SOUND SPEED CORRECTION USING AUTOFOCUS ON SAS IMAGES

Ulrich J. Herter[a], Holger Schmaljohann[b], and Thomas Fickenscher[a]

[a]Helmut-Schmidt-University, Holstenhofweg 85, 22043 Hamburg, Germany
[b]Bundeswehr Technical Center for Ships and Naval Weapons, Maritime Technology and Research, Berliner Str. 115, 24340 Eckernförde, Germany

Ulrich J. Herter, Helmut-Schmidt-University, Holstenhofweg 85, 22008 Hamburg, Germany, ulrich.herter[at]hsu-hh[dot]de, fax: +49 40 6541-2061

Abstract: *The quality of automatic target recognition (ATR) results depends heavily on the quality of the images it works on. One source of quality loss is using an inaccurate sound speed during synthesizing the aperture. SAS images usually comprise a long abeam range whereas the sound speed is measured on the sonar platform directly and/or using CTD profiles taken at few, selected locations. This makes it difficult to get valid sound speed values for the entire area imaged. Errors in sound speed do not only cause wrong range estimates but also blurring in along track direction due to an erroneous range migration correction. Deviations as little as* $1\,\%$ *from the true sound speed may lead to significant image degradation. The blurring forms a characteristic hyperbolic shape which can be exploited to derive an improved average sound speed value.*

We present a method for sound speed correction employing a modified stripmap phase gradient autofocus (SPGA) and a previously published method employing phase gradient autofocus (PGA) and compare the two variants: We used SPGA or PGA to estimate phase error histories from the image data itself and fited error models to the estimates. The fit parameters allowed us to calculate an improved sound speed. After applying an inverse wavenumber algorithm with the old, erroneous sound speed, the new, corrected sound speed value was applied to the artificial raw data by forward wavenumber processing.

A comparison of SPGA and PGA based sound speed correction is shown on simulated as well as on field data comprising different features (a wreck, ripples and stones): The original images were processed using deliberately biased sound speeds as well as the best available values from the missions. In most cases, our correction method was able to significantly improve the images. Even in cases where the best available sound speed was used in the first place, we achieved improvements.

Keywords: *Synthetic aperture, Sonar, Autofocus, Speed of sound*

1 INTRODUCTION

The length of apertures synthesised in Synthetic Aperture Sonar (SAS) imaging requires a precise knowledge not only of the carrying device's pathway but also of the medium's speed of sound values. Different autofocus methods have been developed to estimate and correct SAS imagery for path deviations directly from a processed image. Phase Gradient Autofocus (PGA) was first published by [1], and extended for application in Strip-map images (Strip-map Phase Gradient Autofocus (SPGA)) by [2]. Both algorithms employ common phase estimator kernels, such as shear averaging, but SPGA works on preselected targets which are considered point like scatterers. The second major difference is: SPGA performs a wavenumber transform prior to the estimation and results in a phase error estimate in terms of along track position whereas PGA does the estimation and correction in the spatial Fourier domain directly.

In this paper, we use both algorithms only to estimate phase error histories from which we calculate average errors for the speed of sound values used in processing. In the case of PGA, we minimize the corresponding error function in the Fourier domain following [3]. For SPGA we use a quadratic error model, which we fit to the phase estimate as explained in more detail in Sec. 2. The image correction step is the same in both cases, and uses an inverse wavenumber processor followed by forward processing using a conventional wavenumber processor with updated speed of sound values. For the implementation of the wavenumber algorithms we follow [4].

The process described above was applied to a number of test cases, processed with biased speed of sound values, from simulated and field data of different sea bottom types.

2 THEORY

Let c be the biased speed of sound, x the along track coordinate, x_0 the point of closest approach (PCA) to a given target, Δt_{twt} the two way travel time error and $r_0 = ct_0$ the shortest distance between the sonar and the target. Note that we use here the one way travel time $t_0 = 0.5t_{\mathrm{twt},0}$. Further we consider a monostatic sonar and use the stop-and-hop approximation, assuming the sonar doesn't move during the signals travel time.

2.1 Speed of sound from SPGA

If the phase error estimate

$$\Delta\phi(x) = 2\pi f_{\mathrm{c}} \Delta t_{\mathrm{twt}}(x) \tag{1}$$

at the carrier frequency f_{c} and t_0 are known, we can relate this function to an error in speed of sound Δc as follows:

Without loosing generality we can set $x_0 = 0$, then the two way travel time in terms x is:

$$t_{\mathrm{twt}}(x) \;=\; \frac{2}{c}\sqrt{x^2 + c^2 t_0^2}. \tag{2}$$

For small errors in speed of sound Δc, the error of two way travel time can be approximated:

$$\Delta t_{\mathrm{twt}} = \frac{dt_{\mathrm{twt}}}{dc}\Delta c + \mathcal{O}(\Delta c^2) = -\frac{2x^2\Delta c}{c^2\sqrt{x^2+c^2t_0^2}} + \mathcal{O}(\Delta c^2). \tag{3}$$

Expanding Δt_{twt} around $x = 0$ we get:

$$\Delta t_{\mathrm{twt}} = \frac{-2\Delta c}{c^3 t_0}x^2 + \mathcal{O}(x^3, \Delta c^2) \tag{4}$$

$$\Rightarrow \Delta\phi = \frac{-4\pi f_{\mathrm{c}}\Delta c}{c^3 t_0}x^2 + \mathcal{O}(x^3, \Delta c^2) \approx Ax^2 + Bx + C. \tag{5}$$

and

$$A = \frac{-4\pi f_{\mathrm{c}}\Delta c}{c^3 t_0}; \quad B = C = 0. \tag{6}$$

The discrete version of $\Delta\phi[x_n]$ can be estimates using SPGA and a fit to this allows for estimating A, B and C (n denotes row number in the SAS image). From A we get a value for the sound speed error:

$$\Delta c \approx -\frac{Ac^3 t_0}{4\pi f_{\mathrm{c}}}. \tag{7}$$

2.2 Speed of sound from PGA

Following [3], we use the output of the PGA phase error estimation $\Delta\phi[k_{x,n}]$ and minimize the error function $\epsilon(\Delta c)$ to get a value for Δc. $\epsilon(\Delta c)$ writes as follows:

$$\epsilon(\Delta c) = \sum_n \left\{ \mathrm{detr}\left[\Delta\phi[k_{x,n}] - r_0\left(\sqrt{4k_{\mathrm{old}}^2 - k_{x,n}^2} - \sqrt{4k_{\mathrm{new}}^2 - k_{x,n}^2}\right)\right]\right\}^2 \tag{8}$$

$$\text{with } k_{\mathrm{old}} = \frac{2\pi f_c}{c} \text{ and } k_{\mathrm{new}} = \frac{2\pi f_c}{c - \Delta c}.$$

n is the number of the discrete wavenumber bin and $k_{x,n}$ the corresponding wavenumber.

3 IMPLEMENTATION

The sound speed correction is implemented as shown in Fig. 1. Main inputs are the blurred image and the speed of sound c, which was used to process the image. It is searched for strong targets. For each target i, phase errors are estimated either using SPGA or PGA and sound speed errors Δc_i are calculated. In case of SPGA only small windows (2m by 1m) around each target are fed into the estimator, whereas in case of PGA snippets (5m by 5m) with the target at the centre are used.

In theory, a single point scatterer in an image should be sufficient to estimate Δc. In practice, it is important to use as many estimates as possible, remove outliers and the ones with bad fit statistics and use the median of the remaining values for each iteration m:

$$\Delta c_m = \mathrm{med\,filt}\{\Delta c_i\}. \tag{9}$$

If $\mathrm{abs}\,\Delta c_m/c_{m-1}$ is larger than a given threshold, $c_m = c_{m-1} - \Delta c_m$ is updated and the defocused data is run through a forward $\omega - k$ processor using c_m. The process can also be interrupted by specifying a maximum number of allowed iterations. For updating the image we use the same method as in [5]. First the initial image is defocused by an inverse wavenumber ($\omega - k$) algorithm and, in each iteration, focused using the updated speed of sound by a normal wavenumber processor following [4].

3.1 SPGA estimator

Our implementation of SPGA follows closely the approach described by [2] with a number of additional parameters (see [5]). Instead of running the full process, we break at step 17 in [2] and feed the phase error estimate into the process described in Sec. 2.1. For the fit we use Matlab's `polyfit` function which solves the Vandermonde matrix. Only one iteration of SPGA sound speed error estimation is performed for each target. It turned out that for estimating speed of sound it is better not to perform de-trending of the sheared product as we did for estimating sway errors in [5], but include the parameters B and C in the fitting process to account for offsets and linear trends. It appears that de-trending the sheared products changes the results of the amplitude weighting and unwrapping and therefore the measured curvature (parameter A).

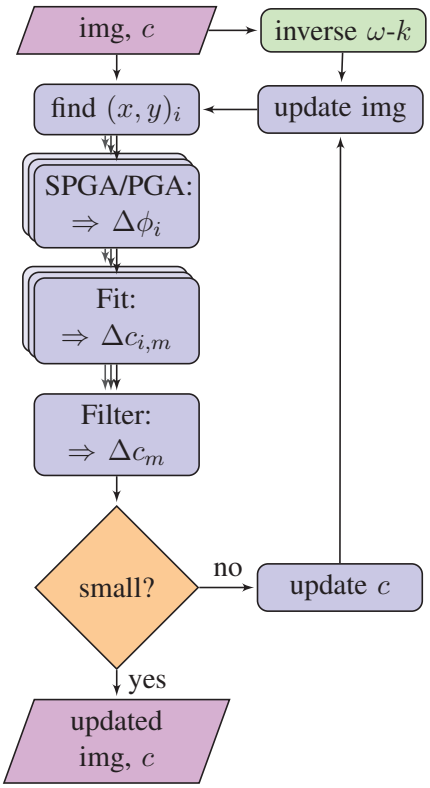

Fig. 1: Flow chart of the image correction process.

3.2 PGA estimator

[3] use FLOS-PGA with a fixed number of 3 iterations. In contrast, we implement PGA following [1] with a dynamic break condition: As soon as the automatic windowing (step B in [1]) increases the window size, the iteration is stopped and the cumulative phase error $\Delta\phi[k_{x,n}]$ in terms of the along track wavenumber bins is returned. From the phase error we get Δc by minimizing the error function (8) and use the centre range of the given snippet as r_0. For the optimization we use Matlab's function `fminbnd` which uses golden section search and parabolic interpolation algorithms on a given search interval.

4 RESULTS

We tested the proposed method on simulated point scatterers, as well as on field data comprising different sea floor types, such as stone fields, ripple fields and a wreck. All examples were processed using a conventional delay-and-sum beam forming, with the best known

Test group	abs. Residual mean %	Contrast gain mean %	Iterations mean	Run time mean [s]	Tests number of
SPGA simul	0.03	106.30	0.80	23.7	5
SPGA field	0.37	9.03	2.82	588.8	28
PGA simul	0.03	106.07	0.80	18.1	5
PGA field	0.53	7.25	2.71	181.0	28

Tab. 1: Metrics of SPGA and PGA based sound speed corrections.

speed of sound, and added errors of $-60\,\mathrm{m\,s^{-1}}$ to $60\,\mathrm{m\,s^{-1}}$. These test cases were run through the sound speed autofocus, with both options for phase estimation, a maximum number of iterations of 5, a break condition of $\mathrm{abs}\,\Delta c/c > 0.002$, and a maximum number of 100 targets per image. The 66 test cases can be divided in four major groups by the estimation method (SPGA or PGA) and the data origin (simulation or field experiment). For efficient computation of the test cases, we used the tool GNU parallel by [6].

As metric for evaluation of the results, we computed the residual absolute error in speed of sound and the contrast value, defined as the quotient of standard deviation and mean of absolute pixel values.

4.1 Focusing simulated and field data

Tab. 1 shows the mean values of the quality metrics along with information about the test case groups. The mean number of iterations below one in the simulation groups was a consequence of the break condition, which stopped the algorithm before completing the first loop in case it measured a small Δc. In all four test groups the residual error was below $0.6\,\%$, and the gain in contrast was positive. For the simulations there were almost no differences in performance of SPGA and PGA based correction. Only the run times were smaller in the cases using PGA. This does not apply for the cases using field data: SPGA yielded better results: The residuals were smaller, the contrast gains higher, but the run times were worse. The SPGA method is more costly, computationally.

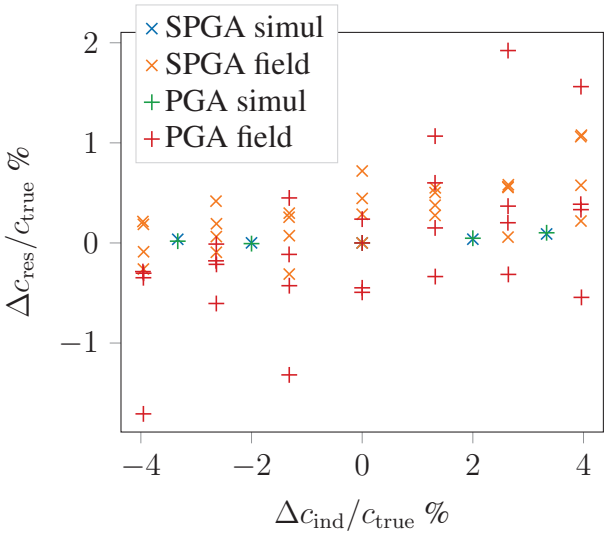

Fig. 2: Relative residuals vs. induced sound speed errors. The four test groups are color coded.

Comparing the two estimation methods, the actual values for Δc were not always consistent. Fig. 2 shows all relative residual errors vs. the induced errors. In all cases the residuals were smaller than the induced errors, in most cases well below $1\,\%$. However there were more outliers when using PGA, indicating a better consistency of the SPGA method.

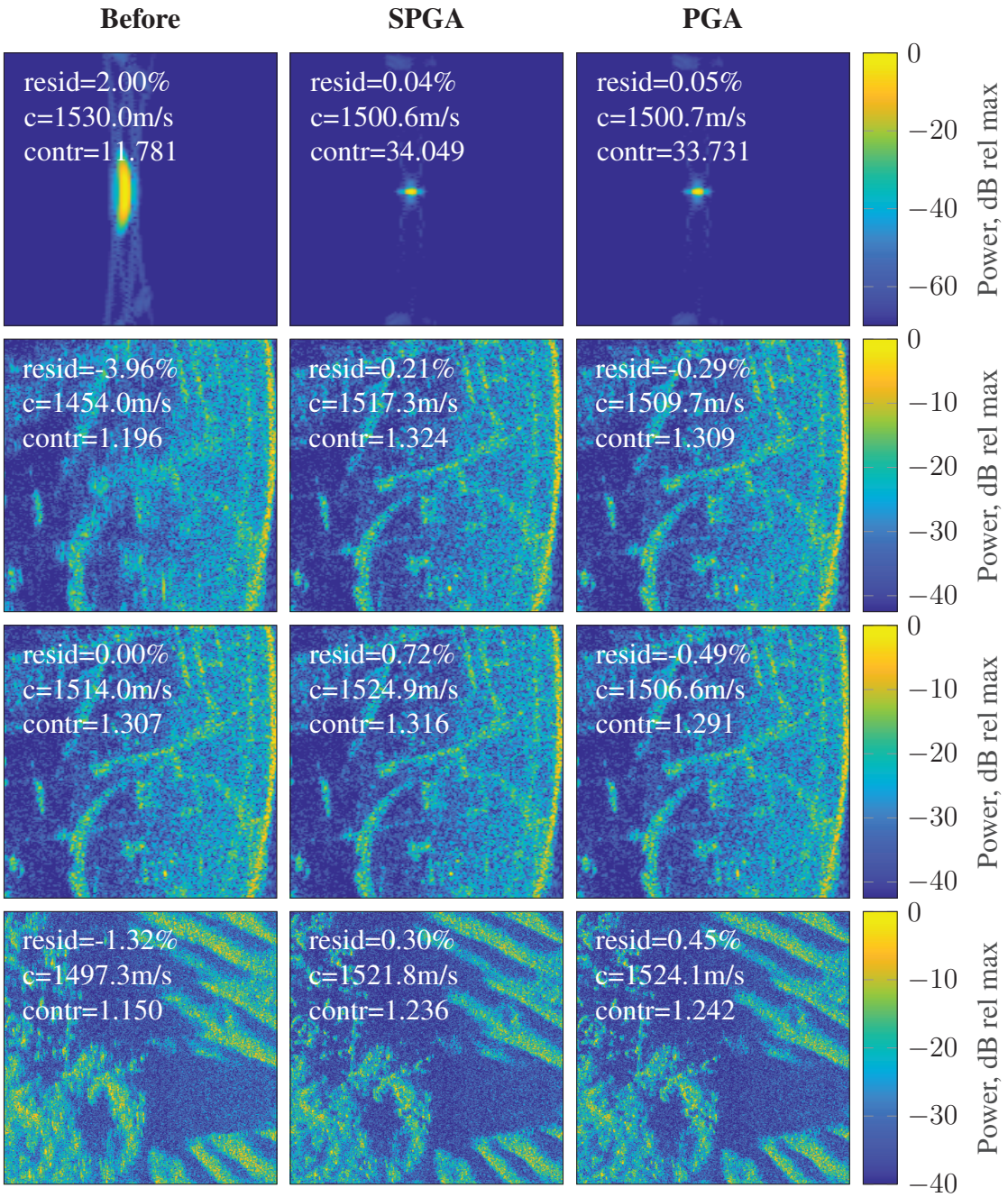

Fig. 3: Snippets of images before and after SPGA/PGA based sound speed autofocus. 1st row: Simulated point scatterer, induced error of $2\,\%$*; 2nd row: Wreck image, induced error* $-3.96\,\%$*; 3rd row: Wreck image, no error induced; 4th row: Stones and ripples, induced error* $-1.32\,\%$*. The contrast values are calculated for the shown snippets.*

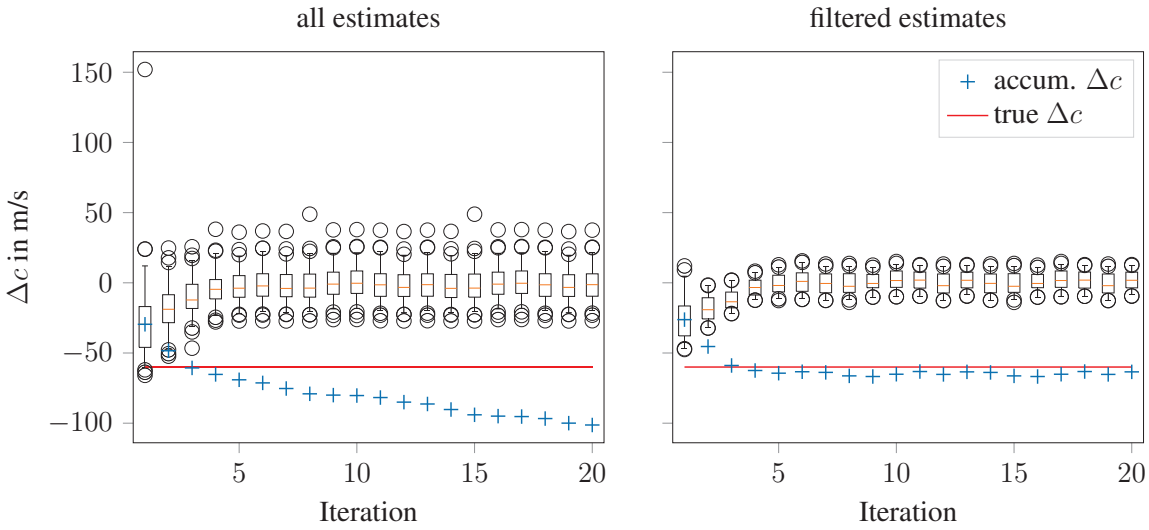

Fig. 4: Box plots of Δc_i estimates during 20 iterations of SPGA based autofocus with and without filtering. The orange dashes are the median values, the boxes contain 50% of the estimates and the whiskers mark the 95% interval. Circles show outliers (outside 95%). The blue + are the cumulated sound speed estimate and the red bare shows the induced error.

Fig. 3 shows the results of four test cases. On noise free (simulated) data, SPGA and PGA performed equally well. In the presence of noise and extended reflecting objects, SPGA seems to be more accurate and yields better contrast values. Interestingly, the SPGA based method was able to improve the wreck image without any induced error (3rd row, centre snippet). Comparing the contrast values of the SPGA focused images of the wreck (2nd and 3rd row, centre) of 1.324 and 1.316 with the contrast of the same image without induced error (3rd row, left) of 1.307, there is in both cases an improvement. This shows that our method is able to improve images, processed with the best known speed of sound values, and also to recover dramatically corrupted images.

A reason for the improved sharpness of the centre snippet in the 3rd row (Fig. 3) probably was the difference in depth between the deck of the wreck and the seafloor. The images were processed using a flat seafloor, so that some of the errors induced by this assumption were compensated by adjusting the speed of sound value.

4.2 Convergence test on field data

Fig. 4 shows the results of a convergence test using SPGA with and without filtering of estimates over 20 iterations. The test was performed on field data with an induced error of $-60\,\mathrm{m\,s^{-1}}$. It is clearly visible that a statistical filtering of estimates was necessary to achieve convergence, otherwise outliers would have biased the final results. Interestingly, the distribution width of estimates decreased during the first four iterations only, and remained at a certain level on-wards. This shows that measuring sound speed on individual targets is not sufficient and it is important to include as many targets as possible. Also it indicates that our filtering method is effective.

5 CONCLUSION

We proposed a new method for estimating sound speed errors from processed SAS images by making use of the existing autofocus algorithm SPGA. We integrated this method, along with the perviously published method using PGA, into an iterative algorithm for applying improved sound speed values to SAS imagery. Our algorithms did not rely on reprocessing raw data but were applied to processed SAS images directly.

The method proofed it's potential on simulated data as well as on field data. In most cases it was able to improve the contrast of the images significantly and to recover the true speed of sound with good precision. In some cases it even improved field images that had been processed using the best known speed of sound in the first place. This could be valuable for further processing and evaluation of SAS images, such as automatic target recognition and reliable localization of objects of interest.

Further work should seek to improve the computational costs of the process, especially for using SPGA. In depth statistical analysis of sound speed estimates could improve efficiency and reliability.

6 ACKNOWLEDGEMENTS

WTD 71 thanks CMRE for the opportunity to operate the AUV SeaOtter during the seatrial ITMINEX'14 from R/V Alliance.

REFERENCES

[1] D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and C. V. Jakowatz. Phase gradient autofocus-a robust tool for high resolution sar phase correction. *IEEE Transactions on Aerospace and Electronic Systems*, 30(3):827–835, Jul 1994.

[2] H. J. Callow, M. P. Hayes, and P. T. Gough. Stripmap phase gradient autofocus. In *Oceans 2003. Celebrating the Past ... Teaming Toward the Future (IEEE Cat. No.03CH37492)*, volume 5, pages 2414–2421 Vol.5, Sept 2003.

[3] Roy Edgar Hansen, Hayden John Callow, and Torstein Olsmo Sæbø. The effect of sound velocity variations on synthetic aperture sonar. In *Proceedings of Underwater Acoustic Measurements 2007*, pages 323–330, 2007.

[4] Mehrdad Soumekh. *Synthetic aperture radar signal processing with MATLAB algorithms*. Wiley, New York [u.a.], 1999.

[5] U. Herter, H. Schmaljohann, and T. Fickenscher. Autofocus performance on multi channel sas images in the presence of overlapping phase centers. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–6, Sept 2016.

[6] O. Tange. Gnu parallel - the command-line power tool. *;login: The USENIX Magazine*, 36(1):42–47, Feb 2011.