New techniques for the deinterleaving of repetitive sequences

H.K. Mardia, BSc, PhD

Indexing terms: Algorithms, Radar and radionavigation, Signal processing

Abstract: Radar signals are generally characterised by repetitive patterns in time. An ESM receiver must intercept and identify several interleaved radar signals. Time-of-arrival (TOA) deinterleaving is employed in ESM processing to identify and extract the pulses of each radar signal. This task is extremely processor intensive and new techniques are required to operate on complex signals in high pulse densities. A new algorithm employing novel techniques is presented for fast, accurate deinterleaving of several repetitive signals. A cumulative TOA difference histogram gives an indication of probable pulse repetition intervals (PRIs) with a minimum number of computations. Validation and identification is given by searching for a sequence of these pulse intervals. The technique presented is less sensitive to interfering pulses and more robust to missed pulses than conventional published techniques. Weighting is used to enhance detection of sequences and a three-pulse priming sequence dramatically reduces unsuccessful searches. By employing a learning process, the efficiency is increased still further. The application of this algorithm to agile PRI signals is shown.

1 Introduction

Time-of-arrival (TOA) deinterleaving is a vital part of electronic support measures (ESM) processing. PRI (pulse repetition interval) determination is necessary to separate the pulses of a given radar from a background of pulses for radar identification and to enable jamming. The parameters associated with the extracted pulses can be analysed to determine parameter variations with time, e.g. scan pattern analysis. High pulse densities containing complex radar signals are projected for future radar scenarios. A significant proportion of the pulse measurements will be corrupted in these conditions [1]. Therefore techniques are presented which are tolerant to missed measurements and which reduce the effect of measurement errors. As unambiguous identification cannot be guaranteed, the algorithms should provide a best fit to the data and indicate confidence levels.

2 TOA deinterleaving

Each radar can be characterised by a pattern of pulse intervals that repeats from a given start time (phase). In the simplest case of the stable PRI signal, only one PRI value is repeated. The range of PRI values is several μs to several ms. The PRI can also be staggered, where several PRIs form a frame which is repeated, or jittered, where the PRI deviates around a nominal value.

The proposed techniques extract stable PRI sequences and are used to analyse staggered PRI, jittered PRI and scanning radar signals. Current design trends are towards radars capable of greater PRI agility. These signals will be more difficult to extract, and greater reliance must be made on instantaneous pulse parameters, e.g. radio frequency (RF), direction of arrival (DOA) etc., to provide sorting prior to TOA deinterleaving [2] as shown in Fig. 1. Thus, received pulses are first separated into sets of

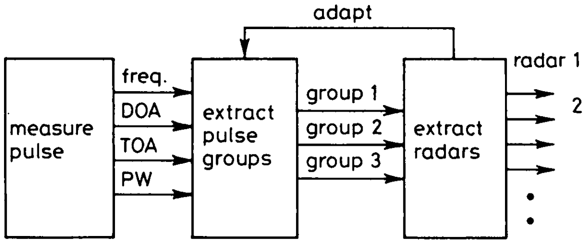

Fig. 1 *Receiver architecture*

similar pulses before attempting TOA deinterleaving. However, the environment contains similar radar types with agility which may not be resolved by grouping instantaneous parameters and thus TOA analysis is necessary. The TOA deinterleaving algorithm is then used to extract all radar patterns that can be reliably recognised, leaving a residue of pulses which can be analysed using other techniques.

TOA is measured at the leading edge of each pulse and is represented as a digital word. Arithmetic computations are performed on a sample of these TOA words by the deinterleaving algorithm. A sufficiently large sample must be taken such that the signal patterns are apparent.

Thus the sample consists of a sequence of events. We define the sample interval k as the TOA measurement resolution and the sample length as N sampling intervals (SI). Thus k is a scaling factor between the stored integer values and actual time in seconds. The TOA of each pulse can be represented by a delta function, i.e. value 1 at the appropriate sampling interval; otherwise the sample intervals are assigned value 0 (Fig. 2). The pulse width and amplitude information is discarded. Each TOA is measured as an integral multiple of the sampling interval. Thus, the ith stable PRI sequence α_i with a PRI of m_i SI, a start time of q_i SI and n_i pulses in the sample (the function 'int' is the greater integer less than the operand) can be written as

$$\alpha_i = \sum_{r=0}^{N} f_i(rk) \tag{1}$$

Paper 6754F (E15), first received 30th August 1988 and in revised form 21st April 1989

The author is with Filtronic Components Ltd., Royal London Industrial Estate, Acorn Park, Shipley, West Yorks. BD17 7SW, United Kingdom

where

$$f_i(rk) = \begin{cases} 1 & \text{when } r = am_i + q_i,\ 0 \leqslant a \leqslant \operatorname{int}\left(\dfrac{N - q_i}{m_i}\right) = n_i \\ 0 & \text{otherwise} \end{cases}$$

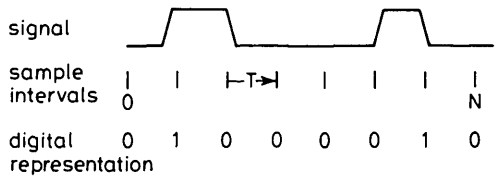

Fig. 2 *Sampled pulse train*

and r, a, m_i, q_i and n_i are positive integers.

The sample of pulses to be sorted consists of several interleaved signals. Where events coincide, only one event is indicated, and thus the resultant sample P is the logical OR of the x individual sequences and is thus represented by the 'max' function:

$$\begin{aligned} P &= \sum_{1}^{x} \alpha_i \\ &= \sum_{r=0}^{N} \max\,[f_1(rk), f_2(rk), \cdots f_x(rk)] \\ &= \sum_{r=0}^{N} p(rk) \qquad (2) \end{aligned}$$

Thus the deinterleaving algorithm will analyse the sample and attempt to extract the individual sequences. We now discuss two types of deinterleaving algorithm: the difference histogram and sequential search.

3 Conventional TOA deinterleaving algorithms

We now discuss the performance of the published TOA algorithms under high pulse densities and complex signal conditions and identify areas for improvement.

3.1 TOA difference histogram

A simple signal deinterleaver is the TOA difference histogram [3]. Each TOA is subtracted from every subsequent TOA and a count accumulated at each TOA difference. Applying this to a stable PRI sequence will result in a count at integral multiples of the PRI. The number of computations required for a sample of E events is of the order of

$$\sum_{i=0}^{E} i \approx \frac{E^2}{2}$$

where $E \gg 1$ and $E \ll N$.

Though a count is given at the correct PRI, when several signals are present counts also occur at multiples, sums and differences of all the PRIs, thus giving ambiguous results. Fig. 3 shows the TOA histogram of a simple signal. This could be two interleaved stable sequences with PRIs of I. The histogram shows many nonzero counts at multiples of I and at sums and differences of multiples of I and x.

The difference histogram is an autocorrelation of the sample as can be seen by applying a delay of d SI to the sample and correlating:

$$y(d) = \sum_{r=0}^{N} p(rk)p\{(r - d)k\} \qquad (3)$$

Thus for each delay d, i.e. each PRI entry in the histogram, $y(d)$ contains a count equal to the number of solutions of the eqn.

$$q_i + am_i = q_j + bm_j + d \quad i, j = 1 \text{ to } x \qquad (4)$$

where a, b, d are positive integers.

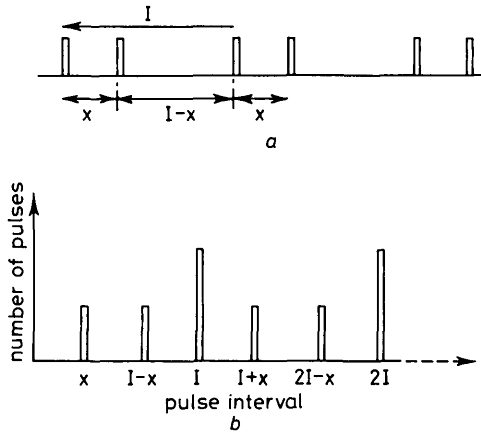

Fig. 3 *TOA histogram*
a Signal
b Histogram of all the differences

Thus the full count of events in the ith stable sequence occurs at the appropriate PRI and multiples of the PRI when

$$d = cm_i \quad y(d) > n_i - c \quad c = 1, 2, \ldots$$

The count can be greater than the calculated value due to additional events in the sample.

Each stable PRI sequence is therefore identified by the correct count at each multiple of the PRI. A threshold above which the sequence is said to be present must be defined. This must allow for missing pulses and interfering pulses. If the counts at the PRI harmonics are less than the thresholds, then the PRI is only a subinterval and does not form a sequence.

However, when several signals, missed pulses and erroneous pulses occur the count will vary and the decision thresholds will be critical. Either signals will not be detected or the false identification rate will be unacceptable. The histogram does not use sequential information, it simply counts the number of event pairs separated by a given PRI. Neither does the histogram identify the sequences. However, as it is based on subtractions it offers fast processing, and this technique is used in modern ESM receivers [4].

3.2 Sequence search algorithms

Sequence search provides identification of sequences and gives greater accuracy and reliability than the difference histogram, but at the expense of processing speed. Sequences of identical intervals are to be extracted from the sample, however the PRIs and phases are unknowns.

As a starting point we can postulate sequences with all of the possible PRIs at all phases and attempt to match these with the sample. The signal postulated is α_p:

$$\alpha_p = \sum_{r=0}^{N} f_p(rk)$$

This can be correlated with the sample to give a count

$$y = \sum_{r=0}^{N} [f_p(rk)p(rk)] \qquad (5)$$

This yields a value of y equal to the number of solutions to

$$q_p + am_p = q_i + bm_i \quad i = 1, \ldots, x \qquad (6)$$

Thus a maximum count of n_p occurs when the entire sequence is matched in the sample. For a PRI of m SI there are m possible phases and the number of TOAs to be correlated is N divided by m. Assuming the PRI of signals can be from 1 to N, then to search for all possible sequences would require of the order of

$$N \times \frac{N}{m} \times m = N^2 \quad \text{computations}$$

As N is much greater than E, a far greater number of computations are required than for the histogram method. Two points must be noted. First, it can be seen from eqn. 6 that multiples of PRIs in the sample will be extracted unless the smallest intervals are examined first. Secondly, signals will not have PRIs that are exact multiples of the sample interval. For example a PRI of $(m + 0.5)$ SI would be measured as successive TOAs separated by m, $(m + 1)$, m, ... due to the quantisation. This sequence rapidly diverges from a sequence with PRI of either m or $(m + 1)$ SI. Thus noninteger PRIs must also be examined and a PRI variation allowed for.

Now, as the actual PRIs and phases in the sample are a small subset of the possible values, the search should be limited to these.

This can be achieved by selecting a pair of adjacent events from the sample and projecting the TOA difference. Sequence search algorithms of this type have been proposed by Davies and Holland [5] and Campbell and Saperstein [6]. An event is selected and the interval with the adjacent event is projected. The postulated sequence is compared against the sample, with a PRI tolerance on each TOA. If the match is insufficient, the interval between the event and every subsequent event is projected and compared. This process is then repeated, starting from different events. When a sequence is found the events are removed, thereby simplifying further processing. These algorithms are efficient for only a small number of PRIs with high-quality data. If a sample contained E random events, the number of computations to determine this condition would be of the order

$$\sum_{i=0}^{E} i^2 \approx \frac{E^3}{3}$$

Furthermore, the extrapolation of a single PRI may rapidly diverge from the actual sequence, as described above. This will prevent detection even if a large PRI tolerance is used. Fig. 4 shows a pair of events that have

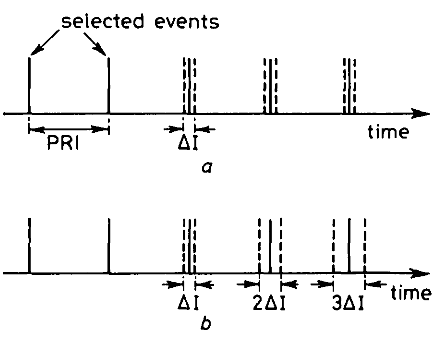

Fig. 4 *Divergence of projected sequence*
ΔI = PRI tolerance
a Postulated sequence
b Position of actual sequence

been selected and projected forward to form the postulated sequence, with a constant tolerance on the position of each event to allow for PRI variation. However, the actual sequence projected from the pair could fall within the increasing PRI bounds shown.

In dense environments a significant proportion of measurement errors and missing pulses will occur. This type of algorithm is prone to extracting multiples of PRIs. Complex PRI signals must also be considered, i.e. multilevel stagger and jittered signals. These will cause a large number of unsuccessful sequence searches and therefore a more robust algorithm is required.

4 The two-pass weighted-search algorithm

A new algorithm is presented which combines histogram techniques with sequence search techniques to obtain optimum deinterleaving.

4.1 The cumulative difference histogram (CDIF)

It is proposed to use the histogram as the first step of a TOA deinterleaving algorithm, providing a rapid indication of the probable PRIs in the sample. To minimise the false indications, the smallest intervals must be examined first.

Initially, a histogram is formed of TOA differences only between adjacent events. This is the first difference. As seen in Fig. 5*a* the resultant histogram is clearer than

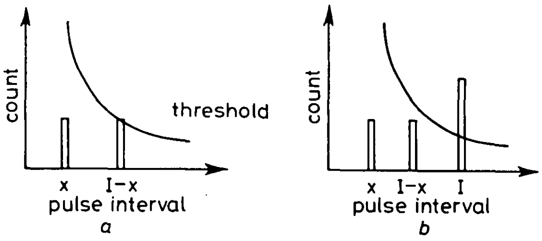

Fig. 5 *CDIF TOA histograms*
a First difference
b First + second difference

the all-differences histogram shown previously in Fig. 3. The count at each interval, and at double the interval, is compared to a threshold. If both counts exceed the threshold then a sequence search is performed at that PRI. If a sequence is not identified, the second difference, i.e. the TOA difference between each event and the next but one event, is calculated and the count is accumulated (Fig. 5*b*). The difference level is increased until detection occurs, or until a particular difference level is reached. By requiring the second harmonic to be present searches are limited to cases where sequences of three events occur, rather than only pairs. If higher harmonics were used then the search would be limited to longer sequences; however the number of difference levels required is multiplied, thus reducing the efficiency.

When the sequence search identifies a sequence, the pulses are removed from the sample and the histogram reset, thus simplifying subsequent processing. The smallest PRI radars will thus be removed quickly and only the optimum number of TOA differences calculated. For example, if five similar PRI signals are interleaved then detection would occur on or before the tenth difference level. A high-level flowchart is given in Fig. 6. Thus, the number of computations for x difference levels of histogramming is of the order

$$\sum_{i=E-x}^{E} i \approx [E^2 - (E - x)^2]/2$$

4.2 The weighted sequence search algorithm

Sequence search techniques can now be restricted to the PRIs that are repeated a sufficient number of times to form a sequence within the sample. The CDIF algorithm quickly determines these PRIs.

The conventional sequence search algorithms described above project from any selected pulse pair. When the sample contains several signals or a staggered

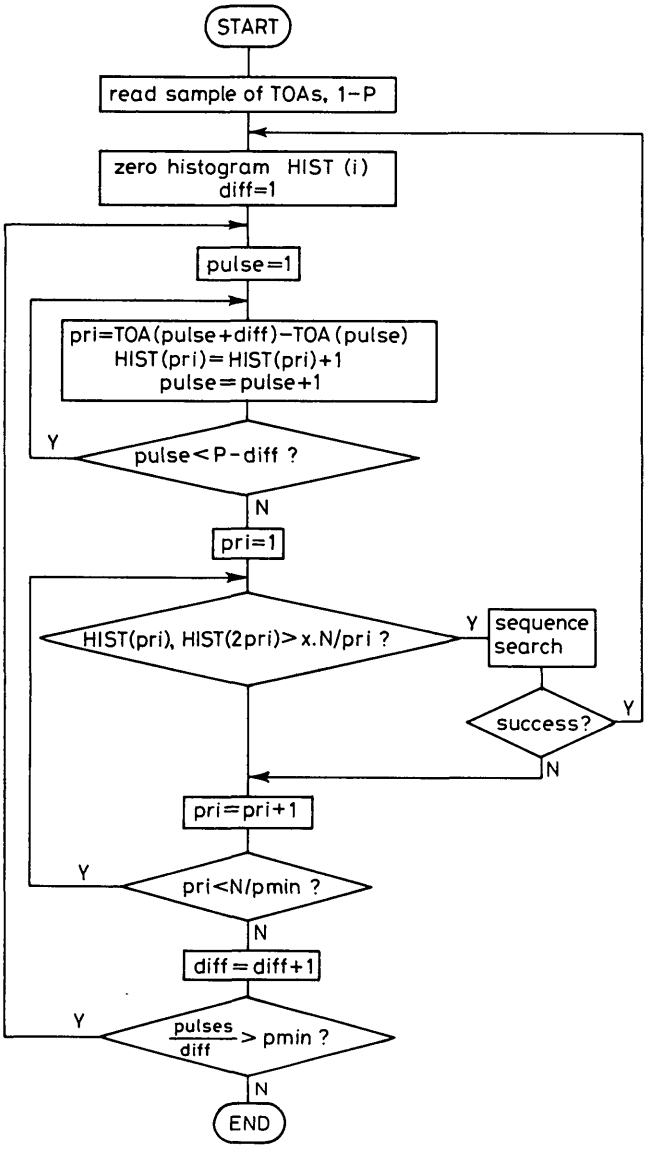

Fig. 6 *Cumulative difference histogram flowchart*

radar (Fig. 7 shows a two-level staggered signal), several unsequenced pairs exist, causing unsuccessful searches. As the possible PRIs have been determined by the CDIF

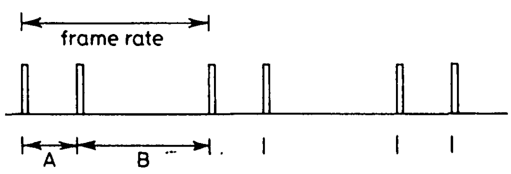

Fig. 7 *Staggered PRI signal*

algorithm, we can substantially reduce the number of iterations by selecting a larger sequence to prime the search. A long uncorrupted sequence may not be measured, therefore a sequence of three pulses has been chosen as the starting point for projection. This will reject random pulse pairs and sub-intervals of a staggered PRI and provide a more accurate PRI.

The fit of a sequence with missing events must be evaluated to determine decision thresholds and confidence levels. The conventional techniques simply counted the number of events matching the expected positions. Consider the two samples B and C in Fig. 8. Both have the same number of events matching the projected signal A. It can be seen that B is a more probable

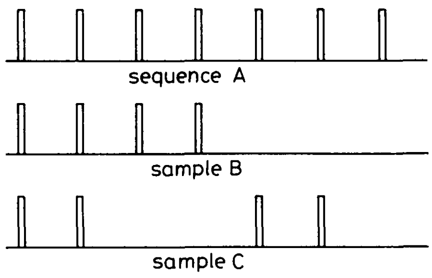

Fig. 8 *Pulse sequences*

fit to sequence A. Also note that sample C may be a better fit to a staggered signal.

We can use weighting schemes to enhance detection of uninterrupted sequences. A simple scheme is to count the number of events fitting the sequence and add the number of these events separated by the correct interval (Fig. 9). An alternative, more complex weighting function

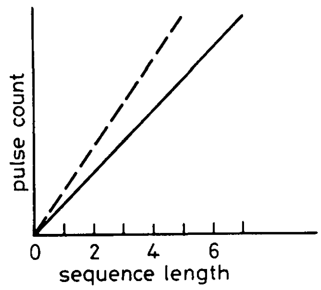

Fig. 9 *A count weighting function*

- - - - weighted
—— actual

is also suggested, which gives greater enhancement of sequences. This is based on the probability of a sequence of events occurring without being blocked. If the average pulse density and pulse width are monitored, then the multiple of the two (R) gives an estimate of the average probability of blocking a TOA measurement. Thus the probability of a sequence of n intervals with PRI m SI being measured is

$$p(n) \approx (1 - R)^n \quad m \gg 1 \quad R < 1 \tag{7}$$

Thus, for each unbroken sequence of events found in the sample, the reciprocal of this probability is added to the count. This automatically gives enhancement proportional to pulse density.

Fig. 10 gives a simplified representation of this algorithm. Histograms are formed and accumulated at each difference level until the count at a PRI (and double the PRI) exceeds a threshold. Then a sequence of three events at that PRI is searched for in the sample. If found, this sequence is projected to the next matching event, the PRI updated and projected from each event found, thus avoiding divergence and the need for large PRI bounds. If the weighted count is greater than the threshold, the events are extracted from the sample, otherwise subsequent three event primers are searched for and the process is repeated until no primer is found or detection occurs. If detection occurs the histogram is reset and recompiled from the remaining events, giving a clear histogram. A minimum of at least five pulses (pmin) are required to identify a sequence [3].

4.3 *Deinterleaving of agile signals*

In the presence of several signals, agile radar signals can be difficult to identify, therefore the signals that can be reliably extracted should be removed first.

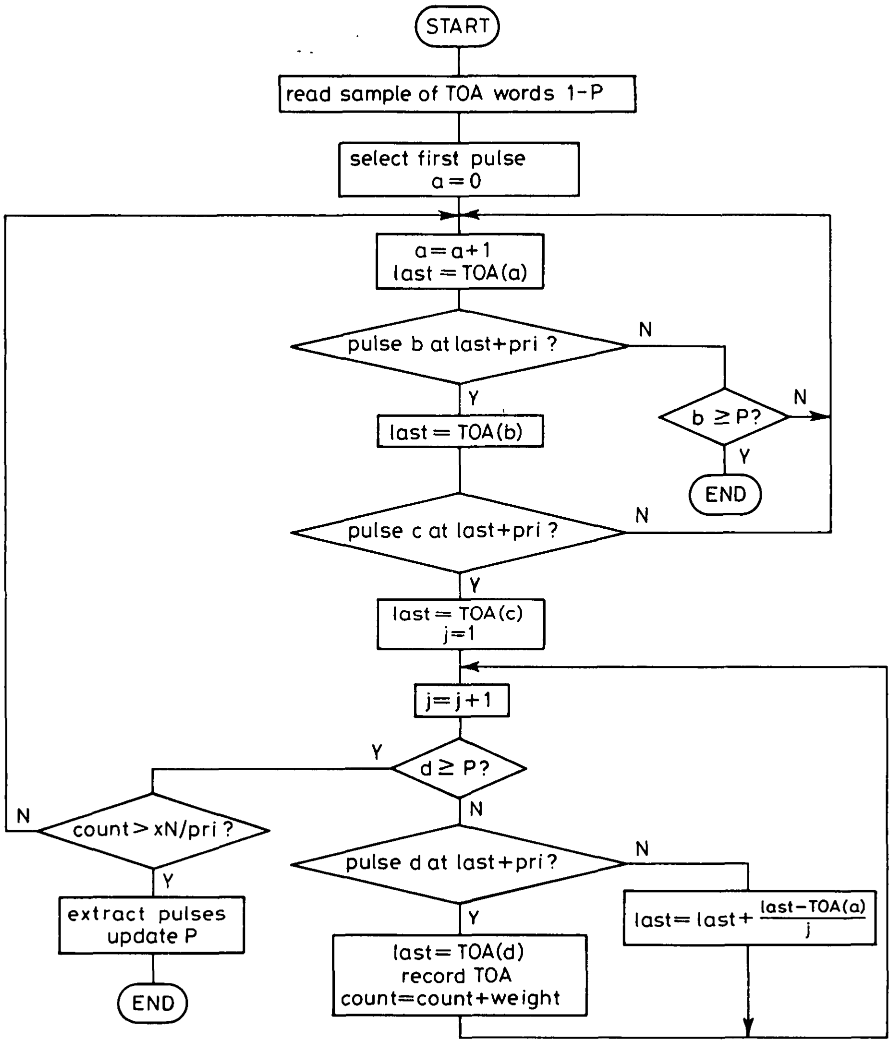

Fig. 10 *Weighted sequence search flowchart (simplified)*

If sufficient samples are taken so as to contain several frames of a staggered PRI signal, it will be identified as a number of constant PRI sequences equal to the number of stagger positions each with a PRI equal to the frame rate, and separated in time by the stagger intervals. Thus, after extraction of stable PRI sequences, the sequences with identical PRI are identified as staggered. Short emissions from scanning radars are extracted using the same algorithm, which is modified to extract bursts.

Jittered signals have a PRI with random or pseudorandom variation and are therefore more difficult to extract reliably. Jittered signals are therefore examined at the final stage. Groups of histogram entries are summed, with group sizes proportional to the PRI to encompass jitter. This indicates probable PRIs and jitter levels. The same sequence search algorithm is used with greater PRI tolerance. Following extraction of jittered signals, a residue of pulses may be left. This can be analysed by other algorithms and accumulated to extract longer PRI signals.

5 Implementation of the TOA deinterleaver

The algorithm that has been presented predominantly performs integer additions or subtractions and can be executed efficiently on a high-speed microprocessor. An optimum sample size is chosen to provide sufficient pulses for reliable identification whilst minimising the processing speed. Learning can be implemented by immediately applying the sequence search algorithm on previously identified PRIs before histogramming, thus increasing the efficiency of the algorithm during a scenario.

This algorithm has been coded in a high-level language and has successfully deinterleaved several test samples. Table 1 shows the comparison between three

Table 1 : TOA sorting algorithm performance

Test	Algorithms		
	Difference histogram	Interval extrapolate	CDIF + sequence
Test 1 (5 signals)			
instructions	14000	14000	23000
false reports	17	4	0
correct reports	2	3	5
Test 2 (6 signals)			
instructions	14000	22000	18000
false reports	28	2	0
correct reports	6	4	6
Test 3 (2 signals)			
instructions	3000	7000	3500
false reports	0	2	0
correct reports	1	0	2

Test 1: 5 signals; 112 events; stable PRI 11, 13, 17, 19, 23 SI
Test 2: 6 signals; 111 events; stable PRI 13.6, 23, 23, 23, 23, 23 SI (5-level stagger)
Test 3: 2 signals; 48 events; stable PRI 13.6; jittered PRI 23 SI

algorithms implemented by the author: the two-pass search algorithm, the conventional difference histogram, and an algorithm based on conventional extrapolation of event pairs. This indicates the relative performance of each of the techniques. The approximate number of low-level microprocessor instructions required to analyse the three test samples and the number of false and correct reports are given. The histogram produces a large number of incorrect reports. The interval extrapolation produces less false reports (these are harmonics of the test PRIs), however a proportion of the signals are not identified. The proposed 'sequence' algorithm generates no false reports and gives full detection on the identical test data.

This TOA deinterleaver has been incorporated into an ESM receiver simulation combined with an adaptive clustering process [1] (this does not include merge and analysis routines). This has demonstrated the capability for real-time deinterleaving on corrupted data. Of course, the processing rate is dependent on the data. For an example scenario, with a pulse density of 250 000 pulses per second, a 33% blocking rate and 11 radar signals including agility, processing of the order of 25 MIPS steady state was required for successful TOA deinterleaving without optimisation of the algorithm. There is scope for optimisation, particularly in the histogramming process. Detailed evaluation of a practical implementation of this algorithm is required with actual scenario information.

6 Conclusions

New techniques for high-accuracy, real-time TOA deinterleaving have been presented which can perform in high-pulse-density radar environments with complex signal types. Two-pass deinterleaving combines accuracy and efficiency. Weighting of sequence enhances detection of corrupted signals and a three-pulse sequence primer provides rejection of interfering pulses and increases efficiency. Thus a robust TOA deinterleaver can be implemented to give a substantial increase in performance over conventional published techniques.

7 Acknowledgments

I would like to thank the Electrical Engineering Department of Leeds University and Filtronic Components for the opportunity to undertake this work. I would also like to thank Prof. J.D. Rhodes for his support.

8 References

1 MARDIA, H.K.: 'Digital signal processing for radar recognition in dense radar environments'. PhD thesis, Electrical Eng. Dept., Leeds University, 1988, pp. 32–48, 108–115
2 MARDIA, H.K.: 'Adaptive clustering for ESM'. IEE Colloquium on Signal processing for ESM systems', 1988/62, pp. 5/1–5/4
3 WILEY, R. G.: 'Electronic intelligence: the analysis of radar signals' (Artech House, 1982), pp. 156–160, pp. 212–214
4 ROGERS, J.A.V.: 'ESM processor system for high pulse density radar environments', *IEE Proc. F, Commun. Radar & Signal Process.*, 1985, **132**, (7), pp. 621–624
5 DAVIES, C.L., and HOLLAND, P.: 'Automatic processing for ESM', *IEE Proc. F, Commun. Radar & Signal Process.*, 1982, **129**, (3), pp. 164–171
6 CAMPBELL, J.W., and SAPERSTEIN, S.: 'Signal recognition in complex radar environments'. Watkins-Johnson Tech. Notes, November/December 1976, 3, (6)