FREQUENCY TRACKING BASED ON A DYNAMIC PROGRAMMING SEARCH OF POTENTIAL TRACKS

C R Walters

Cranfield University, Royal Military College of Science, Shrivenham

1. INTRODUCTION

A frequency tracking technique applicable to narrowband passive sonar is described. The technique is based on a search of all potential tracks in a waterfall time-frequency display using dynamic programming concepts. The lowest-cost track, or set of tracks, is identified using a cost-function derived from the ratio of track weight to average inter-track spectral level, the track weight being calculated from a summation of the spectral magnitudes in the frequency bins which make up the track under consideration. To improve the performance of this tracking technique, two approaches are currently being investigated; the first is to process the track data before the application of the cost-function with the objective of enhancing true tracks; the second approach is to extend the scope of the cost-function such that it measures track parameters other than magnitude. Specifically, phase alignment of track components is being investigated as an indicator. Results of the application of the technique to simulated data are presented and compared to another dynamic programming based tracker which utilises a Hidden Markov model.

2. BASIC CONCEPTS

The task of a tracker in the context of a narrowband passive sonar system, is to identify and follow a target signal, modelled as a sinusoid in additive noise with amplitude and phase subject to random variations due to the effects of the propagation medium. The frequency of the sinusoid is also subject to similar variations but more significantly, will change due to Doppler effects resulting from relative motion between source and sonar platform. These variations can be used as the input data for a passive localisation scheme and it is important that a tracker provides a good estimate of target frequency as a function of time as well as indicating the continued presence of a target.

Tracking as described here is generally referred to as frequency- or line-tracking and has similarities to the tracking of targets used in radar or optical systems, the methods employed being similar; typically, a Kalman filter would be employed to provide an estimate of the target position over an extended period of time. Recently however, other techniques have been proposed and investigated, in particular those based upon a Hidden Markov Model, where the track is modelled as a Markov chain and a set of probabilities govern the initial location of the track; thereafter transition probabilities govern how the track changes position. The model also makes assumptions about the signal and noise characteristics such that the measurement system can be modelled, as is done by a Kalman filter. As the measurements are taken, the most likely track which fits the model is found by a dynamic search using a likelihood-based cost-function.

3. DYNAMIC PROGRAMMING

Dynamic programming techniques are commonly used to select one object from an extensive set of similar objects, the dynamic nature of the process being that the scope of the search is increased as the search proceeds and that the current search encompasses previous results. The technique has a long history but was brought to modern attention by the work of Bellman [1] and can be applied to a diverse range of problems such as route-finding and production-line optimisation. It is commonly implemented as the Viterbi algorithm [2] which can be considered as an efficient solution to finding the optimal state sequence of a Hidden Markov Model using dynamic programming techniques. The dynamic nature of the Viterbi algorithm in this tracking context is that as each measurement is taken the range or scope of the search is increased. The observation consists of the measured track position in the current displayed line; the search range then extends from the start of the display to the current measurement position.

There are a number of variations to the basic scheme described above. Arnold et al [3], working with satellite images, base the dynamic search on a statistical cost-function derived from individual cost-functions for each track state-space domain such as object measured position, velocity, brightness etc. The cost-function in each domain is calculated from the ratio of supporting to contradictory evidence, thus allowing different parameters to be combined in one cost-function. Additionally, multiple targets are handled by removal of detected targets from the initial data prior to a further pass through the tracker. This is an extension of earlier work by Barniv et al [4, 5] where the data, again originating from satellite images, was passed through a filter bank which matched all possible target trajectories in target space. Barrett et al [6, 7, 8] described HMM-based frequency trackers which produced either a discrete frequency-time track or a continuous 'mean-cell occupancy' parameter. Later work by Barrett [9] introduced phase as well as amplitude into the HMM offering improved performance at some increase in computation. Martinerie et al [10, 11, 12] combined target tracking and motion analysis in their work. Given a set of position measurements at successive time intervals, then all possible elementary target paths were constructed and evaluated. The evaluation required the calculation of a path likelihood from velocity and acceleration components required to achieve that path. Transition probabilities from one elementary path to another were also computed. This approach differs from the described other work in that the HMM uses an elementary path as a Markov state in place of the more common target position or single frequency cell.

Although a dynamic programming search is commonly implemented using the Viterbi algorithm, it is not always so and the method to be described, referred to as the Graph-theoretic tracker because of its derivation, has similarities to the Viterbi algorithm but it also has distinct differences. These differences can be illustrated by considering the example of an F-bin by T-line display which is to be processed by both tracking systems. As each spectral line is made available to the Viterbi system, the most likely track would be calculated based upon the 'surviving' track prior to the arrival of that line. Thus the track growth direction is vertical, along the line of the track. By comparison, the graph-theoretic tracker would pass through the complete display searching for the most likely complete track from the set of all possible tracks, the search direction being horizontal. This immediately leads to a direct comparison between the two methods in terms of the amount of computation required. In the case of the Viterbi tracker, the addition of a new line requires a search over F possible locations for the growth of the track. From the initial track position therefore, FT possible tracks require evaluation. By comparison, the graph-theoretic tracker searches through FT^2 potential tracks, assuming a deviation of one frequency bin from one line to the next. The graph-theoretic tracker will now be described in more detail.

4. GRAPH-THEORETIC TRACKER

The principle of this tracker, as described by Lanfear and Constantinides [13], is that the frequency-time space can be mapped to a graphical network with each frequency bin at an instant in time equating to a node in that graph. A partition of the graph is then equivalent to a potential track, assuming that certain

FREQUENCY TRACKING BASED ON DYNAMIC PROGRAMMING

constraints are made on the partition. For example a partition must progress from line (spectral record) to line; it cannot progress along a line since a track cannot occupy more than one frequency bin at any one instant. (Although the signal energy may be spread over adjacent frequency bins, dependent on the window used in the FFT process and the signal frequency, it is assumed that tracks have a width of one bin.) To optimally partition the network, a lowest cost solution is sought using a cost function which is derived according to the application. In the case of the GT tracker, a basic version of the cost function is related to the ratio of track weight to inter-track noise level, equivalent to a measure of track signal-to-noise ratio. The procedure is therefore :

- generate all feasible partitions for the network as an ordered set; track geometry constraints including limits on track deviation from one line to the next are introduced at this stage [14]
- determine the optimum partitioning using a DP search based on a cost function as described; the partitioning may produce multiple partitions or tracks [15]

The significant tracker parameters are the maximum frequency deviation from one spectral record or line to the next, the width of the spectral window and the number of lines to be considered. Generally, a full width spectral window is processed, limited to a small number of lines to reduce the amount of processing. This window is updated by inclusion of each spectral record as it is made available and rejecting the oldest line. This produces a significant degree of overlap which can be reduced by delaying the processing until the complete window is refreshed. Alternatively, if processing with overlap a 'voting' scheme can be introduced to increase the robustness of the tracker, only accepting a frequency bin as part of a track if it appears in a number of consecutive passes through the tracker.

The differences in computation and search direction between the HMM and the GT trackers have already been described. A further difference between the two systems is that potential track geometries are composed of transitions, governed by a transition probability matrix in the former whereas all transitions are considered equally likely to occur in the latter. Constraints on the track geometries in this case relate to the requirement for the track to progress and on the track source. Thus if the source of track variation is Doppler shift, the maximum frequency shift from one line to the next can be related to the maximum expected speed of potential targets. It should also be noted that the HMM tracker searches for the most likely single track which fits the measurement; the GT tracker produces a set of tracks which minimise the cost-function, and so can handle multiple tracks without difficulty.

An outline procedure has been described above. Analysing the sequence in more detail, the enumeration of track geometries establishes the maximum range of the dynamic search which is limited by constraining track geometry. For an L-line display, if a track can deviate by a maximum of D bins from one line to the next, there are $(2D + 1)^{L-1}$ potential track shapes, illustrated in Figure 1 for L=3 and D=1

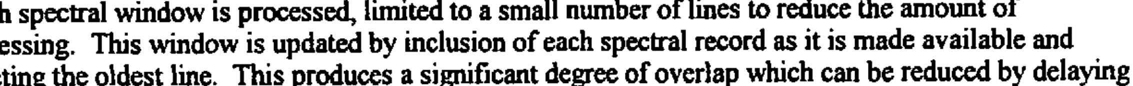


Figure 1 Range of possible track geometries for $L = 3$ and $D = 1$

Therefore, for an $F \times L$ display, there are $F(2D + 1)^{L-1}$ possible tracks, illustrated in Figure 2. Note that the partitioning process requires there to be a 'dummy' or reference track in the first and last positions.

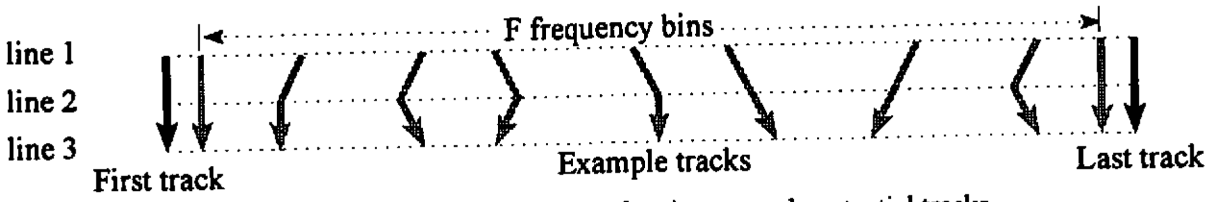


Figure 2 Display segment showing example potential tracks

5. TRACK EVALUATION

The evaluation of the enumerated set of all tracks, to derive the optimal partition, uses a cost function which balances the components in support of the hypothesis that the partition is a track and the components which contradict this hypothesis. For the search to be dynamic, the cost-function of a particular track j, C_j, must be separable into the cost of track i, C_i, where $j > i$ and track i has previously been evaluated, and the cost of the space between j and i, C_{ij}. The minimum cost partition of the entire display is then the value of i which minimises C_j when j is equal to the last track. This value of i will then be minimised by a track number, r for example, where $i > r$, which is minimised by a track s where $r > s$ etc. Thus the scope of the search gradually extends as the search proceeds and multiple tracks are automatically considered.

The dynamic search is based therefore on the minimisation of the cost of partition or track j:

$$C_j = \min_{i=0}^{j} \left[C_i + C_{ij} \right]$$

where the minimisation range is from first to last track, and the expression has previously been solved for track i, giving C_i. The inter-track cost, C_{ij}, is application-dependent and in this instance can be related to the ratio of track weight to inter-track noise. The components of the inter-track cost are calculated from the DFT of consecutive blocks of the signal data which form the frequency-time spectral display.

If it is assumed that the target signal is the sum of M sinusoids, the received sampled-data signal is:

$$x(n) = \sum_{m=0}^{M-1} A_m \cos(\omega_m n T_s + \theta_m)$$

where A_m, ω_m and θ_m are the amplitude, frequency and phase of the m^{th} component and T_s is the sample interval. If the recorded signal is divided into N-point records, the k^{th} record has the sample values :

$$x_k(n) = \sum_{m=0}^{M-1} A_m \cos(\omega_m n T_s + \theta_m^0 + k\omega_m N T_s)$$

where θ_m^0 is the initial phase of the m^{th} frequency component in the first block, for which $m = 0$.

FREQUENCY TRACKING BASED ON DYNAMIC PROGRAMMING

The Discrete Fourier Transform of the k^{th} block is :

$$X_k(p) = \sum_{n=0}^{N-1} x_k(n)\, e^{-j2\pi np/N}$$

$$= \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} A_m \cos(\omega_m nT_s + \theta_m^0 + k\omega_m NT_s)\, e^{-j2\pi np/N}$$

Expressing the cosine as the sum of two exponentials, re-arranging, and summing the DFT geometric progression produces the expression:

$$X_k(p) = \frac{1}{2} A_0\, e^{j\theta_0^k} \left(e^{j\frac{1}{2}(N-1)\theta_+} \frac{\sin(\frac{1}{2}N\theta_+)}{\sin(\frac{1}{2}\theta_+)} + e^{j\frac{1}{2}(N-1)\theta_-} \frac{\sin(\frac{1}{2}N\theta_-)}{\sin(\frac{1}{2}\theta_-)} \right)$$

assuming that only one component is present, ie $M = 1$. In this expression θ_+ and θ_- are defined as :

$$\theta_+ = \frac{2\pi p}{N} + \omega_0 T_s = T_s(\omega + \omega_0)$$

$$\theta_- = \frac{2\pi p}{N} - \omega_0 T_s = T_s(\omega - \omega_0)$$

At the value of p such that $\omega \approx \omega_0$ the θ_- component reduces to unity and the complex DFT sample is :

$$X_k(p) = \frac{1}{2} A_0\, e^{j\theta_0^k} \left(1 + e^{j(N-1)N\omega_0 T_s} \frac{\sin(N\omega_0 T_s)}{\sin(\omega_0 T_s)} \right)$$

The expression in brackets represents leakage effects. If each block contains an integral number of cycles, all of the signal energy is contained in one frequency bin and the leakage term will have the value 2. More usually the signal energy will be spread among adjacent bins according to the $\sin x / x$ function. Note that the initial phase of each block carries through the DFT and can be measured from the DFT values at the signal sample value. The track weight function is derived from the sum of the magnitudes of the spectral components along the line of a track :

$$W_q = \sum_{k=0}^{K-1} \left| X_k(p_q^k) \right|$$

In this instance, track q is composed of K frequency bins; p_0^q is the frequency bin number on the first display line of the track, p_1^q the bin number of the second, etc. Two example track weight functions are shown in Figure 3 from a 16-frequency-bin by 3-line display segment calculated from successive 128-

FREQUENCY TRACKING BASED ON DYNAMIC PROGRAMMING

point DFTs of a single sinusoid. It is assumed that the maximum track deviation, $D = 1$. It should be noted that the track-weight function does not map linearly to frequency bin or display line because of the assumed potential track geometries.

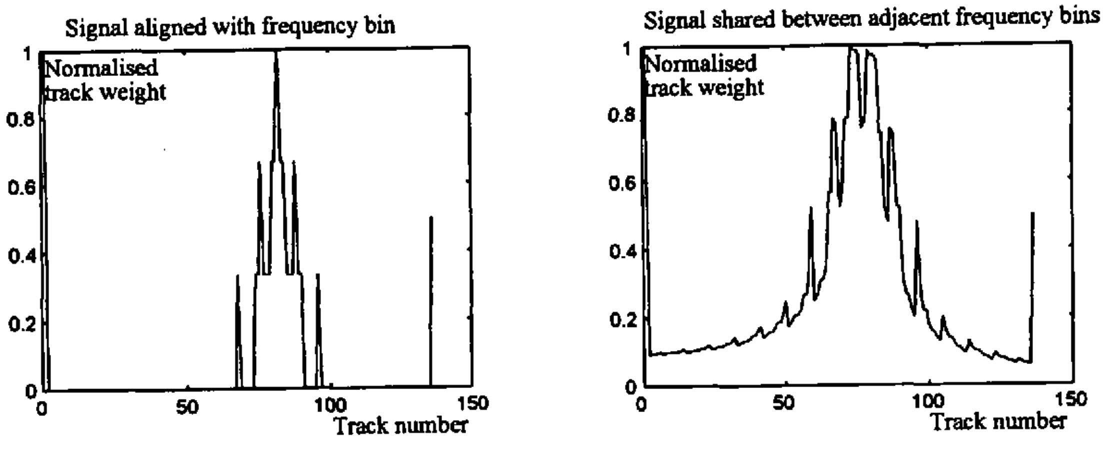


Figure 3 Example track-weight functions

Having evaluated the track-weight function, the basic inter-track cost function is:

$$C_{ij} = a\mathrm{N}_{ij} - b\,W_i$$

where a and b are the weights of the cost-function components, adjusted according to performance, N_{ij} is the inter-track noise and W_i the weight of the i^{th} track, defined above. The inter-track noise is found from:

$$\mathrm{N}_{ij} = \frac{\sum_{q=i}^{j} \sum_{k=0}^{K-1} \left| X_k\left(p_q^{\,k}\right) \right| - W_i - W_j}{\varepsilon_{ij}}$$

where ε_{ij} is the number of elements between the two tracks i and j.

In addition to the track weight function, phase alignment is also used as a track discriminator, the underlying concept being that a sinusoidal component captured in equal length blocks will have a phase component which progresses linearly with block number. A variation of this concept is used by Barrett et al [16] as a frequency estimator in the Phase Interpolation Estimator method. Here the concept is used to identify frequency components and track them from one block to the next. If the phase of the complex frequency sample $X_k(\omega)$ is $\phi_k(\omega)$ where $\omega = 2\pi p/Ts$ is the frequency in the p^{th} frequency bin and k is the display line number, then phase differences can be evaluated along the track components:

$$\Phi_q(k) = \phi_k\left(\omega_{q,k}\right) - \phi_{k-1}\left(\omega_{q,k-1}\right)$$

The frequency, $\omega_{q,k}$ is the frequency bin of line k of track q. The phase differences will be a constant if measured along the frequency bins which constitute a track, but vary for other incorrect tracks. Thus by calculating the variance of the phase differences for all tracks an indication of the true track position should be presented, when the variance or Mean Square Error is a minimum. There are some difficulties however; a phase discontinuity occurs at the frequency bin which corresponds to the signal, and this introduces large discrepancies in the phase differences if evaluated for a test track which crosses the true position of the track; the periodic nature of phase measurement causes problems in the unwrapping of phase differences also . An example of the phase alignment function is illustrated in Figure 4. Because of the many discontinuities - due to the non-uniform nature of the relationship between track number and frequency bin, a smoothed version is used as the track discriminator.

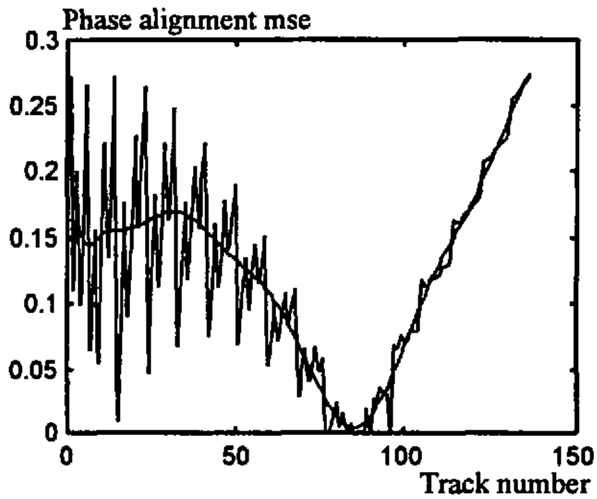
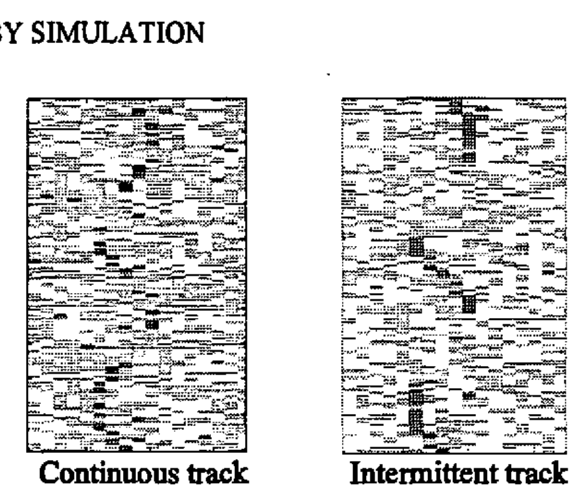


Figure 4 Example phase alignment function

In addition to the track discrimination based on track weight and alignment, processing in track-weight space is being investigated to reduce the effects of noise which acts to introduce spurious track elements in the processed display or to suppress track components. Actions performed on the track-weight function include filtering to suppress noise, and matched filtering to highlight the characteristic track-weight function. The effects of such processes will be described in the section relating to the results of the simulations.

6. COMPARISON BY SIMULATION

To assess the performance of the GT tracker it was compared to the HMM tracker using a simulated narrowband signal in noise, the frequency of the signal varying sinusoidally. Two versions were used, one continuous, the other intermittent, the signal parameters chosen to match the simulated data used in the work of Barrett et al [6]. The parameters of the two tracking systems were varied to determine the sensitivity to such variations and, in the case of the GT tracker, different forms of cost function were used. Model parameter variation is important in the case of the HMM since exact knowledge of these parameters can provide an ideal but unrealistic environment for tracking.



Continuous track **Intermittent track**

Figure 5 Simulated tracks for comparison of tracking systems

7. SIMULATION RESULTS

The HMM system was capable of significantly better performance than the GT system, but only if the assumed model parameters matched the presented data. Values which were deliberately varied from the optimal values produced significant reductions in tracking ability, particularly when the intermittent data set was presented. Variation of model signal amplitude and noise variance for example, which affected the measurement matrix elements produced the effects illustrated; variation of the assumed parameters in the transition matrix had similar effects. There are methods of matching the assumed model parameters to the presented data using the Baum-Welch [17] reestimation formulas but these were not implemented.

Figure 6 shows some typical results for the HMM tracker, concentrating on the intermittent signal. Since the effect of varying the HMM parameters was either to suppress the tracking of the test signal or to continue tracking when the signal was no longer present, then the intermittent input provided a more stringent test for the tracker.

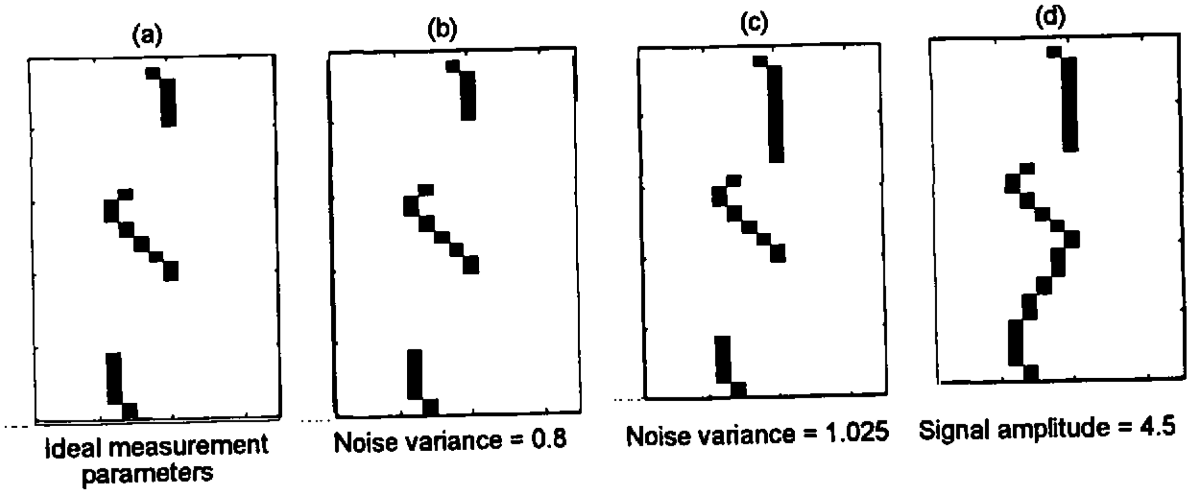


Figure 6 Typical results from HMM tracker

Figure 6.(a) shows the tracker output when the Markov model parameters match the simulated signal and noise values. Unsurprisingly, the tracker output matches the intermittent signal. Figures 6.(b), (c) and(d) illustrate the effects of a mismatch between the assumed model parameters and the simulated data. Generally, strong signal components are tracked well; weaker components are disregarded, resulting in a false extension of track component. Figures 6.(b) and (c) show the effects of under and over-estimation of the system noise, figure 6.(d) the effect of over-estimation of the signal amplitude. In the case of the latter, the HMM tracker becomes a peak follower.

Some representative results for the GT tracker are shown in Figures 7 and 8. Figure 7 illustrates the output of the tracker using correlation of the track-weight function with a noise-free 'replica' before the cost-function evaluation. The intermittent and the continuous input signals in noise are used, with the tracker advancing through the display in either 1-line or 3-line increments. (The display is processed in a 17-frequency bin by 3-line horizontal strip.) Advancement by one line at a time allows for a frequency bin voting procedure, since each line is processed on three consecutive occasions, thus allowing for spurious track elements to be more clearly identified,. since the displayed intensity of the tracker output is proportional to the number of times that a frequency bin is identified as being part of a track.

Figure 8 indicates the effects of using alternative track-weight processing, in particular median filtering, a combination of correlation and median filtering, and no processing. These three cases should be compared to the output in Figure 7.(b), which represents the case when correlation only is used. Finally, figure 8.(d) shows the tracker output when the display is processed in a 5-line strip rather than 3-lines. Note: the phase-alignment routines were not tested with this data set since it was not complex.

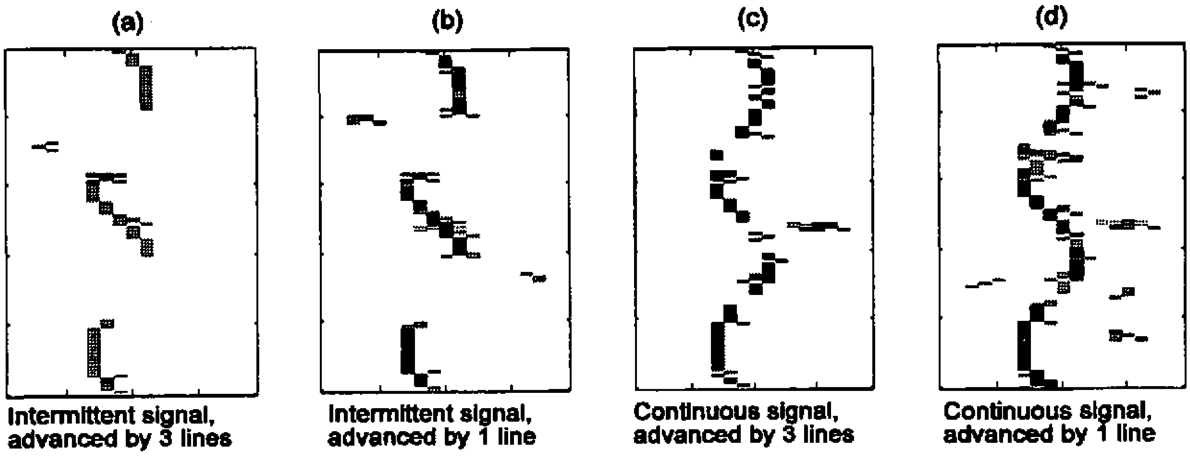


Figure 7 Graph-theoretic tracker, employing correlation of the track-weight function before cost-function evaluation; display processed in a 17-bin by 3-line strip, advanced in steps of either 1 or 3 lines

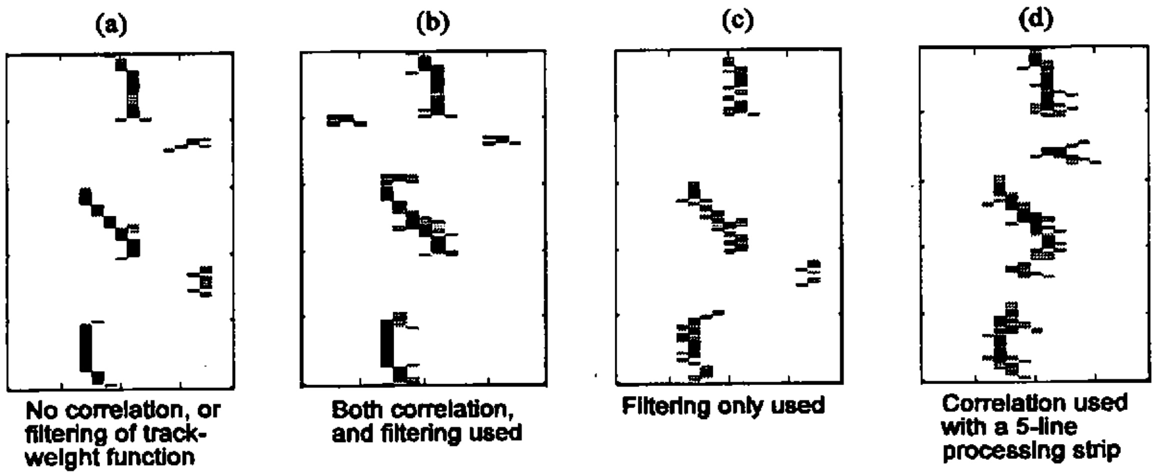


Figure 8 Graph-theoretic tracker, employing various processing schemes before cost-function evaluation, 8.(a) - (c), and a larger processing window - 17-bin by 5-lines, 8.(d)

FREQUENCY TRACKING BASED ON DYNAMIC PROGRAMMING

8. CONCLUSIONS

Because of the differences between the HMM and the GT tracker the simulation results are different. The HMM is a single-frequency tracker and errors occur as mis-tracking, most commonly a continuation of the track when the signal is no longer present; the multiple track capability of the GT tracker results in errors such as spurious track elements and missed track components. The 'voting procedure', made possible by incorporating an overlap into the processing scheme enabled spurious track elements to be more easily identified. Despite the differences between the trackers, both systems track the noisy signal; in the case of the HMM mismatch between the model parameters and the measured data results in mis-tracking; in the case of the GT tracker, the noise present introduces missed and false track elements.

9. REFERENCES

[1] R BELLMAN, *Dynamic programming* Princeton University Press (1957)
[2] G D FORNEY 'The Viterbi algorithm' *Proc. IEEE* 61 (1973)
[3] J ARNOLD, S SHAW & H PASTERNAK 'Efficient target tracking using dynamic programming' *IEEE Trans AES* 29 (1993)
[4] Y BARNIV 'Dynamic programming solution for detecting dim moving targets' *IEEE Trans AES* 21 (1985)
[5] Y BARNIV & O KELLA 'Dynamic programming solution for detecting dim moving targets, Part II : Analysis' IEEE Trans AES 23 (1987)
[6] R F BARRETT & R L STREIT 'Frequency line tracking using hidden Markov models' *IEEE Trans ASSP* 38 (1990)
[7] R F BARRETT & A K STEELE 'Frequency line tracking algorithms' *Proc NATO Advanced Study Institute Conference on Underwater Acoustic Data Processing,* (1988)
[8] R F BARRETT, B G QUINN & S J SEARLE 'The estimation and HMM tracking of weak narrowband signals' *Proc ICASSP* (1994).
[9] R F BARRETT & D A HOLDSWORTH 'Frequency tracking using hidden Markov models with amplitude and phase information' *IEEE Trans SP* 41 (1993)
[10] F MARTINERIE & P FORSTER 'Data association and tracking using Hidden Markov models and dynamic programming' *Proc ICASSP* (1992)
[11] F MARTINERIE & P FORSTER 'Data association and tracking from distributed sensors using hidden Markov models and evidential reasoning' *Proc CDC* (1992)
[12] F MARTINERIE & C BARRAS 'Continuous HMM Tracking' *Signal processing VII : Theories and Applications* EurASP (1994)
[13] T A LANFEAR & A G CONSTANTINIDES 'Graph partitioning with applications to signal processing' Imperial College Signal Processing section, Technical report, (1986)
[14] M BELLMORE & P A JENSEN 'An implicit enumeration scheme for proper cut generation' *Technometrics* 12 p775-788 (1970)
[15] P A JENSEN 'Optimum network partitioning' *Operational Research* 19 p916-932 (1970)
[16] R F BARRET & D R A McMAHON 'Comparison of frequency estimators for underwater acoustic data' *JASA* 79 (1986)
[17] L R RABINER & B H JUANG 'An introduction to Hidden Markov Models' *IEEE ASSP Magazine* (1986)

UNDERWATER ACOUSTIC DATA TRANSMISSION IN SHADOW ZONES WITH PSEUDORANDOM WAVEFORMS

Yu. Zakharov and V. Kodanev
N.N. Andreev Acoustics Institute, Russian Academy of Sciences,
ul. Shvernika 4, Moscow, 117036, Russia
E-mail svfedorov@glas.apc.org

Acoustic signals only reflected from a ruffled surface and a rough bottom arrive in acoustic shadow zones in an ocean. As a result, the energy of signals received in shadow zones is distributed (scattered) on a delay-Doppler shift domain. An effective way to increase reliability of data transmission under such conditions is to use a delay and Doppler shift diversity reception.

Results of an experimental study of underwater acoustic data transmission with a large ensemble of pseudorandom waveforms are described. Three adaptive reception algorithms are considered. The algorithms are based on an estimation of current distribution of the received signal energy on the delay-Doppler shift domain and a coherent or incoherent weight addition of signal scattered components. The experiments were carried out in first and second shadow zones of Indian Ocean. Experimental results have shown that under these conditions the reception with delay and Doppler diversity provides essential improvement of reliability in comparison with case when only delay diversity is used.