IMPROVEMENT OF THE COMMON-REFLECTION-SURFACE STACK BY DIFFERENTIAL EVOLUTION AND CONFLICTING DIP PROCESSING

J. Walda and D. Gajewski

email: *jan.walda@uni-hamburg.de*
keywords: *diffractions, partial CRS, conflicting dips, CRS, optimization*

ABSTRACT

The common-reflection-surface method stacks more traces than the common-midpoint stack since neighboring midpoints are taken into account as well. This means, rather than picking one parameter, a three parameter search is necessary. Most implementations estimate an initial guess and search locally for all three parameters together. We use a global optimization scheme instead where a priori information can be used as additional input or constrains. Furthermore, we split the whole search space in smaller parts with respect to the angle parameter so that conflicting events at a specific sample can be recognized in the stack and attributes. The CRS attributes are used for diffraction separation and prestack data enhancement using partial CRS to show the potential to regularize and interpolate the whole wave-field rather than mostly reflections as previously. Results for marine data prove this conclusion. We also applied this data-driven approach to land data to reveal the improvements achievable even in challenging data.

INTRODUCTION

The Common Reflection Surface (CRS) stack (Müller, 1999; Jäger et al., 2001; Mann, 2002) increases the signal-to-noise ratio significantly and its attributes can be used for further applications like diffraction separation (Dell and Gajewski, 2011), pre-stack data enhancement (Baykulov and Gajewski, 2009), traveltime decomposition (Bauer et al., 2015), prestack slope determination (Schwarz et al., 2015) or data-driven pre- and poststack time migration (Bobsin et al., 2015). However the treatment of conflicting dips during the parameter estimation of the CRS operator (Mann, 2001; Müller, 2009) is not reliable and often leads to suppression of the less dominant events. An alternative approach was proposed by Höcht et al. (2009), who interpolates parameters at an arbitrary position from previously determined traces. Since multiple traces contribute to the target trace, conflicting dips can be imaged but this approach does not enable conflicting dips in attribute based methods. Similar to the dip-moveout (DMO) Soleimani et al. (2009) proposed to stack along all angles, thus fixing the angle and estimating a mixed curvature parameter called R_{CDS}. This approach uses the CRS operator for diffractions which lacks information about either R_{NIP} or R_N depending on the choice of aperture. Walda and Gajewski (2014b) extended this idea to the full CRS and i-CRS operator and encountered signal stretch. Walda and Gajewski (2014a) showed that a global optimization scheme can help to improve the CRS parameter estimation when no velocity information is available. This can also solve the problem of the signal stretch in combination with a better description of the parameter space as shown by Walda and Gajewski (2015).
In this work we describe the algorithm in detail, investigate parameter quality and show field data applications for marine and land data-sets.

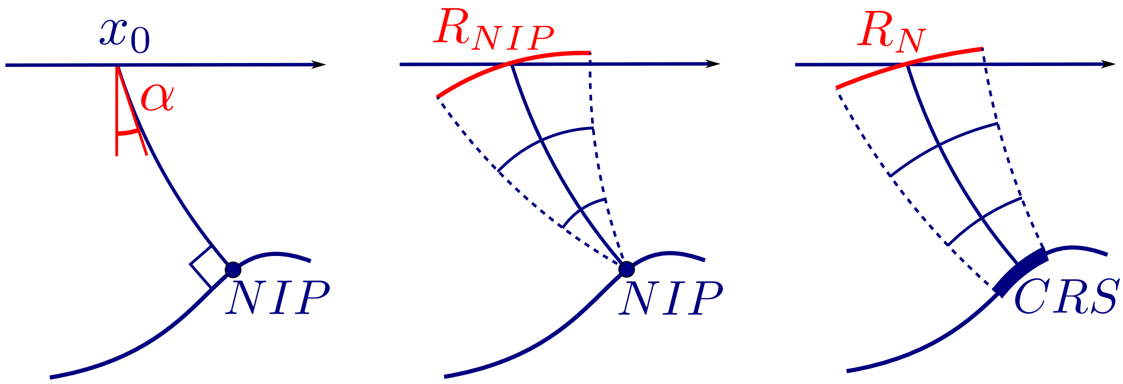

Figure 1: Two hypothetical experiments: The normal-incidence-point wave is shown in the middle with its radius of curvature R_{NIP}. The normal wave with the radius of curvature R_N, is caused by an exploding reflector experiment (right). Both have the angle of emergence α (left) which is attached to the NIP-ray.

COMMON-REFLECTION-SURFACE STACK

The CRS stack is a multi-parameter stacking technique that considers neighboring midpoints as well as the offset while the Common Midpoint (CMP) method uses only offsets. Therefore, more traces are stacked and the signal-to-noise ratio is improved significantly.
The CRS operator consists of three wave-field attributes, which are related to two hypothetical one-way experiments as shown in figure 1. The resulting two waves are described by the angle of emergence α of the ZO ray and the corresponding radii of curvature: R_N for the normal (N) wave and R_{NIP} for the normal-incidence-point (NIP) wave (Hubral, 1983). The N wave is generated by an exploding reflector model around the normal-incidence-point. The NIP wave is generated by a point source at the normal-incidence-point for a specific reflector.
The CRS formula in its hyperbolic variant is given by:

$$t^2(x_m, h) = \left(t_0 + \frac{2\sin\alpha}{v_0}\Delta x_m\right)^2 + \frac{2t_0\cos^2\alpha}{v_0}\left(\frac{\Delta x_m^2}{R_N} + \frac{h^2}{R_{NIP}}\right) \tag{1}$$

where $\Delta x_m = x_m - x_0$ is the midpoint displacement, h the half-offset, t_0 the two-way traveltime (TWT) of the zero-offset (ZO) ray and v_0 the near surface velocity.

DIFFERENTIAL EVOLUTION

Differential evolution (DE) is a meta-heuristic optimization algorithm that iteratively optimizes an objective function, without any assumptions about the physical problem itself. In our case the objective function is the semblance (Taner and Köhler, 1969). DE was originally introduced by Storn and Price (1997) and can be classified as an evolutionary algorithm (EA). Heuristics are experience based strategies for problem solving that are able to learn and therefore adapt to a problem. They do not rely on gradients, specific type of functions or continuity. The solution is usually not guaranteed to be optimal but is mostly sufficient for a certain task. In practice the algorithm converges more often to the desired result than other optimization schemes, especially when the shape of the objective function is unknown in general.
DE is popular as a modern global optimization technique and is used in a variety of scientific fields (see, e. g. Das and Suganthan (2011)). DE is very similar to the better known genetic algorithm (GA) (Holland, 1975) since both are EAs. Like all EAs both algorithms use a number of initial solutions. In most cases they are randomly distributed and called starting population in the algorithms. It is also possible to provide previously determined solutions into the starting population, for example neighboring samples or velocity estimations. In the next step information obtained from the initial solutions is used to generate a new set of trial solutions. Each iteration, which generates new potential solutions, is called generation. The process of creating new generations is done until a satisfying solution is found. DE and GA differ in the way the next generations are generated. In the description here we focus on the DE since it was used in this work.

In DE for each candidate solution (or individual) x_i a trial vector u_i is generated which only gets into the next generation if its fitness value (i.e., semblance) is better, than the original candidate solution. The trial vector is generated by so called mutation and crossover. The index i denotes the individual in the population. The mutation of an individual x_i is done by the formula

$$y_{i,j} = a_j + F \cdot (b_j - c_j) \tag{2}$$

where a, b and c are randomly selected individuals of the population that are different from each another and x_i. The index j denotes one dimension of the problem, here the number of parameters, and $F \in [0, 2]$ is called the differential weight. The crossover parameter $CR \in [0, 1]$ is chosen by the user. A random number $r_i \in [0, 1]$ is generated and if $r_i < CR$ equation 2 is applied. The crossover parameter therefore determines the permutation probability of a dimension. To ensure that at each iteration a trial vector different from the candidate solution is tested, at least one randomly determined dimension j is forced to mutate. If the fitness value of the trial vector $f(u_i)$ is higher than the fitness of the former solution $f(x_i)$ the trial vector becomes the new candidate solution, otherwise it is discarded. This process is called reproduction. A simplified pseudo code is given by the algorithm 1.

Algorithm 1 Pseudo code for differential evolution.

```
Random initialization of population
while abort criteria not reached do
  for i = 1 to N do
    choose 3 random different individuals a, b and c
    for j = 1 to M do
      y_{i,j} = a_j + F · (b_j − c_j)
      if r_i ∈ [0, 1] < CR then
        u_{i,j} = y_{i,j}
      else
        u_{i,j} = x_{i,j}
      end if
    end for
    evaluate fitness
    if f(u_i) > f(x_i) then
      x_i = u_i
    end if
  end for
end while
```

In the pseudo code M is the number of variables, in our case three, and N is the population size.

DE has three control parameters CR, F and N which is the size of the population. They dramatically influence the computational effort and accuracy. A trend is to adapt them within the process of optimization (Price, 2005; Brest et al., 2006; Liu and Lampinen, 2005; Qin and Suganthan, 2005; Qin et al., 2009). This introduces new parameters and more complexity rather than a simple solution. After some tests we decided to stay with the simple standard DE, as described in this section, denoted as *DE/rand/1/bin* in literature. *DE* stands for differential evolution, *rand* how the individual a is chosen, *1* is the number of difference vectors considered perturbing x_i and *bin* refers to the crossover operation performed (Das and Suganthan, 2011). There are various guidelines on how to choose the parameters CR, F and N (Storn, 1996; Price, 2005; Liu and Lampinen, 2002). However, we used parameters determined by a meta-optimizer by Pedersen (2010) as they performed best. The choices were $N = 20$, $CR = 0.7455$ and $F = 0.9362$. Barros et al. (2015) also used differential evolution in the context of CRS but without treating conflicting dips.
Due to the stochastic nature of the algorithm, each run is slightly different with respect to the number of iterations required and accuracy of the solution.

IMPLEMENTATION

The principal implementation of the DE approach is rather simple as shown in algorithm 1. However, since it is a global optimization, which in principal can use any combination of parameters, premature convergence for a number of specific cases can happen, which are not necessarily physical. An example are values close to zero for either R_N or R_{NIP}. In this case the traveltime becomes close to infinity and the aperture lacks traces. If the semblance comprises just one trace, it becomes one, which is the highest possible value and the algorithm gets stuck with that solution. This is not the desired result and needs to be excluded. On the other hand for land data at low traveltimes the R_{NIP} value is naturally close to zero. To gradually increase the lower limit of R_{NIP} in the search space accordingly the user needs to specify a minimum expected moveout velocity from which a lower limit is calculated by

$$R_{NIP}^{lb} = \frac{V_{min}^2 \cdot t_0 \cdot \cos^2(\frac{1}{2}\alpha_{min} + \frac{1}{2}\alpha_{max})}{2 \cdot v_0} \tag{3}$$

where lb is the lower boundary, α_{min} and α_{max} are the boundaries of the angle range, v_0 the surface velocity and V the moveout velocity. In a similar way it is possible to provide a maximum expected velocity and calculate an upper boundary for R_{NIP} accordingly. If a velocity model is already available this can also be used to further confine the search space for R_{NIP} while a percentile variation needs to be defined. The search space for R_N is discontinuous as we restricted values close to zero based on $|R_{NIP}|$ since R_N can have positive and negative values. An alternative approach is to project R_N on the Riemann sphere as done by the implementation of Mann (2002) and limit the search space there. However, thanks to the DE algorithm a continuous search space is not required which avoids transformation artifacts in the search space shown by Witte (2014). In principal R_N can become infinite, in practice it is recommended to limit the search space since less possible solutions have to be determined and the variation of the semblance for high R_N values is rather small.
The conflicting dip handling is done by a discretization of the α space with varying discretization steps. We call this fragmentation of the α space. In our current implementation the choice of the number of individual search intervals and their range needs to be specified by the user. An angle range of $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ means only the most coherent events are searched. A fragmentation like $\left[-\frac{\pi}{2}, -\frac{\pi}{4}, -\frac{\pi}{8}, \frac{\pi}{8}, \frac{\pi}{4}, \frac{\pi}{2}\right]$ would mean 5 events are searched, one from $\left[-\frac{\pi}{2}, -\frac{\pi}{4}\right]$, another one from $\left[-\frac{\pi}{4}, -\frac{\pi}{8}\right]$ and so on. This is not the ideal case. We would rather like to search for the most coherent event, then exclude it from the search space and repeat for the desired amount of times until we covered all coherent events for that sample. Since the DE algorithm does not require a continuous search space, this is in principal possible and object of future research.
After the global optimization a local search in form of a downhill simplex (Nelder and Mead, 1965) is added if necessary for further refinement. This search is unconstrained for the purpose of events at boundaries so that the event is not stacked with inaccurate parameters from the neighboring search space. This can happen for a wide global maximum or in case a maximum lies very close to a boundary.

Each dip range creates their own attribute sections and stack. Therefore they can be used in further CRS attribute based methods allowing for conflicting dip treatment in these implementations as well. Furthermore the user can choose what to stack and thus suppress noise and artifacts.

MARINE DATA EXAMPLE

The field-data was acquired in the Levantine Basin in the Mediterranean Sea. The subsurface contains salt rollers, a slump complex and several fault systems providing a high amount of diffractions (Netzeband et al., 2006). The stack in figure 2 shows many conflicting dips due to diffractions as well as sea floor multiples starting at roughly 3s in most parts of the data. They are on top of the more interesting subsurface but can be recognized very well, especially in the velocity field in figure 3 calculated from CRS attributes. This can be used to suppress multiples (Dümmong, 2010; Vefagh et al., 2015). The velocity field can also be dip corrected and used for a data-driven time migration approach (Bobsin et al., 2015). The coherence in figure 7 is very consistent indicating a high quality of estimated attributes and shows what might be considered noise and what are actual coherent events. The section highlights a discontinuity at 2s between

CMP 1000 and 2000. The figures 4-6 show the estimated attributes for the most coherent event. They can be seen as a quality control section whether the general results seem appropriate. In the actual applications each dip section is treated individually. We can see that dips shown in the angle section (figure 4) appear consistent along diffractions and dipping layers. As expected the parameter R_{NIP} in figure 5 increases smoothly with time except when multiples are encountered. The radius of the N wave in figure 6 has high absolute values for low curvature reflections and low absolute values close to R_{NIP} for diffractions. According to Dell and Gajewski (2011) this can be exploited in the diffraction separation shown in figure 8. Applied in the prestack domain this can directly be used to determine migration velocities (Bakhtiari Rad et al., 2014). We observe some residuals at strong reflections. An investigation of the contributions of each dip shows, that these stem from high dips as the reflections are strong enough to fit a coherent diffraction tail into it. Contributions from low dips show almost no residuals.

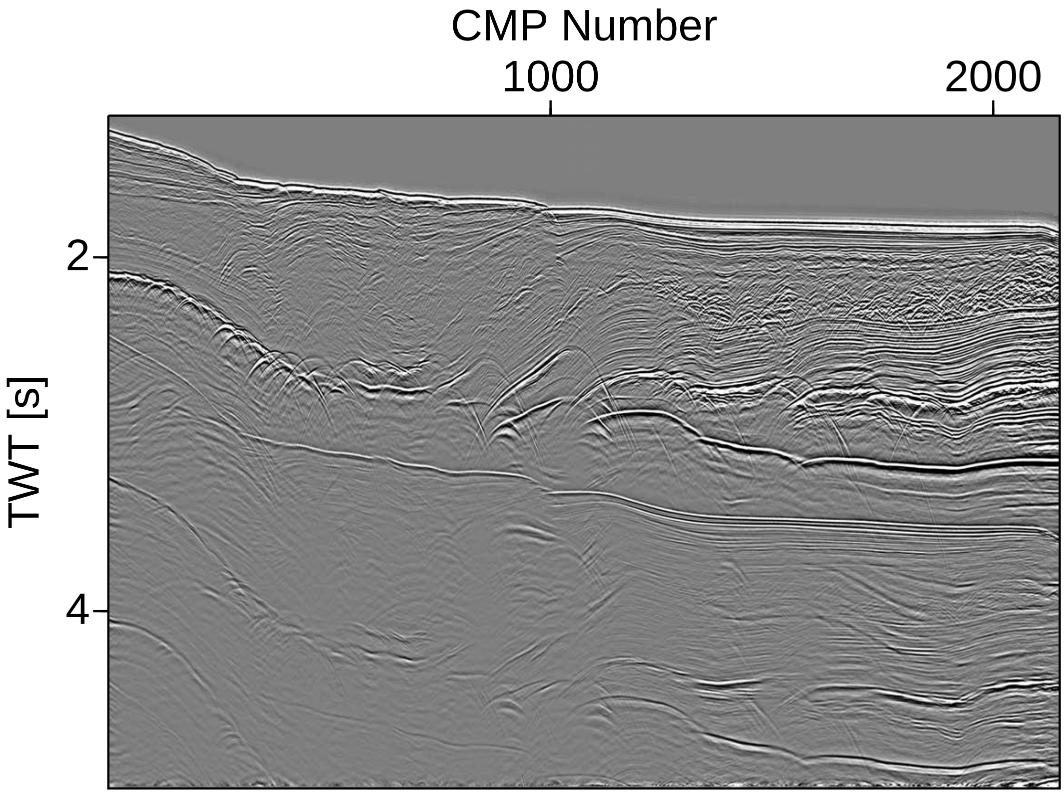

Figure 2: CRS stack obtained by the proposed method. Contributions from each dip are added.

A highly valuable application of the shown CRS attributes is the partial CRS stack, where regularized and enhanced prestack data is generated from extrapolated traveltimes based on zero offset CRS attributes. Results for a CMP 1595 of the marine data are shown in figure 9. We see a general reduction of noise and regularized offsets while all conflicting events are recovered which was not possible so far. However, the input data here was already of high quality. Therefore we applied our approach to a more challenging land data set, where many data gaps are present and the acquisition in some areas was sparse.

LAND DATA EXAMPLE

The data set was acquired in 2000 by a consortium of three universities: Hamburg, Amsterdam and Copenhagen in cooperation with the company Ukrgeofisika. Some pre-processing was done by Zhurovich (2015). Due to computational cost we only used the upper part of an area where the acquisition was dense and the geology is of interest. Figure 10 shows the input data on the left and the enhanced partial CRS results on the right. Events become more pronounced, trace density is increased and regularized. Additionally, we

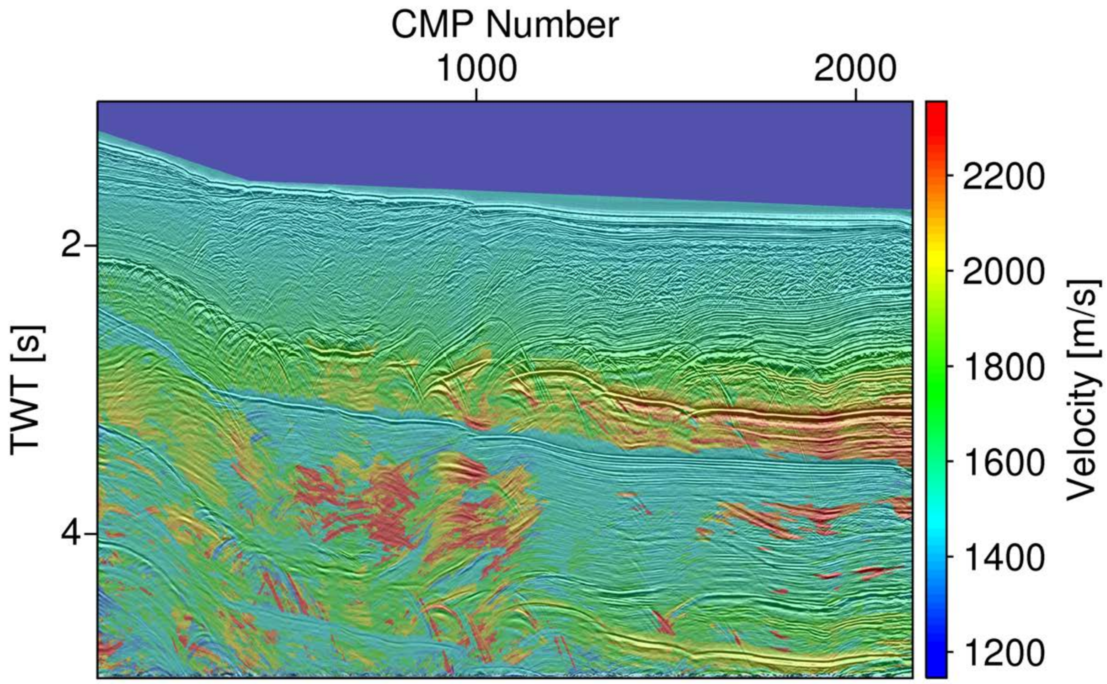

Figure 3: Overlay of the CRS stack with the from CRS parameter estimated moveout velocity V_{NMO}.

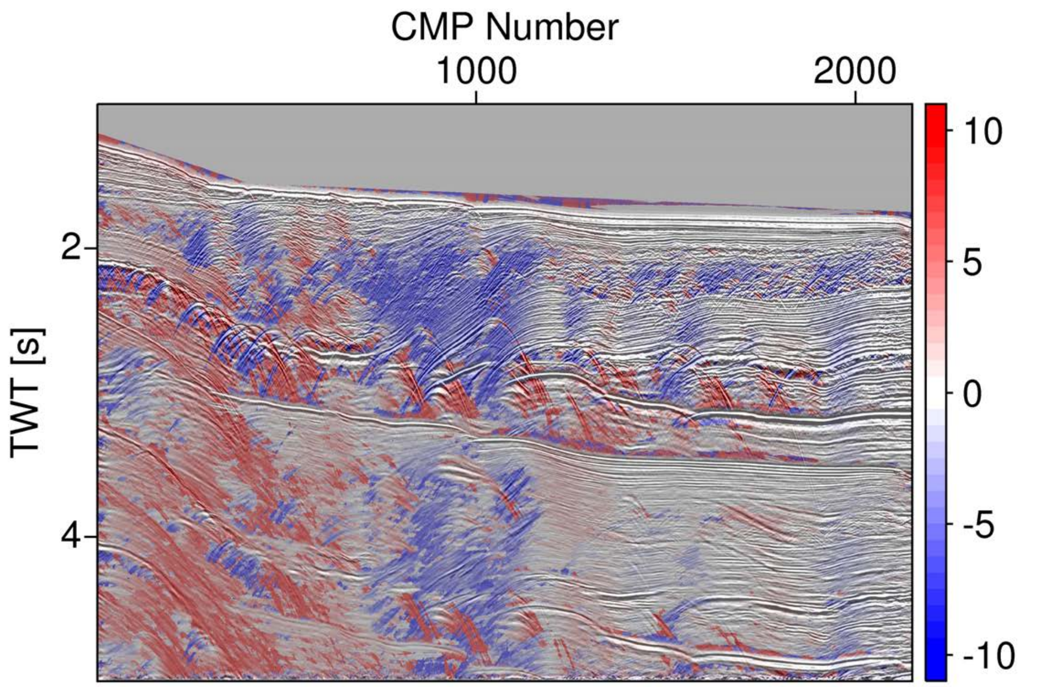

Figure 4: Overlay of the CRS stack with the CRS parameter α.

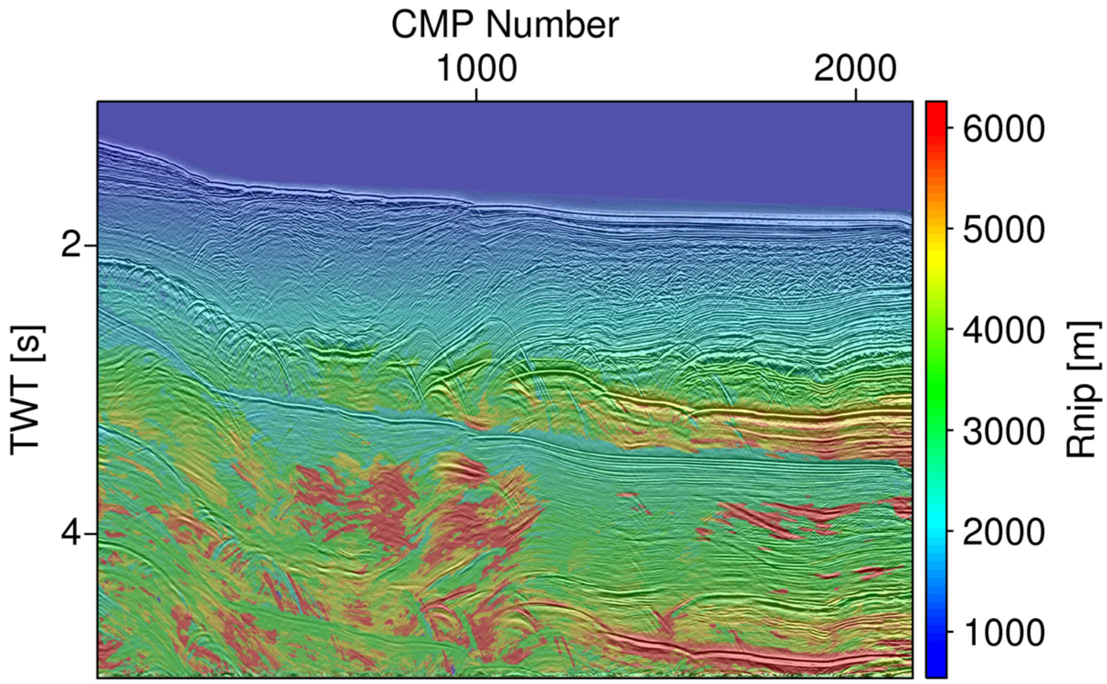

Figure 5: Overlay of the CRS stack with the CRS parameter R_{NIP}.

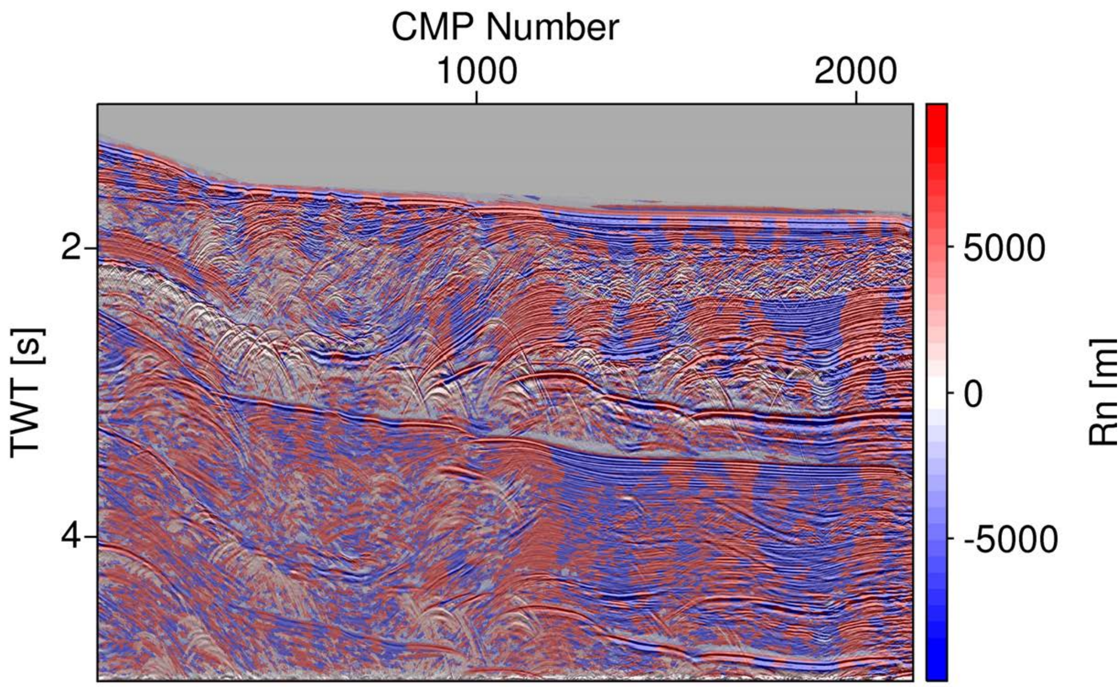

Figure 6: Overlay of the CRS stack with the CRS parameter R_N.

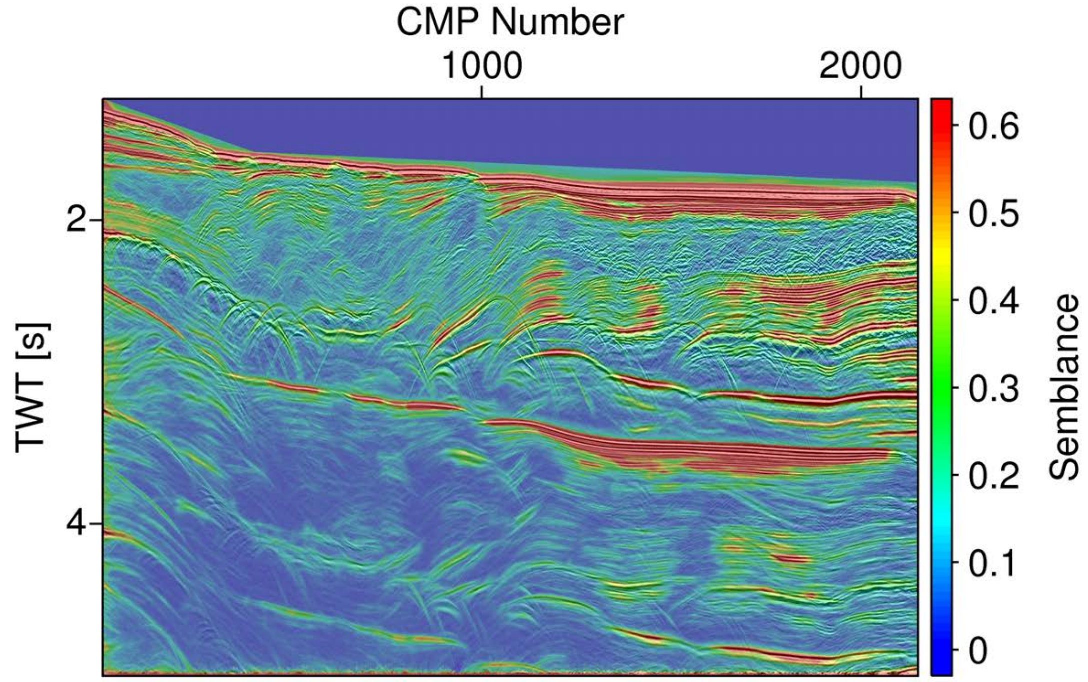

Figure 7: Section showing the highest coherence found per sample. The stack is overlain.

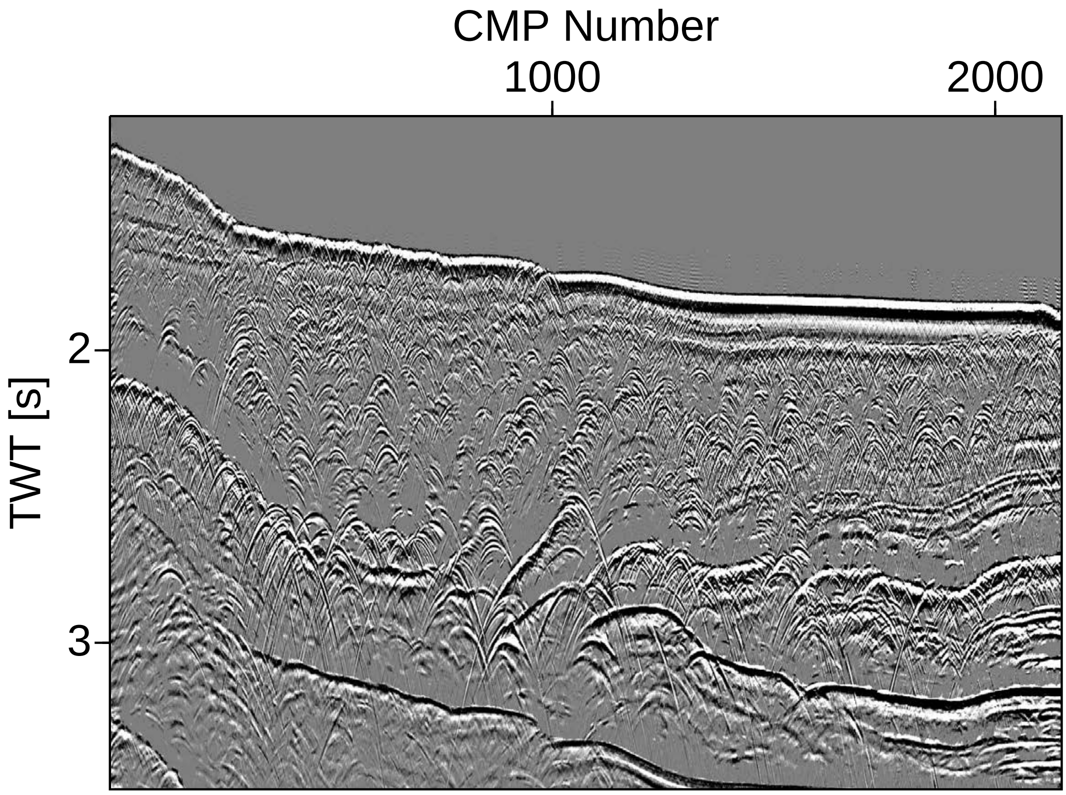

Figure 8: Diffraction separation considering conflicting dips. Most diffractions can be successfully separated. However residuals remain for events at the seafloor, the bottom of salt and the respective multiples.

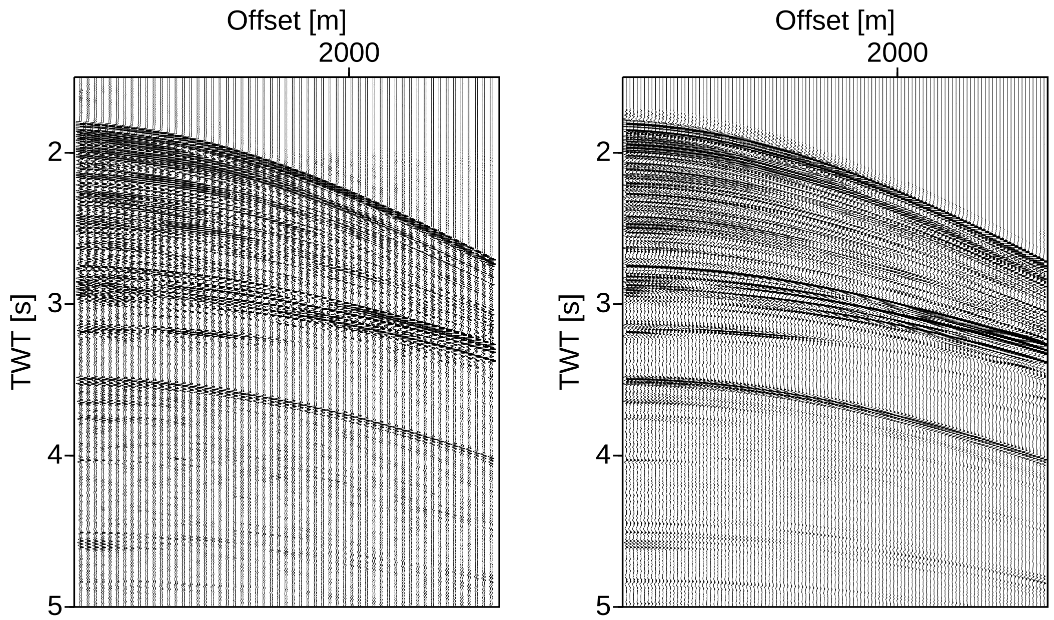

Figure 9: Input (left) and partial CRS enhanced (right) CMP gather 1595 using 19 dip ranges and DE optimization algorithm.

observe interference indicating that conflicting events are preserved and resolved. A common-offset-gather for an offset of 910 m is shown in figure 11. We observe many gaps especially in an interesting area between CMP 2500 and 3000. The enhanced data (figure 12) shows enhanced traces where gaps are filled and spacing is regularized.

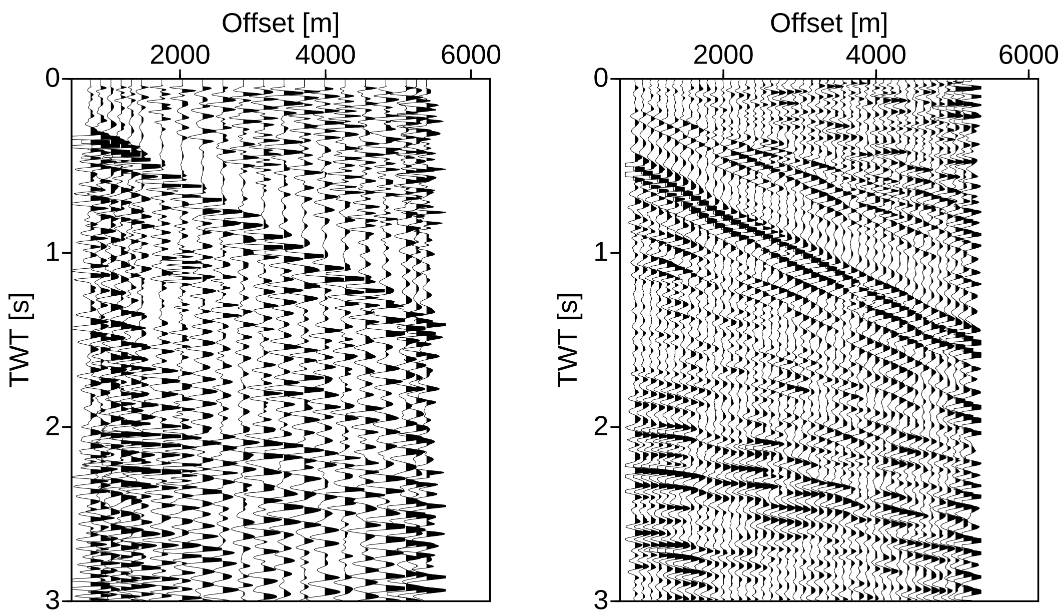

Figure 10: Input data (left) and partial CRS (right) CMP gather 3060 using 15 dip ranges and DE optimization algorithm.

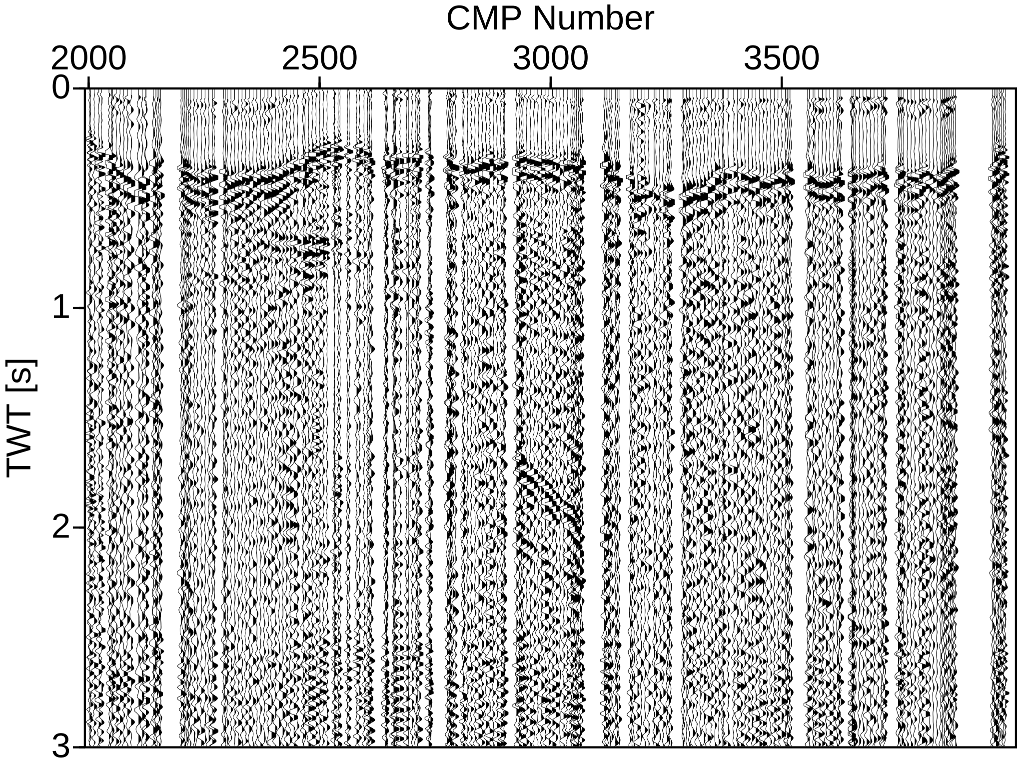

Figure 11: Input common-offset-gather.

The corresponding stacks, overlain with the coherence in figure 13 (input) and 14 (partial CRS) also show, that for the enhanced data more coherent energy is collected at locations where events are present and noise is significantly reduced. Since no a priori information was used during the processing, the results shown here can be improved, for example, by using a stacking velocity model as constraint. However, even the entirely data-driven application shows that CRS can do very well on its own, when proper optimization and conflicting dip treatment are considered.

CONCLUSIONS

We extended previous works to account for conflicting dips in the CRS method and adapted them into a global optimization scheme, which provides better attributes and also treats diffractions very well. The benefits can be used in existing CRS attribute based methods shown exemplary by diffraction separation and partial CRS. Results of the diffraction separation reveal great potential for diffraction imaging and migration velocity analysis. The prestack data enhancement resolves conflicting events in the prestack domain while regularizing the prestack data volume and increasing the signal to noise ratio significantly. This is especially important for land data.

OUTLOOK

The differential evolution algorithm is straightforward to extend to more dimensions, which is required for 3D applications since the CRS operator comprises eight attributes. However, DE is tested for a high number of dimensions in computer science. The main issue is computational costs and the relatively high amount of function evaluations required. A solution can be the combination of DE with a local optimization scheme, accelerating the convergence. Conflicting dips processing in 3D also becomes more complicated. The suggested approach would require too many dip and azimuth intervals to properly separate events. A

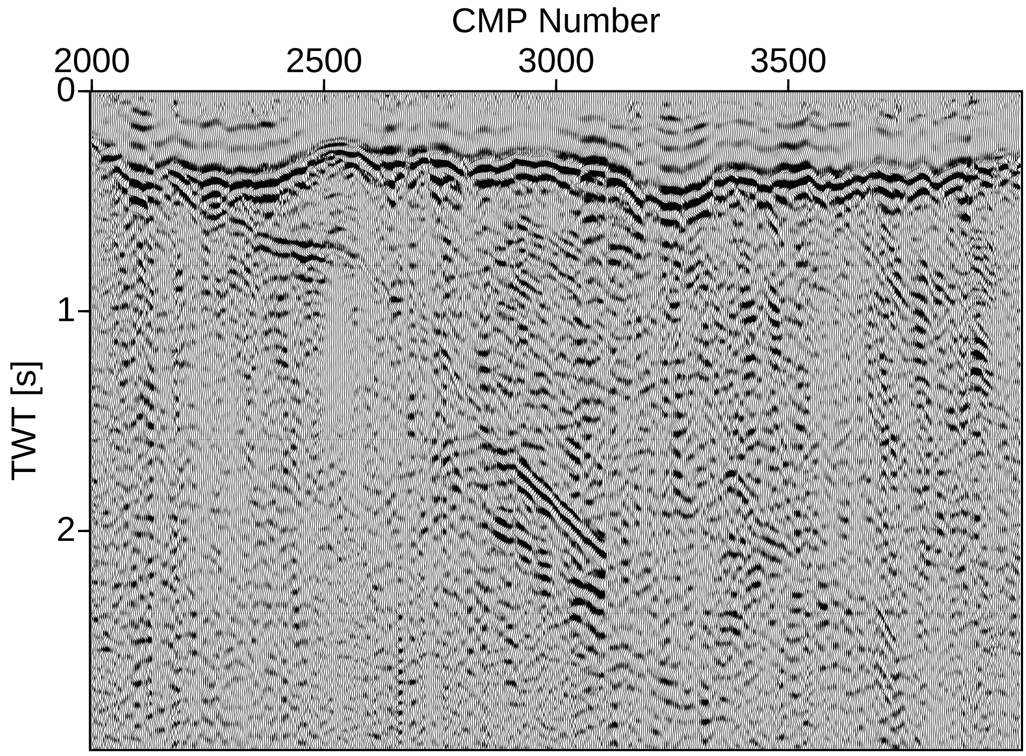

Figure 12: Partial CRS common-offset-gather using 15 dip ranges and DE optimization algorithm.

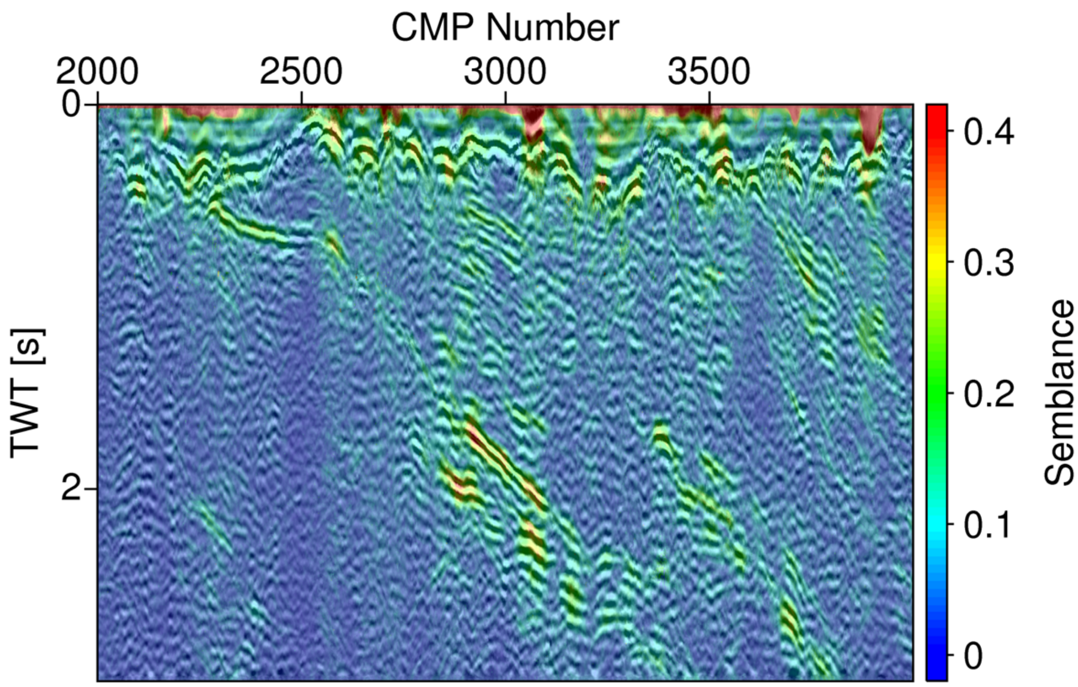

Figure 13: Coherence of the original input data overlain with the resulting stack.

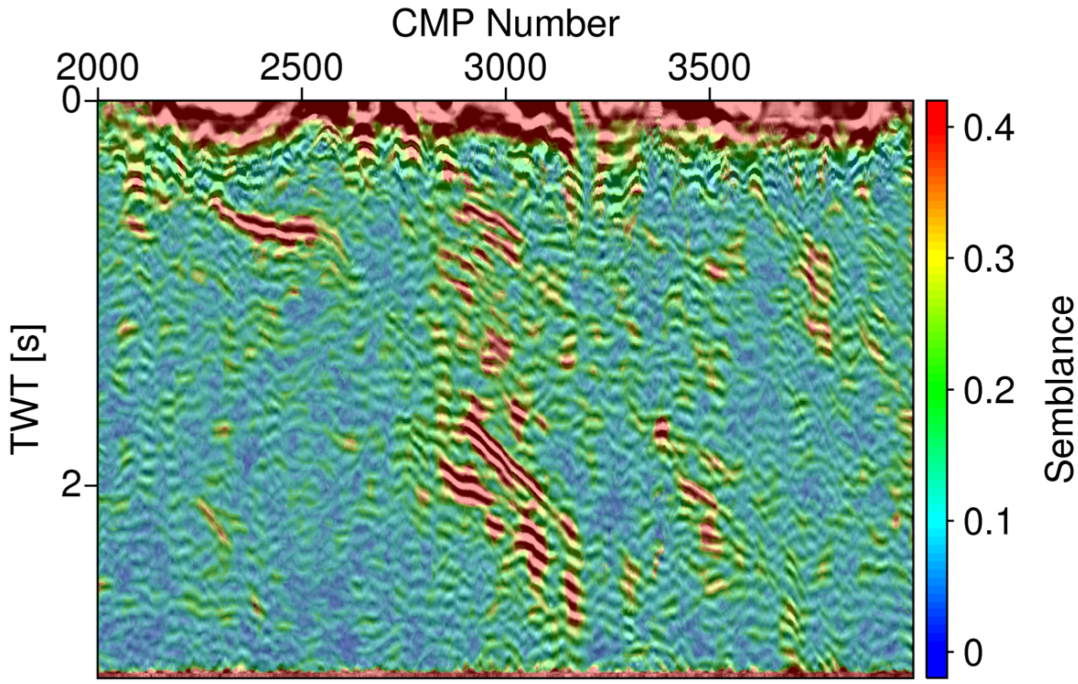

Figure 14: Coherence of the partial CRS input overlain with the resulting stack.

solution might be to identify the global maximum, measure its extent in the search space and remove it. Afterwards a new search can be applied until all coherent maxima are found. The resulting discontinuous search space does not influence the differential evolution algorithm making this approach viable.

ACKNOWLEDGMENTS

The work was supported by the WIT consortium. The marine data were kindly provided by TGS. The land data was acquired in the DOBREflection 2000 project. We would like to thank the Applied Seismics Group Hamburg for continuous discussions, TEECware for scientific exchanges and Thomas Hertweck for constructive criticism. We also thank Denys Zhurovich for providing the pre-processed land data.

REFERENCES

Bakhtiari Rad, P., Schwarz, B., Vanelle, C., and Gajewski, D. (2014). Common reflection surface (CRS) based pre-stack diffraction separation. *SEG Technical Program Expanded Abstracts*, pages 4208–4212.

Barros, T., Ferrari, R., Krummenauer, R., and Lopes, R. (2015). Differential evolution-based optimization procedure for automatic estimation of the common-reflection surface traveltime parameters. *Geophysics*, 80(6):WD189–WD200.

Bauer, A., Schwarz, B., and Gajewski, D. (2015). Prestack diffraction enhancement using a traveltime decomposition approach. *SEG Technical Program Expanded Abstracts*, pages 3820–3824.

Baykulov, M. and Gajewski, D. (2009). Prestack seismic data enhancement with partial common-reflection-surface (CRS) stack. *Geophysics*, 74(3):V49–V58.

Bobsin, M., Schwarz, B., Vanelle, C., and Gajewski, D. (2015). Data-driven time migration using a multi-parameter approach. *SEG Technical Program Expanded Abstracts*, pages 4350–4354.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., and Zumer, V. (2006). Self-adaptive control parameters in differential evolution: a comparative study on numerical benchmark functions. *IEEE Transactions on Evolutionary Computation*, 10(6):646–657.

Das, S. and Suganthan, P. N. (2011). Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 15(1):4–31.

Dell, S. and Gajewski, D. (2011). Common-reflection-surface-based workflow for diffraction imaging. *Geophysics*, 76(5):S187.

Dümmong, S. (2010). *Seismic data processing with an expanded common reflection surface workflow*. PhD thesis, University of Hamburg.

Höcht, G., Ricarte, P., Bergler, S., and Landa, E. (2009). Operator-oriented CRS interpolation. *Geophysical Prospecting*, 57(6):957–979.

Holland, J. H. (1975). *Adaption in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor.

Hubral, P. (1983). Computing true amplitude reflections in a laterally inhomogeneous earth. *Geophysics*, 48:1051–1062.

Jäger, R., Mann, J., Höcht, G., and Hubral, P. (2001). Common-reflection-surface stack: Image and attributes. *Geophysics*, 66:97–109.

Liu, J. and Lampinen, J. (2002). On setting the control parameter of the differential evolution method. In *Proceedings of the 8th International Conference on Soft Computing (MENDEL)*, pages 11–18, Brno, Czech Republic.

Liu, J. and Lampinen, J. (2005). A fuzzy adaptive differential evolution algorithm. *Soft Computing*, 9(6):448–462.

Mann, J. (2001). Common-reflection-surface stack and conflicting dips. *71st Annual Internat. Mtg., Soc. Expl. Geophys*, Extended Abstracts:Session SP 5.3.

Mann, J. (2002). *Extensions and Applications of the Common-Reflection-Surface Stack Method*. PhD thesis, Logos Verlag, Berlin.

Müller, N. A. (2009). Treatment of conflicting dips in the 3D common-reflection-surface stack. *Geophysical Prospecting*, 57(6):981–995.

Müller, T. (1999). *The Common Reflection Surface Stack Method: Seismic Imaging without explicit knowledge of the velocity model*. PhD thesis, University of Karlsruhe.

Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–313.

Netzeband, G. L., Gohl, K., Hübscher, C. P., Ben-Avraham, Z., Dehgani, G. A., Gajewski, D., and Liersch, P. (2006). The levantine basin-crustal structure and origin. *Tectonophysics*, 418:167–188.

Pedersen, M. E. H. (2010). Good parameters for differential evolution. Technical Report HL1002, Hvass Laboratories.

Price, R. (2005). *Differential Evolution: A Practical Approach to Global Optimization*. Springer.

Qin, A. K., Huang, V. L., and Suganthan, P. N. (2009). Differential evolution algorithm with strategy adaption for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, 13:398–417.

Qin, A. K. and Suganthan, P. N. (2005). Self-adaptive differential evolution algorithm for numerical optimization. In *Proceedings of the IEEE congress on evolutionary computation (CEC)*, pages 1785–1791.

Schwarz, B., Vanelle, C., Wißmath, S., Bauer, A., and Gajewski, D. (2015). Efficient common-reflection-surface-based prestack slope determination for stereotomography. *SEG Technical Program Expanded Abstracts*, pages 5248–5253.

Soleimani, M., Piruz, I., Mann, J., and Hubral, P. (2009). Solving the problem of conflicting dips in common reflection surface (CRS) stack. *1st International Petroleum Conference & Exhibition*, EAGE, expanded abstracts, A39.

Storn, R. (1996). On the usage of differential evolution for function optimization. In *Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS)*, pages 519–523.

Storn, R. and Price, K. (1997). Differential evolution - a simple and efficient heuristic for global optimization over continues spaces. *Jornal of Global Optimization*, 11:341–359.

Taner, M. T. and Köhler, F. (1969). Velocity spectra-digital computer derivation applications of velocity functions. *Geophysics*, 34(6):859–881.

Vefagh, M., Dümmong, S., and Gajewski, D. (2015). A zero-offset picking approach for pre-stack multiple attenuation. *SEG Technical Program Expanded Abstracts*, pages 4475–4479.

Walda, J. and Gajewski, D. (2014a). *Global optimization of the CRS operator using a genetic algorithm*, pages 281–291. WIT Report. WIT Consortium.

Walda, J. and Gajewski, D. (2014b). *Handling the conflicting dip problem in the CRS/i-CRS method*, pages 164–172. WIT Report. WIT Consortium.

Walda, J. and Gajewski, D. (2015). Common-reflection-surface stack improvement by differential evolution and conflicting dip processing. *SEG Technical Program Expanded Abstracts*, pages 3842–3847.

Witte, P. (2014). Optimization of the semblance in seismic processing. Master's thesis, University of Hamburg.

Zhurovich, D. (2015). Bearbeitung reflexionsseismischer Daten aus dem Donez-Becken. Master's thesis, University of Hamburg.