the number of isochone values, S_i the amplitude at a sample location, and $a_t^{StraightStack}$ is the amplitude of the stacked trace at a respective time.

B. Stacking with predefined weighting

1) Weighting by muting: The amplitude values at every sample in the gather are assigned with a weighting, a value 1 or 0. If this hold true for a threshold value this is a mute function. For strong multiples like the ocean bottom multiples an inner-trace-mute can be reasonable.

2) Weighting as function of the offset: This is comparable to an inner-trace-mute and is used to increase the stacking response of primary reflections. Far offset traces get a higher weighting where multiples get out of phase. Weights are calculated from velocity ratios.

3) Weighting as function of the array response: Based on the ratio of overall response of the array system relative to response of the primary signals plus multiples. Used in cases of water coverage absorbing too much energy, important for strong multiples scenarios

 ISBN: 978-1-61208-178-6

C. Stacking with data adaptive weighting

1) Optimum Weighted Stack (OWS), Weighted Stack: Based on optimum criteria this algorithm is used before summation, including optimum stacking filters. The algorithm for the stacking value s_t^{OWS} is based on:

$$r_{j,t} \quad \text{with} \quad j = 1, 2, \ldots, J \quad \text{and} \quad t = 1, 2, \ldots, T \tag{2}$$

$$s_t^{OWS} = \sum_{j=1}^{J} (w_j r_{j,t}) \tag{3}$$

where $r_{j,t}$ represents the traces to be stacked, w_j the weights, T the number of samples per trace and J the number of traces per CMP gather.

2) Diversity Stack (DS): The result of the DS is the amplitude variation of the input data:

- Subdivide trace into time windows.
- Calculate the overall energy per window:
$$E = \sum_{\Delta T} \left(a_t^2\right) \tag{4}$$
with amplitude a_t at a sample and window length ΔT.
- Calculate scale factor $D = C/E$ with $C = const.$ for each window,
- Determine gain trace, scaling factor D, this can be a selected trace or every trace for itself.
- Scale trace by application of gain trace to the original trace using cross multiplication.
- Summation for scaled traces and gain traces. The sum of scaled traces is divided by the sum of gain traces.

3) Coherency Stack (CS): The application of the CS is: Choose windows for NMO corrected CMP gather. calculate coherency values for windows using coherency measure like semblance, coherency model trace can be calculated by sorting and interpolation relative to subsequent central values. Coherency stacking is applied by addition of choosen percentages of the coherences model trace on the conventional stacked trace.

D. Iterative and comparable methods

The term iterative stacking is often used synonym to the term Superstack. It is especially important with iterative methods to take care of reflected signals with phase reversal on far offset. This will require preprocessing or reduction to nearer offset when optimising the SNR.

1) Iterativer Stack, Superstack (IS, SS): Iterative stacking with the Superstack [13] is based on separating Amplitudes, positive a_j and negative b_j, for all reflection times with amplitude r_j. The Number of iteration is n, after the first iteration for the data matrix holds $n = 1$. The norm factor M for sums after an iteration is called multiplicity. In basic form the algorithm is described by:

$$a_j^n = \begin{cases} r_j & \text{for} \quad r_j > 0 \\ 0 & \text{for} \quad r_j \leq 0 \end{cases} \tag{5}$$

$$b_j^n = \begin{cases} 0 & \text{for} \quad r_j \geq 0 \\ r_j & \text{for} \quad r_j < 0 \end{cases} \tag{6}$$

$$s_+^n = \frac{1}{M} \sum_{j=1}^{J} a_j^n \qquad \text{and} \qquad s_-^n = \frac{1}{M} \sum_{j=1}^{J} b_j^n \tag{7}$$

with sums s_+^n and s_-^n of isochrone positive and negative amplitudes at a respective time and NMO corrected amplitudes r_j in the CMP gather, with

$$r_{j,t} \quad \text{with} \quad j = 1, 2, \ldots, J \quad \text{and} \quad t = 1, 2, \ldots, T \tag{8}$$

The weighting of a single amplitude value at a time sample is done using:

$$\begin{array}{llll} a_j^{n+1}{=}s_+^n \text{ for } a_j^n{>}s_+^n & \text{and} & a_j^{n+1}{=}a_j^n \text{ for } a_j^n{\leq}s_+^n \\ b_j^{n+1}{=}s_-^n \text{ for } b_j^n{<}s_-^n & \text{and} & b_j^{n+1}{=}b_j^n \text{ for } b_j^n{\geq}s_-^n \end{array} \tag{9}$$

If I is the overall number of samples $i = 1, \ldots, I$ of a trace, stacking can be described using the matrix of the original data set $C_{I,J}$, utilising n iterations for the modified matrix $C_{I,J}^n$ to the resulting stacked trace $S_{I,1}$ with the following formula. Separation in positive and negative amplitude sums is done for every iteration.

$$\begin{bmatrix} c_{1,1} & \cdots & c_{1,J-1} & c_{1,J} \\ c_{2,1} & \cdots & c_{2,J-1} & c_{2,J} \\ c_{3,1} & \cdots & c_{3,J-1} & c_{3,J} \\ \vdots & \ddots & \vdots & \vdots \\ c_{I-1,1} & \cdots & c_{I-1,J-1} & c_{I-1,J} \\ c_{I,1} & \cdots & c_{I,J-1} & c_{I,J} \end{bmatrix} \tag{10}$$

$$\Big\downarrow \; n \text{ iterations}$$

$$\begin{bmatrix} c_{1,1}^n & \cdots & c_{1,J-1}^n & c_{1,J}^n \\ c_{2,1}^n & \cdots & c_{2,J-1}^n & c_{2,J}^n \\ c_{3,1}^n & \cdots & c_{3,J-1}^n & c_{3,J}^n \\ \vdots & \ddots & \vdots & \vdots \\ c_{I-1,1}^n & \cdots & c_{I-1,J-1}^n & c_{I-1,J}^n \\ c_{I,1}^n & \cdots & c_{I,J-1}^n & c_{I,J}^n \end{bmatrix} \xrightarrow{\sum_{j=1}^{J}} \begin{bmatrix} s_{1,1}^n \\ s_{2,1}^n \\ s_{3,1}^n \\ \vdots \\ s_{I-1,1}^n \\ s_{I,1}^n \end{bmatrix} \tag{11}$$

For reduction of the polarity of a small number of amplitudes a Higher Degree Stacking (HDS) can be done, separating positive and negative amplitudes and exponentiating it with the degree of the stack.

2) Single Trace Iterative Stack (STIS): This is an alternative application of the above Superstack algorithm [14]. With many applications the term Near Trace Iterative Stack is a good description as increased weighting is on near offset traces and not on a single trace.

3) Iterative Weighted Stack (IWS): The IWS algorithms is used the following way:

- A CMP gather is stacked (Straight Stack) into one trace (pilot trace).
- At every sample the amplitude variance along the gather is calculated. Often the mean amplitude value from the pilot trace is used.

 ISBN: 978-1-61208-178-6

- Weights are calculated (Gauß distribution) for every sample, the mean value from b) is used. For every trace the weighting is smoothed with the time. For every sample time weights are normed along the gather.
- The weighted stacking is calculated and 1) used as finally stacked trace or 2) defined as new pilot trace and the process is iterated from b) on.

4) The N^{th}-Root Stack (NRS): The NRS is not an iterative method but it is based on a comparable principle. In the most simple form the NRS can be written in the following form for calculating a stacking element s_t^{NRS}.

$$s_t^{NRS} = \left[\frac{1}{J} \sum_{j=1}^{J} sgn(s_{j,t}) \mid s_{j,t} \mid^{\frac{1}{N}} \right]^N \quad (12)$$

$$\text{for} \quad t = 1, 2, \ldots, T \quad \text{with} \quad sgn(s_{j,t}) = \frac{s_{j,t}}{\mid s_{j,t} \mid} \quad (13)$$

with the number of traces J in the CMP gather and N a number 2^P, with $P \in \text{IN}$. In most cases $N = 2$ or $N = 4$ are used. There are various ways of application [15].

E. Other Methods

1) Median Stack (MS), Alpha-Trimmed Mean Stack (ATMS): The median amplitude values from traces to be stacked are picked. The stacked trace contains for every sample the median value at the same time with the amplitudes along the CPM gather. Thus the MS stack does not result from summed up values. Given amplitudes a_i with $i = 1, 2, \ldots, J$, and J is an odd integer value, the ATM a_α can be calculated as:

$$a_\alpha = \frac{1}{J - 2L} \sum_{i=L+1}^{J-L} a_i \quad (14)$$

Trimming is done by choosing the value $L = \alpha J$, where α is the trimming parameter ($0 \leq \alpha < 0.5$). This means with the ATMS it is possible to do a step wise and weighted combination of Straight Stack for $\alpha = 0$ and Median Stack for $\alpha \rightarrow 0.5$ for stacking. The MS result can appear like adding high-frequency noise. This is reduced by summing up more than one amplitude, which after resorting the input values follow in rising sequence around the central position. It can be used to exclude extreme amplitude value groups from the stack. This is done using the ATM method.

2) Trimmed Mean Stack (TMS): The TMS can be described by the following algorithm: The amplitudes of a gather are sorted by value, numbered, and summed up at a time using the values up to a defined amplitude number. The summation for non symmetrical elimination of extreme amplitudes can be performed as:

$$a^{TMS} = \frac{1}{N - K} \sum_{i=\frac{K}{2}+1}^{N-\frac{K}{2}} S_i \quad (15)$$

with number N of samples, overall number K of excluded sample values, the amplitude S_i at the respective sample, and the amplitude a^{TMS} of the stacked trace at the respective time. The TMS is a generalisation of the Straight Stack.

3) Minimum/Maximum Sample Value Exclusion Stack: The “Min/Max Stack” excludes samples with the largest and smallest amplitudes from the stacking.

4) Signed Bit Stack (SBS): The SBS adds $+1.0$ to the stacked sum if the absolute amplitude value at a sample is positive or zero and -1.0 if the amplitude value is negative.

5) Random Stack (RS): The RS can be described with:

a) Given a NMO corrected gather:

$$r_{j,t} \quad \text{with} \quad j = 1, 2, \ldots, J \quad \text{and} \quad t = 1, 2, \ldots, T \quad (16)$$

For determination of random sample trace values for every discrete sample time a random value $k(t)$ is picked from the J values of the corrected gather. The amplitude of the random value is $r_{k(t),t}$. The result of the process is a Constant Velocity Gather (CVG).

b) The traces of the CMP gather are combined in one trace: For every sample time amplitudes from the random traces are stacked, if they show they same sign. In the other case the output trace is set to zero. The result is a Constant Velocity Stack (CVS).

V. Discussion

There are several consequences and conditions for geoscientific interpretation and integrated systems, resulting from the different stacking algorithms described.

Precondition of the Straight Stack is uncorrelated noise and amplitudes and SNR comparable in magnitude over the traces. Otherwise Optimum Weighted Stack is an option. Normalisation the results using a scalar is used in praxi, e.g., root power scalar for stack normalisation. It is of common use, computation depends on size and dimensions of data but is easy to implement and can be widely parallelised.

Weighted stacking methods are important for very long source arrays, whose effect can be considered by calculation of the special geometrical parameters for the ‘predetermined’ weighting. They rely on data analysis a part of the processing. The result of the OWS depends on two main criteria, the adaption of the model to the data and the precision of calculating or estimating the signal amplitudes and the SNR. The DS is best suited for excluding high noise levels, especially noise bursts. In use are Diversity Power Stack and Diversity Amplitude Stack. It can reduce interference noise and is mostly used on land data. The CS reduces the effect of strong, non-coherent amplitudes in the final stack. It is mostly used on post-stack (Straight Stack). Relative amplitudes and therefore resolution can be disturbed.

The Superstack can be most useful for very low SNR. Very large or small abnormal amplitudes, that rarely occur in lateral view regarding isochrone samples, are reduced. In tendency amplitudes are reduced towards the smallest

 ISBN: 978-1-61208-178-6

absolute value, the stacking converges. With increasing number of iterations frequency spectra of single traces result in higher frequencies. The number of iteration should be suitably small in order to minimise disturbance of the wave form, in case the wave form is relevant. This is less relevant with a high noise percentage on longer travel times, when looking for appearent velocities near infinite. A large number of traces present can help correct this. Due to the algorithm of the Superstack/HDS, the large computational requirements can be encountered by vectorisation and parallelisation. STIS is mostly used when primary events in recorded traces to be summed up are not equal. The IWS is used for enhancing the velocity discrimination and reduction of multiples. The NRS is used in order to eliminate false alarms from strong noise amplitudes in single channels. NRS is most helpful being used on reflection data. The effect and the modification of the wave form is comparable to that of the Superstack. With most data there is less need for parallelisation but efficiency will profit nevertheless.

The significance of the MS/ATMS results from the fact that it removes all abnormal strong noise amplitudes that may occur with a small number of input traces and it is only less impacted by partly coherent signals that appear at the same time with the primary signal in less than half of the number of traces. With the TMS a given percentage of extreme amplitude values are excluded from the stack. This can be used to reduce the noise quota, as from interfering seismics events or experiments at the same time. Thus the TMS can, in extreme situations, result in decreased SNR compared to conventional stacking, when applied on data without significant noise. The Min/Max Stack is mostly used depending from the data and array facts, it most suitable for manual application. The SBS is a simple algorithm for excluding extreme values from stacking, without considering the real amplitudes. This method is only suitable with a large number of stackable traces. In other case even low noise quota will result in loss of information for interpretation. The SBS is most suitable for manual application. The RS is an alternative to conventional NMO corrected stacking. It can be used for velocity analysis. The ides of the RS is that is all primary amplitudes are the same, they should reside where the NMO corrections for the primary reflections are ideal. If signals are not all in phase the signal form can be destroyed. The RS destroys the noise signal form significantly whereas the primary reflections are conserved.

VI. Evaluation

Table I shows the matrix resulting for the investigated stacking methods and applicability regarding processing and computing with integrated systems for defined qualities:

- A: common method, batch use;
- B: combined use, mostly depending on other methods and pre- and post-processing;
- C: increased processing and resources requirements;
- D: parallelisation for the overall application;
- E: automation with integrated systems and workflows.

Table I
Stacking methods and qualities.

Stacking Method	*A*	*B*	*C*	*D*	*E*
Straight Stack, Mean Stack	✓	–	–	✓	✓
Stacking with predefined weighting					
by Muting	✓	✓	–	–	–
as Function of the Offset	✓	✓	–	–	–
as Function of the Array Response	✓	✓	–	–	–
Stacking with data adaptive weighting					
(Optimum) Weighted Stack	✓	✓	✓	–	–
Diversity Stack	✓	✓	✓	–	–
Coherency Stack	✓	✓	✓	–	–
Iterative and comparable methods					
Iterative Stack, Superstack, HDS	✓	–	✓	✓	✓
Single Trace Iterative Stack	✓	–	✓	✓	✓
Iterative Weighted Stack	✓	✓	✓	✓	✓
The N^{th}-Root Stack	✓	–	–	–	–
Other methods					
Median Stack, Alpha-Trimmed Mean	✓	✓	–	–	–
Trimmed Mean Stack	✓	✓	–	–	–
Min/Max Stack	✓	✓	–	–	–
Signed Bit Stack	✓	✓	✓	–	–
Random Stack	✓	✓	–	–	–

The stacking methods provide various algorithms handling signals and noise, suitable for the different nature of data. This goes along with different array properties as well as with the special attributes of the measurement. For all stacking methods discussed, errors in the calculated weighting can result in stacks that can be less suitable than the Straight Stack. They do have different geophysical focus and should not be seen competitive. The named stacking methods are mostly empirical and not thought as strong mathematical consequence of presumptions made [9]. Nevertheless some methods like the iterative Superstack algorithm use statistical and empirical information for the processing. The third category besides the "qualities" with integrated systems and their suitablity for geo-conditions is the order of magnitude for the number of instances (Table II) that will be used with one interactive application.

Table II
Number of instances with one interactive application task.

Instances	*Straight*	*Weighted*	*Iterative*	*Other*
Single	✗	✗	✗	✗
Several	✗	✗	(✗)	✗
Many	✗	(✗)	–	(✗)

The case studies have shown which compute intensive methods can be used with several instances per application, simpler methods may easily be used with thousands.

VII. Lessons Learned

Regarding the results, we can divide the methods into several main groups for use with different application scenarios. Common use: Batch use, this will work for mostly

 ISBN: 978-1-61208-178-6

all methods. Human interaction and semi-manual use, like with the SBS and Min/Max Stack. This group is of common interest. Combined use: Methods demanding for combined use with additional methods like migration or various pre- and post-processing. These methods will be difficult to be integrated in automated processes or workflows. Integration into special application scenarios: Focus on iteration parallelisation, Straight Stack, Superstack, Fresnel Section support. These are especially interesting for future multi-dimensional measurement and processing. If the complexity is demanding large data or compute intensive processing this can be solved using the appropriate HEC architectures with integrated systems. The integration of secondary –geological and environmental– information with the workflow does provide benefits for the interpretation efficiency.

VIII. Conclusion and Future Work

The analysis of the different stacking methods has demonstrated that the applicability of the algorithms is very different for application scenarios regarding processing and computing with integrated systems and can be categorized by properties and purpose. Stacking can be used for integrated information systems as with dynamical concepts. For example if there is only uncorrelated noise the Straight Stack or Optimum Weighted Stack is most effective. If the interpretation needs to process data regarding increased sharpness in time in order to localise reflection elements, for deep structures this increasingly correlates with sharpness in space, then a Superstack can be the means of choice, even when it goes along with lateral smoothing of the trace domain. The application allows a smooth segmentation for using distributed resources. Some implemented methods have proved useful for application of dynamical processing. The Straight Stack can be efficiently integrated for standard information, loosely coupled parallel Superstack and HDS for depth analysis and low SNR, not concentrating on signal form. The implementation is computing intensive and has been vectorised for vector architectures and parallelised for parallel architectures and is as well suitable for automation and integration if appropriate HEC and HPC resources are available. Most of the other methods are suitable with stepwise workflows. In the future, the integration of suitable methods for depth data processing with information systems will be done according to these results.

Acknowledgements

I am grateful to all my scientific colleagues at the Westfälische Wilhelms-Universität Münster and the Leibniz Universität Hannover for their support as well as to international academic and industry partners in GEXI collaborations for innovative constructive test beds, IRI, HLRN, ZIV Münster, D-Grid, and the participants of the postgraduate EULISP Programme for prolific discussion and participation in case studies on integrated systems over the last years.

References

[1] K. Waters, *Reflection Seismology, A Tool for Energy Resource Exploration*, 3rd ed. A Wiley-Interscience Publication, John Wiley and Sons, N. Y., Chilchester, Brisbane, Toronto, Singapore, 1987.

[2] C.-P. Rückemann, "Implementation of Integrated Systems and Resources for Information and Computing," in *Proceedings of the International Conference on Advanced Communications and Computation (INFOCOMP 2011), October 23–29, 2011, Barcelona, Spain*. XPS, Xpert Publishing Solutions, 2011, Rückemann, C.-P. (ed.), 6 pages, ISBN: 978-1-61208-161-8.

[3] C.-P. Rückemann, "Anwendung verschiedener Stapelmethoden auf gering überdeckende reflexionsseismische Daten aus der Heimefrontfjella, Antarktis," Diplomarbeit, Institut für Geophysik, Westf. Wilhelms-Universität Münster, 1994.

[4] C.-P. Rückemann, "Application and High Performant Computation of Fresnel Sections," *Symposium on Advanced Computation and Information in Natural and Applied Sciences, Proceedings of the 9th International Conference on Numerical Analysis and Applied Mathematics (ICNAAM), September 19–25, 2011, Halkidiki, Greece, AIP Conference Proceedings 1389, Proceedings of the American Institute of Physics (AIP)*, 2011, ISBN: 978-0-7354-0956-9, URL: `http://proceedings.aip.org/` [accessed: 2011-08-21].

[5] C.-P. Rückemann, "Software Seismic Workshop (SSW)," *[Internet]*, 1995, URL: `http://www.user.uni-hannover.de/cpr/x/rprojs/en/index.html#SSW` (Project information).

[6] "Seismic Unix," 2011, Center for Wave Phenomena, URL: `http://www.cwp.mines.edu/cwpcodes/` [accessed: 2011-08-21].

[7] "Applications with Active Map Software, Screenshots," *[Internet]*, 2005, URL: `http://wwwmath.uni-muenster.de/cs/u/ruckema/x/sciframe/en/screenshots.html` [accessed: 2011-02-20].

[8] "Geo Exploration and Information (GEXI)," 1996, 1999, 2010, 2011, URL: `http://www.user.uni-hannover.de/cpr/x/rprojs/en/index.html#GEXI` [accessed: 2011-08-21].

[9] O. Naess and L. Bruland, *Stacking Methods other than simple Summation, in: Developments in Geophysical Exploration Methods*, F. (ed.), Ed. Elsevier Applied Science Publishers, 1985, vol. 6.

[10] K. Bullen and B. Bolt, *An Introduction To The Theory Of Seismology*. Cambridge Univ. Press, Cambridge, 1963, 1985.

[11] W. Mayne, "Common reflection point horizontal data stacking techniques," *Geophysics, (Classic Papers), GE 50-11-1856*, vol. 50, no. 11, 1962.

[12] W. Mayne, "Practical considerations in the use of common reflection point techniques," *Geophysics, GE 32-02-0225*, vol. 32, no. 2, p. 225, 1967.

[13] O. Naess, "Superstack – an iterative stacking algorithm," *Geophysical Prospecting, GP 27-01-0016*, 1979.

[14] O. Naess, "Single-trace processing using iterative cdp-stacking," *Geophysical Prospecting, 30, GP 30-05-0641*, pp. 641–652, 1981.

[15] E. Kanasewich, C. Hemmings, and T. Alpaslan, "Nth-root stack nonlinear multichannel filter," *Geophysics*, vol. 38, pp. 327–338, 1973.

 ISBN: 978-1-61208-178-6