Multibeam Swath Consistency Detection and Downhill Filtering from Alaska to Hawai‘i

Brian Calder*
Center for Coastal and Ocean Mapping & NOAA-UNH Joint Hydrographic Center, University of New Hampshire, Durham, NH 03824, USA

Abstract

A common problem with some MBES systems is inconsistent bottom detection where there is a significant slope to the bathymetry. The outer beams of the swath either never interact with the seafloor or intersect with such a low grazing angle that the bottom return is very diffuse. This generally renders the swath unusable past some maximum angle on one side. A similar effect can be engendered by intentionally increasing the pulse repetition rate of the sonar so that the next transmit cycle occurs before the outer beams of the last are captured. Frequently, these problems result in pseudo-random picks in the water-column, either detecting water column noise or side-lobe returns from the strongest normal incidence interaction with the seafloor.

The spatial cohesion of this noise source, its relationship to the strongly sloping bathymetry and the density of data with respect to the next deeper swath in sequence makes this particularly problem-atic for human operators and automatic algorithms alike. This paper shows that some simple measures of consistency can be used to pick a ‘breakpoint’ after which the swath is considered to be essentially noise; conventional flagging of soundings follows, leaving a pre-processed dataset for further analysis. Multiple methods of detection are considered, and it is shown that these can be suitably combined to provide either a more robust method or a layered solution where increasingly complex methods are introduced sequen-tially to trade off complexity and run-time.

1 Introduction

Bottom detection performance in MBES systems can be affected by a number of different factors, both systematic and environmental. One particular problem is when significant slope occurs across the swath of the sonar. In all systems, very low relative grazing angle leads to little return and hence lower SNR, making detection difficult. In hydrographic surveys, it is typically the case that the highest possible ping rate is required so that the system can travel as fast as possible but still achieve the mandated along-track coverage requirements (typically that the along-track foot-prints overlap to some extent [5]). Since the ping rate is a function of the maximum range observed, where slopes occur this would normally be controlled by the deepest side of the swath, leading to lower ping rates than might be preferred for the (hydrographically) more significant shoal areas. On some systems, the maximum range may be set directly so that the ping rate is kept artificially high. An immediate consequence of this, however, is that the outer beams on the downhill side of the swath only see watercolumn data (Figure 1) and hence cannot ever detect the bottom. The watercolumn data is affected by returns from previous pings in the water and sidelobe interference from the rest of the bathymetry (typically the most normal incidence). Consequently, data from this outer swath region can be very noisy, leading to significant time and expense in remediation. Even with modern al-gorithmic tools for hydrography this problem causes difficulties because most algorithmic tools are predi-cated on consistency of soundings. Due to the nature of the ‘noise’, the soundings look both consistent and shoaler than the alternatives, and occur most com-monly in regions where the alternative data is sparser due to the rapidly increasing depths. The result is clouds of dubious data points that need to be removed or reduced before further processing, Figure 2. A very similar problem is observed in very deep data where the maximum (physical) range of the sonar is being tested. In this case, the outer beams never receive enough energy from the seabed to make a detection decision purely due to absorption and spreading of the acoustic energy.

The observed problem has strong geometric structure and is engendered in line survey mode. Therefore it is probably best approached per line and with ge-ometric primitives. The approach taken here is to assume that the data is being observed ‘raw’ — that
Figure 1: Schematic for the downhill problem. The difficulties are due to very oblique incidence on heavy slopes, and user-specified maximum ranges. The maximum range limit is set to increase ping rate and hence along-track density in the shoaler regions of the swath.

Figure 2: Example data showing the downhill noise problem, primarily due to significant slope in the region. Data vertical exaggeration is 1x here. Note the spatial coherency of returns indicated, which are due to side-lobes of the main beam receiving energy from the strong normal incidence return further up the slope. Data from Valdez, AK, courtesy of the NOAA Ship RAINIER.

is, after corrections for motion, etc., but before any automatic or manual processing. Realizing that no one algorithm will be capable of detecting all problems at all times and under all circumstances, the system has been built to allow for a variable number of algorithms to be utilized either concurrently or in sequence. Each algorithm attempts to predict where in the swath the breakpoints to bad data occur, and the meta-algorithm then attempts to fuse these predictions in order to better determine the true breakpoints. ‘Better’ here can be taken to mean either faster, or more reliably. The breakpoints determined by the fused algorithms are then used to conventionally ‘flag’ the data as not for use in order to make a product compatible with general hydrographic processing systems.

2 Theory

2.1 Safe Aperture Tracker

A number of the algorithms used need to estimate some basic properties of the data (e.g., slope, noise variance). Given that the data has not been ‘cleaned’ or otherwise processed, it is important that, in addition to suitably robust techniques, the algorithms are given an idea of where in the swath is likely to be unaffected by the noise problems. This is called a ‘safe aperture’, and is represented as a start and stop beam number, \( \hat{b}(p) \) and \( b(p) \) respectively, for each ping. In order to ensure that the safe aperture remains safe, feedback from the declared aperture breakpoints is required.

The feedback loop is modeled after a leaky peak-detector circuit so that if the breakpoint determined after data fusion, \( \hat{b}_{F}(p) \) (using the port side as an example), approaches the nominal safe zone determined by the user, \([\hat{b}_0, b_0] \), the tracker reacts quickly; after the breakpoint recedes, the tracker slowly returns to the nominal safe aperture at a rate determined by the user. After some manipulation, this resolves to iteration of the equations:

\[
\hat{b}(p) = \begin{cases} 
\hat{b}_0 + \epsilon(p) & \hat{b}(p-1) - \hat{b}_{F}(p) \geq \tau_s \\
\hat{b}_P(p) + \tau_s & \hat{b}(p-1) - \hat{b}_{F}(p) < \tau_s 
\end{cases}
\]

where \( \tau_s \) is a ‘safety buffer’ specified by the user, intended to be the minimum gap between the start of the aperture and the breakpoint determined and \( \alpha = \exp \left\{ n_{max}^{-1} \ln(0.5/\tau_s) \right\} \) is the time constant for a decay to the nominal aperture in \( n_{max} \) pings.

The safe aperture tracker is executed after the fusion decision for each ping, and provides guidance for the processing of the following ping. Examples of the
The simplest explanation for problems observed is that the grazing angles at which the MBES observes the seafloor at too low. At very low grazing angles, little energy comes back and it becomes difficult to make a reliable detection. Normally, this is limited because of the maximum opening angle of the MBES, which is in turn limited by refraction correction and physical aperture construction limitations. However, on a significant slope, this condition can occur somewhere in the outer beams of the swath.

\[
g_0(p, b) = \frac{\Delta z(p, b)}{\sqrt{\Delta x^2(p, b) + \Delta y^2(p, b)}} \quad (\hat{b}(p) \leq b \leq \hat{b}(p))
\]

\[
g_1(p) = \text{med}_{0 \leq \delta_p < W_G} \left\{ g_0(p - \delta_p, b) \right\}
\]

\[
g_2(p) = \text{med}_{0 \leq \delta_p < W_G} \left\{ g_1(p - \delta_p) \right\}
\]

where \( \Delta x(p, b) = x(p, b + 1) - x(p, b) \) for ping \( p \) and beam \( b \) with corresponding definitions for the other variables, \( \text{med} \{ \cdots \} \) is a simple median filter, and \( N \) is the number of beams in the ping. The first median filter provides a rough estimate of gradient from the noisy first differences \( g_0(p, b) \); the second median filter provides an improved estimate through re-filtering. It would be possible to improve the estimate robustness by filtering over a wider window. However, the dual method used here re-uses previous filtering and is therefore more efficient.

To compute the maximum aperture allowed, the gradient is converted to slope \( s(p) = \arctan(g_2(p)) \), and the minimum and maximum angles allowed are computed:

\[
\hat{\theta}_G(p) = \max \left\{ \hat{\theta}_s - \min \left\{ -\hat{\theta}_p - s(p), -\hat{\theta}_p \right\} \right\}
\]

\[
\hat{\theta}_G(p) = \min \left\{ \hat{\theta}_s + s(p), \hat{\theta}_p \right\}
\]

with corresponding beam limits \( \hat{b}_G(p) = N/2 + \hat{\theta}_G(p)/\Delta \theta \) and \( \hat{b}_G(p) = N/2 + \hat{\theta}_G(p)/\Delta \theta \), where \( \hat{\theta} \) and \( \theta \) are minimum and maximum opening angles for the MBES, and \( \hat{\theta}_p \) and \( \hat{\theta}_s \) are the corresponding angles adjusted for any static roll applied to the transducer.

Finally, to make an output compatible with the other algorithms, we must assess the probability associated with the breakpoint detections \( \hat{b}_G(p), \hat{b}_G(p) \) for all beams in the ping. Since this algorithm does not compute this directly, we must define the structure instead. Based on the observation that this simple scheme generally underestimates the extent of the problem, we model the pdf as a half-Gaussian:

\[
N_h(x; \mu, \sigma^2) = \frac{2}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(x - \mu)^2}{2\sigma^2} \right\} \quad H(x - \mu)
\]

where \( H(x) \) is Heaviside’s step function. Setting the variance so that the distribution falls in the gap between the breakpoint detected and the safe aperture, the port and starboard distributions are \( P_L(p, b|\alpha_G) \)
and $P_{R}(p, b|\alpha_C)$, respectively:

$$\hat{P}_{L}(p, b|\alpha_C) = \mathcal{N}(b; \hat{b}_G(p), |\hat{b}_G(p) - \hat{b}(p)|^2/9)$$  \hspace{1cm} (10)

$$P_{L}(p, b|\alpha_C) = \hat{P}_{L}(p, b|\alpha_C) / \sum_{b} \hat{P}_{L}(p, b|\alpha_C)$$  \hspace{1cm} (11)

$$\hat{P}_{R}(p, b|\alpha_C) = \mathcal{N}(\hat{b}_G(p) - b; 0, |\hat{b}_G(p) - \hat{b}(p)|^2/9)$$  \hspace{1cm} (12)

$$P_{R}(p, b|\alpha_C) = \hat{P}_{R}(p, b|\alpha_C) / \sum_{b} \hat{P}_{R}(p, b|\alpha_C)$$  \hspace{1cm} (13)

where $P_{L}(p, b|\alpha_C)$ is defined over $0 \leq b < \hat{b}(p)$ and $P_{R}(p, b|\alpha_C)$ is defined over $\hat{b}(p) \leq b < N - 1$, and are both zero elsewhere.

### 2.2.2 Beam-to-beam Magnitude Vectors

The second algorithm implemented examines the ping-to-ping variance of the data, comparing against a theoretical variance computed from an MBES error model [4] as implemented for the CUBE MBES data processing algorithm [2]. Let $r(p, b)$ be the vector from MBES to sounding, $r(p, b) = (x(p, b), y(p, b), z(p, b))$, and compute first differences $\Delta r(p, b) = r(p, b + 1) - r(p, b)$, $0 \leq b < N - 1$. Then, dropping the ping and beam indicators for brevity, the uncertainty associated with the magnitude of $\Delta r$, $\Delta r = \left\| \Delta r \right\|$ is:

$$\sigma_{\Delta r}^2 = 2\Delta r^{-2} \left\{ \sigma_{xy}^2 (\Delta r_y)^2 + (\Delta r_x)^2 + \sigma_z^2 (\Delta r_z)^2 \right\}$$  \hspace{1cm} (14)

A robust estimate of the expected variance per beam for the magnitude is computed by median filtering:

$$\hat{\sigma}^2(p, b) = \text{med}_{0 \leq p < W} \left\{ \sigma^2_{\Delta r}(p - \delta p, b) \right\}$$  \hspace{1cm} (15)

and then a sample estimate of the observed variance is computed using the normal equations:

$$m(p, b) = \frac{1}{W_V} \sum_{\delta p = 0}^{W_V - 1} \Delta r(p - \delta p, b)$$  \hspace{1cm} (16)

$$s^2(p, b) = \frac{1}{W_V - 1} \sum_{\delta p = 0}^{W_V - 1} (\Delta r(p - \delta p, b) - m(p, b))^2$$  \hspace{1cm} (17)

Normal statistical techniques indicate that the test statistic $t(p, b) = (W_V - 1)s^2(p, b)/\hat{\sigma}^2(p, b)$ is distributed as a $\chi^2_{W_V - 1}$ random variable given the null hypothesis $H_0 : s^2 = \hat{\sigma}^2$, and hence an indicator function can be computed for each beam as a result of the test:

$$i_V(p, b) = \begin{cases} 1 & t(p, b) < \chi^2_{W_V - 1}(\alpha) \\ 0 & t(p, b) \geq \chi^2_{W_V - 1}(\alpha) \end{cases}$$  \hspace{1cm} (18)

for suitable significance $\alpha$.

Finally, we compute the probability that each beam is the true breakpoint by observing that the binary indicator should be distributed as a Bernoulli trial with probability $1 - \alpha$ on the safe side of the breakpoint, and probability $\alpha$ on the right. Therefore, the sum of the indicator to left and right of the breakpoint are distributed as a Binomial random variable $B_n(x; N, p)$:

$$B_n(x; N, p) = \binom{N}{x} p^x (1 - p)^{N - x}$$  \hspace{1cm} (19)

Therefore, the port and starboard mass functions are $P_{L}(p, b|\alpha_V)$ and $P_{R}(p, b|\alpha_V)$, respectively:

$$\hat{P}_{L}(p, b|\alpha_V) = \text{Bin} \left( \sum_{j=b}^{b(p)} i_V(p, j); \hat{b}(p) - b, 1 - \alpha \right) \times \text{Bin} \left( \sum_{j=0}^{b(p)} i_V(p, j); b + 1, \alpha \right)$$  \hspace{1cm} (20)

$$P_{L}(p, b|\alpha_V) = \hat{P}_{L}(p, b|\alpha_V) / \sum_{b} \hat{P}_{L}(p, b|\alpha_V)$$  \hspace{1cm} (21)

$$\hat{P}_{R}(p, b|\alpha_V) = \text{Bin} \left( \sum_{j=b+1}^{N - 2} i_V(p, j); N - 2 - b, \alpha \right) \times \text{Bin} \left( \sum_{j=0}^{b(p) - 1} i_V(p, j); b - \hat{b}(p) + 1, 1 - \alpha \right)$$  \hspace{1cm} (22)

$$P_{R}(p, b|\alpha_V) = \hat{P}_{R}(p, b|\alpha_V) / \sum_{b} \hat{P}_{R}(p, b|\alpha_V)$$  \hspace{1cm} (23)

where $P_{L}(p, b|\alpha_V)$ is defined over $0 \leq b < \hat{b}(p)$ and $P_{R}(p, b|\alpha_V)$ is defined over $\hat{b}(p) \leq b < N - 1$, and both are zero elsewhere. If required, the algorithm-specific breakpoints can be determined by maximization of the individual mass functions:

$$\hat{b}_V(p) = \arg \max_{0 \leq b < \hat{b}(p)} P_{L}(p, b|\alpha_V)$$  \hspace{1cm} (24)

$$\hat{b}_V(p) = \arg \max_{\hat{b}(p) \leq b < N - 1} P_{R}(p, b|\alpha_V)$$  \hspace{1cm} (25)

### 2.2.3 Maximum Range Estimate

The final algorithm uses the auxiliary information from the sonar meta-data about the user’s selected maximum range ($r(p)$), which is recorded on a ping-by-ping basis. A characteristic of the ‘noise’ observed is that it frequently reaches the maximum range set;
‘real’ data seldom does. Therefore, if the excess between the maximum range and the range per beam is computed, a comparison against the maximum range clearly shows the breakpoint.

To express ‘close’ to the maximum range, a pseudo-normal metric is computed from the maximum range:

\[
n(p, b) = \exp \left\{ -\frac{(\hat{r}(p) - r(p, b))^2}{2\sigma_k^2(\hat{r}(p))} \right\}
\]

\[
\sigma_k^2(x) = (kx)^2
\]

where \( r(p, b) = \|r(p, b)\| \) and \( k \) is a constant designed to express the degree of ‘closeness’ as a function of the maximum range. The use of a scaled version of the maximum range is suggested by properties of the data in our example datasets, but may not be everywhere optimal. The ad hoc nature of this choice is a potential difficulty with application and universality of this algorithm, a topic which is addressed in section 4.

To form a compatible binary indicator sequence in the form of equation 18, a simple threshold of the pseudo-normal metric is used:

\[
\hat{n}(p, b) = \max_{0 \leq \delta p < W_R - |W_R/2| \leq \delta b \leq |W_R|} \left\{ n(p - \delta p, b - \delta b) \right\}
\]

\[
i_R(p, b) = H(\hat{n}(p, b) - \tau)
\]

where \( \tau \in [0, 1] \) is a suitable threshold value, and the maximization of \( n(p, b) \) over a small window serves to compensate for the randomness of ranges after the breakpoint is reached. The remainder of the algorithm follows the form of equations 20–23 with breakpoints determined, if required, in the same way as equations 24–25.

2.3 Algorithm Fusion

Since each component algorithm generates evidence about the potential ‘noise’ breakpoint in the swath using a different metric over the observed data, combination of the results should allow for more robust decisions about the true location of the breakpoint. The failure modes of the metrics should be different, so when one algorithm fails to detect the breakpoint, or detects it inconsistent with the others, combination of the results should allow for compensation for the failure. Alternately, if the application is time or resource constrained, it should be possible to schedule use of the component algorithms in order to minimize the resource usage leading to the best available decision. The process of combination is a form of multi-sensor data fusion [1, 3].

2.3.1 Combination for Robustness

By construction, the component algorithms all generate compatible probability mass functions (pmf) over the range of beams between the edges of the swath and the ‘safe’ aperture maintained by the meta-algorithm. It would be possible to compute a breakpoint pair from each algorithm’s pmfs and then somehow combine the breakpoint beam estimates either directly, or as evidence for some form of Kalman filtering breakpoint tracker with suitable inertia [3]. However, since the algorithms should in theory all have significant mass in the same places, it is simpler to fuse the pmfs directly, and then determine the breakpoint from the fused pmf. Given some knowledge of the reliability of the algorithms expressed via a priori probabilities, this form of fusion resolves to computing the total probability:

\[
\tilde{P}_L(p, b) = \sum_{i \in \{G,V,R\}} P_L(p, b; \alpha_i)P(\alpha_i)
\]

\[
P_L(p, b) = \tilde{P}_L(p, b) \bigg/ \sum_{b} \tilde{P}_L(p, b)
\]

\[
\tilde{P}_R(p, b) = \sum_{i \in \{G,V,R\}} P_R(p, b; \alpha_i)P(\alpha_i)
\]

\[
P_R(p, b) = \tilde{P}_R(p, b) \bigg/ \sum_{b} \tilde{P}_R(p, b)
\]

where \( P_L(p, b) \) is defined over \( 0 \leq b < b(p) \) and \( P_R(p, b) \) is defined over \( b(p) \leq b < N - 1 \). The fused breakpoints for the ping are then computed analogously to equations 24–25:

\[
\hat{b}_F(p) = \arg \max_{0 \leq b < b(p)} P_L(p, b)
\]

\[
\hat{b}_F(p) = \arg \max_{b(p) \leq b < N - 1} P_R(p, b)
\]

which are then used to feed the ‘safe’ aperture tracker, equations 1–3, and flag the data outside the breakpoints as ‘not for use’.

Specification of the a priori probabilities is problematic. Specification by the user based on observed past performance of the component algorithms is, of course, the simplest option. This topic is still under active investigation, and is considered more in section 4.

2.3.2 Sequential for Speed

The primary goal of the algorithm here is to provide a robust means of identifying the breakpoints between good and bad data. However, if there are additional constraints on processor time (e.g., when operating in
Figure 4: Example data in intermediate depths in the south-west Pacific around the south-west shore of the island of Rota (14°09’ N, 145°12’E). Data is colored by depth and clearly shows the heavy ‘noise’ induced by a maximum depth $\hat{r}(p) = 300$ m. Starboard is to the right of the image.

Figure 5: Estimated slope of the example line. ‘Slope’ is not well defined in variable cross-track environments, but only a rough estimate is required here.

Figure 6: Estimated breakpoints using the grazing angle estimator. Note that the port side breakpoint is fixed at zero since there is only one estimate of slope, and it is always to starboard in this example.

3 Examples

The example here is focused on data from the south-west Pacific around the island of Rota (14°09’N, 145°12’E, part of the Mariana Islands group in Western Micronesia). The data was gathered with a Reson 8101 MBES in intermediate-deep hydrographic depths, and clearly show evidence of swath breakpoints on both sides of the swath, Figure 4. The problems in this example occur mostly on the starboard side of the swath, but occasionally break-through on the port side as shown here. The estimated slope of the surface is shown in Figure 5, indexed in pings from start of line. The line is approximately 9 min., or 2.2 km long, with a ping rate of $\sim 3$ Hz.

3.1 Component Algorithms

The example file was processed with $W_G = 10$ pings, $\bar{\theta} = 30^\circ$, $\bar{\hat{\theta}} = -30^\circ$, $s_0 = 5$ beams and $n_{\text{max}} = 20$ pings. The breakpoint locations predicted by the grazing angle algorithm, $(\hat{b}_G(p), \bar{b}_G(p))$, are shown in Figure 6, and the corresponding flagged data is shown in Figure 7. The breakpoints picked by the algorithm are generally good, although close examination of the filtering results shows that the algorithm does underestimate the extent of the problem, failing to flag many of the points that are affected by the noise problem. This algorithm does, however, provide basic information on the likely location of the breakpoints, and very rarely has any difficulties with outliers.

The breakpoints predicted from the variance based algorithms, $(\hat{b}_V(p), \bar{b}_V(p))$, are shown in Figure 8. (The algorithm used $W_V = 10$ pings, and $\alpha = 0.05$...
for the $\chi^2$ testing.) The breakpoints here show significantly finer detail than those of Figure 6, primarily because of the fine detail in the analysis of the data. Both sides of the swath are processed, clearly showing the `fail safe' nature of the algorithm where no problem exists. The algorithm has predicted rather more significant breakpoint extents than the previous algorithm, which are more in keeping with the expected values observed from the data. In areas of slight slope, Figure 9, the algorithm corresponds well with what a user might do, although it can be seen to slightly underestimate the extent of the problem on the starboard side of the swath. This in itself is not a very significant difficulty, since this is intended to be a pre-filter to an algorithm like CUBE, which is robust in itself. If the algorithm removes enough of the dubious data to improve the SNR sufficiently for the subsequent algorithm to correctly determine the true depth, then the user’s remediation time will be very small. However, the algorithm is also observed to over-estimate the problem when there is very significant slope or very rapid changes in bottom configuration, Figure 10, essential because the model of expected variance does not currently take these effects into account. This is a more significant problem since this data, pre-flagged, will never be seen by any subsequent algorithm. This example emphasizes the importance of having multiple algorithms available to counter-act each other in the fusion stage of processing.

Finally, the breakpoints computed from the range-limit algorithm are shown in Figure 11. (The algorithm was run with $W_R = 3$ pings, $k = 0.20$ and $\tau = 0.80$. Significance for the Bernoulli trial testing was $\alpha = 0.05$.) The level of detail and general range is similar to the values from the variance excess algorithm (Figure 8), and both sides of the swath are processed. The algorithm also appears robust to noise in both gently sloping and steep areas, Figures 12-13. This consistency of performance is a good property, but may be specific to this environment where the majority of problems are range-limited. This perfor-
Correctly Flagged

Overestimated Flagging

Figure 10: Example of flagging by variance excess algorithm in steep slopes and rapidly changing bathymetry.

Figure 11: Estimated breakpoints using the range limit algorithm. The estimates are as detailed as those from the variance excess algorithm, but are apparently more stable with fewer episodes of spikes and over-estimation.

performance may not translate to other regions where problems with low grazing angle become more pronounced, for example, whereas the variance excess algorithm would clearly detect this problem. Again, a requirement for fusion is clearly indicated.

3.2 Fusion Algorithm

The component algorithms were fused using equations 31–33 with *a priori* probabilities $P(\alpha_G) = 0.90$, $P(\alpha_V) = 0.35$, $P(\alpha_R) = 0.90$, which were assessed from experience with this dataset. The values reflect the judgements that the grazing algorithm information is diffuse, but almost always correct; the variance excess algorithm information is precise, but frequently inaccurate; and that the range data is generally both precise and accurate. The fused result, along with the components, is shown in Figure 14; only the starboard side is shown, since it illustrates most clearly the following discussion. Figures 15–16 show illustrative sections of the data file with flagged data, showing the combination properties of the fusion algorithm in better balancing the decision process.

The result in Figure 14 shows that the algorithm
Figure 14: Fused result of all breakpoints, showing starboard side only. The fusion algorithm attempts to balance the contributions of the various algorithms according to their expected reliability. This generally works well as long as the algorithms are suitably accurate. When the algorithms are precise but inaccurate, occasional bimodal results are possible, and the algorithm sometimes switches between modes rapidly.

Figure 15: Example of a gently sloping environment flagged using the fused breakpoints.

Typically performs well as long as the results from the component algorithms are suitably consistent. Figure 17 shows a portion of the breakpoint analysis in more detail around ping 500, and Figure 18 shows the corresponding probability mass functions for the fusion. Clearly, the contributions of the component algorithms are weighted and the fused pmf indicates an appropriate breakpoint. However, if there are significantly different assessments of breakpoint location, the fusion decision becomes more difficult. Figure 19 shows the situation at ping 689, and Figure 20 shows the corresponding pmf estimates. Here, the component algorithms are all precise about their estimate of the location of the breakpoint, but are mutually inaccurate. A subjective assessment of the breakpoint location indicates that the variance excess algorithm is closest to reality although the breakpoint is not clear here, and could be anywhere from beam 55–68. The range limit algorithm’s over-estimate of the breakpoint here is due to the configuration of the bottom in the region, which drops very quickly and then flattens to a plateau at almost the maximum depth. Consequently the appropriate metric (equation 26) is high for the plateau, leading to the early breakpoint pick. However, while the algorithm (correctly) uses the variance excess estimate in this case, the disjoint component pmfs lead to a basically bimodal distribution, and the very similar probabilities assessed for the two principal modes lead the algorithm to jump between modes from ping to ping. This process is clear in Figure 19, and re-occurs at intervals in the dataset. There are a number of potential methods to resolve this situation, and they are further addressed in section 4.

Further experiments were carried out with different a priori probabilities. It was found that the difficulties illustrated in Figure 19 increased as $P(\alpha_V)$ as might be expected from equation 33. The action of the grazing angle algorithm is to provide generalized location information, rather than precise estimates of the breakpoint. Increasing $P(\alpha_G)$ results in stabilization of the breakpoint estimates, less mode switching, and more frequent selection of more conservative breakpoint locations (i.e., where a breakpoint further from nadir is chosen).
4 Discussion

The algorithm presented here clearly detects the inconsistencies in the example data, and in general does so at the positions that would be indicated by a human operator (although such indications are, of course, very subjective). The complexity of the algorithms are not particularly high, although there are a number of parameters to determine which can have significant effect on the performance of the algorithm. Methods to reduce or ‘harden’ the parameter choice (i.e., make them estimatable physical parameters rather than arbitrary constants) are currently under investigation. The grazing angle and variance excess algorithms have well established theoretical support, but the range limit algorithm is more ad hoc (although ad hoc in the strict definition of the term). This makes the parameter choice for the range limit algorithm problematic, although in the experiments reported here only the maximum range scale was found to be particularly sensitive. Too low a value results in insufficiently robust detection of the breakpoint; too high a value results in over-estimation of the inconsistent region’s extent and subsequent over-flagging. It may be possible to improve on this situation by estimating the variance of the ranges with respect to the maximum range using the grazing angle algorithm’s estimate of the breakpoint as something suitably ‘safe’.

A similar objection might be made to the a priori
specification of algorithm performance prior probabilities. Indeed, there is no reason to believe that the performance of any algorithm should necessarily be a constant. Iteratively estimating the performance of an algorithm from the data would be significantly more robust, but the primary difficulty is that there is no objective measure of the true breakpoint. If may be possible to devise a penalty/reward system based on closeness to the fused breakpoint, e.g. postulate a penalty-reward function of the form:

$$ \delta P(x; d, r, m) = k \tanh (r(d - x)) $$

(36)

and then update the component algorithm prior probability at each ping step with:

$$ P_{p+1}(\alpha_i) = \min \{1, \max \{0, P_p(\alpha_i) + \delta P([b_i(t) - b_F(t)];d, r, m)\}\} $$

(37)

so that the prior probability is updated at each ping with appropriate checks to ensure that the value remains a valid probability. The reward function postulated here, $\delta P(x; d, r, m)$ is illustrated in Figure 21, and is adjustable via $d$ for how much difference to allow before applying a penalty; via $r$ for how quickly the penalty or reward accumulates with distance from $d$; and via $m$ for the maximum magnitude of penalty or reward at each step. This structure would allow the designer to specify how close is ‘close enough’, and the adaptation rate of the algorithm. Methods of this type are the subject of on-going research.

The stability of each of the component algorithms is paramount to the construction of a skillful detection and fusion scheme. The evidence from the example dataset shows that three algorithms may be insufficient to make the fusion scheme suitably robust. Since the grazing angle algorithm only provides diffuse information, failure of one of the other two algorithms leads to an essentially bimodal fused pmf and hence the potential for a hard-failure decision process. Addition of one or more algorithms should result in a better definition of the fused pmf in the face of individual algorithm failures. Other potential information sources include backscatter imagery associated with the data, and analysis of the beam-to-beam turning angles. As an alternative, a meta-algorithm could be implemented to monitor the breakpoints as a sequence and check that the breakpoint chosen makes contextual sense [6]. Estimation of the probability $P_L(p, b_F(p, t - 1))$ would be an alternative metric to attempt to ensure that the fused solution stayed within one mode, as would minimizing the distance $|b_F(p) - b_F(p - 1)|$. Of course, ensuring that the algorithm did not follow an inadvisable track consistently would also be essential, and such algorithms are the subject of current research.

This problem is a specific example of a more general problem with inconsistency of swath data, particularly where the data is being monitored in real-time as it is being captured. Many observed problems with field data can be traced to operators who do not always appreciate the level of data quality that can theoretically be achieved with the systems being used. If a system was available that could predict the performance of a sonar system given the real-time environmental conditions, and then quantitatively rate the current data according to this prediction, it would provide valuable real-time feedback for operators as to the likely quality of the data being collected. One potential method for implementing such a system would be to continually monitor a number of different conditions using schemes such as the component algorithms used here, which are specifically designed to look for particular problems. Appropriate fusion of the individual schemes should lead to an overall quality indicator per beam, which would then be readily summarized for the user. Although presently only a sketch, such a scheme has the potential to assist in diagnosis of problems, as well as their indication to the operator in a timely manner.

5 Conclusions

Many MBES systems can produce inconsistent data over an extensive range of the swath due to effects of heavily sloping bathymetry and user-specified maximum acoustic ranges. This ‘noise’ data is spatially correlated and therefore difficult for either humans or algorithms to remediate. This paper has shown

Figure 21: Potential component algorithm probability penalty-reward function. Careful parameterization of equation 36 allows control of adaptation rate, allowable slope factor and harshness of penalization.
that relatively simple algorithms can be used to
detect causes of this problem, although no one algorithm
is individually sufficient to do so reliably everywhere.
A meta-algorithm has been proposed to take the in-
dividual algorithms and fuse them into a more robust
solution.

Example data from the south-west Pacific has been
used to illustrate the algorithm in practice, and high-
light the benefits of a data fusion scheme. The flags
applied automatically to the data are analogous to
those (subjective) decisions that a human operator
would make. Future directions include more compo-
nent algorithms, more robust data fusion and methods
for adaptive update of the component algorithm prior
probability functions.

Acknowledgements

The support of NOAA grant NA90G0241 is grate-
fully acknowledged. The data used in the examples
was supplied by the Captain and Crew of the NOAA
Ship RAINIER (Alaska) and Scott Fergusson, NOAA
Pacific Islands Fisheries Science Center (Hawai`i).

References

data fusion: A comparative review with classifi-
[2] B. R. Calder. Automatic statistical processing of
multibeam echosounder data. Int. Hydro. Review,
racy estimation of canadian swath (multibeam)
and sweep (multitransducer) sounding systems.
Technical report, Canadian Hydrographic Service,
1995.
[5] NOAA Office of Coast Survey. NOS Hydrographic
Surveys Specifications and Deliverables. NOAA
and Dynamic Models. Springer-Verlag, 2 edition,
1997.