

DATA ADAPTIVE CONSTANT FALSE ALARM RATE NORMALIZER DESIGN FOR ACTIVE SONAR AND RADAR

D. W. Tufts

Department Of Electrical Engineering
University of Rhode Island
Kingston, RI 02881
USA
tufts@ele.uri.edu

E. C. Real

Sanders, A Lockheed Martin Company
P.O. Box 868
Nashua, NH 03061-0868
USA
edward.c.real@lmco.com

ABSTRACT

We present a method for estimating threshold values for signal detection and classification systems in which a prescribed value of false alarm probability is needed. The threshold values are determined directly from observed test statistic data without knowledge of the probability distribution of the data. Our method uses the concept of tolerance intervals from nonparametric statistics.

1. INTRODUCTION

The ability to accurately estimate constant false alarm rate (CFAR) thresholds in an environment in which the distribution of the noise or clutter is both unknown and possibly time varying is a critical requirement in many systems. Difficulty in dealing with these issues often leads to ad-hoc methods that are unsatisfying theoretically and can breakdown unexpectedly. To avoid these problems we propose a new method for estimating CFAR thresholds that is based on tolerance interval analysis [1]-[7]. Strict application of the original results of Wilks [2], Wald [3], Scheffé [4] and Tukey [4, 5, 6] requires statistically independent, real valued samples of the data. However, we argue that, for applications in signal processing, the number of independent samples, can often be replaced by an effective number of independent samples; which may be estimated from the data.

The concept of tolerance intervals has been inappropriately neglected in statistical signal processing. Its' value was pointed out to us by Roy Streit. A use of tolerance interval analysis in classification has been presented by Streit and Luginbuhl [8]. Their paper also contains a useful summary of the development of tolerance interval analysis.

We demonstrate the utility of our method with two simulated examples which are intended to approximate two different sonar or radar interference environments.

2. ESTIMATING THE SAMPLE SIZE

In this technique, the accuracy of the result is directly related to the number of independent samples used in the computation. We can estimate the number of samples required in the following way. Let n represent the number of real independent samples chosen from the available data. Let X denote this set of n independent samples. Let $X_{(r)}$ be the r -th order statistic of X , where $X_{(1)} < X_{(2)} <$

$\dots < X_{(n)}$. For continuous distributions the probability that all of the samples are less than ξ is [1]:

$$P(X_{(n)} < \xi) = F(\xi)^n \quad (1)$$

where $F(x)$ is the cumulative distribution function for the unknown distribution. Similar results can be derived for discrete distributions (e.g. [6, 7]); however in this paper we will limit ourselves to considering only those cases in which data is drawn from continuous distributions.

Let X be drawn from a noise or clutter source under consideration, and let $\alpha = 1 - F(\xi)$ be the desired false alarm rate.¹ The quantile ξ is the threshold we seek to achieve this false alarm rate. To insure the proper functioning of our algorithm, we want this quantile to lie within the range of our independent samples. That is we wish $X_{(1)} < \xi < X_{(n)}$. Therefore we choose n so that the probability given by equation (1) poses an acceptable level of risk.

For example, if the desired false alarm rate α is five percent, then for $n = 100$ independent real samples drawn from a continuous distribution the probability that all of the samples are less than ξ is $F(\xi)^n = 0.95^{100} = 0.0059$. If this probability constitutes an acceptable level of risk then we may be satisfied with 100 independent samples. If this is not the case then we can increase n until the risk is acceptable.

More directly, from equation (1) above, let $R = P(X_{(n)} < \xi) = F(\xi)^n$ be a specified level of risk that the true false alarm threshold is greater than the largest order statistic $X_{(n)}$. We then estimate the required number of samples as:

$$n = \lceil \log(R) / \log(1 - \alpha) \rceil \quad (2)$$

where $\lceil x \rceil$ indicates that x is to be rounded up to the next integer.

It is assumed above and throughout this paper that any estimate of n is also consistent with estimates of the time or space intervals over which the data can be assumed to be stationary. If these stationary intervals (in samples) are longer than n then the algorithms presented here will work well. If this is not the case, then we may decide to reduce the sample size n to the point that it becomes consistent with these intervals. If following this, the risk R becomes too high, then we may decide to suffer the effects of mixing data drawn from statistically different distributions which result from using a larger value of n , or we may choose another

¹Note that specifying α is equivalent to specifying $F(\xi)$ so that knowledge of $F(x)$ is unnecessary.

approach to estimating the false alarm rate threshold. However, we note that any other approach based on sampled data must also address these same trade-off issues.

3. ESTIMATING THE THRESHOLD

From [1] we know that for continuous distributions the probability that the quantile ξ lies between the order statistics $X_{(r)}$ and $X_{(v)}$ where $X_{(r)} < X_{(v)}$ is:

$$P(X_{(r)} < \xi < X_{(v)}) = \sum_{k=r}^{v-1} \binom{n}{k} F(\xi)^k (1 - F(\xi))^{n-k} \quad (3)$$

Using this general result we can compute the specific probability that the true threshold ξ lies between any two consecutive order statistics by letting $r = 1, 2, \dots, n-1$ and $v = r+1$. For example, if we select $n = 100$, $\alpha = 1 - F(\xi) = 0.05$ and assuming we have a continuous distribution we can use equation (3) to obtain the result shown in figure (1).

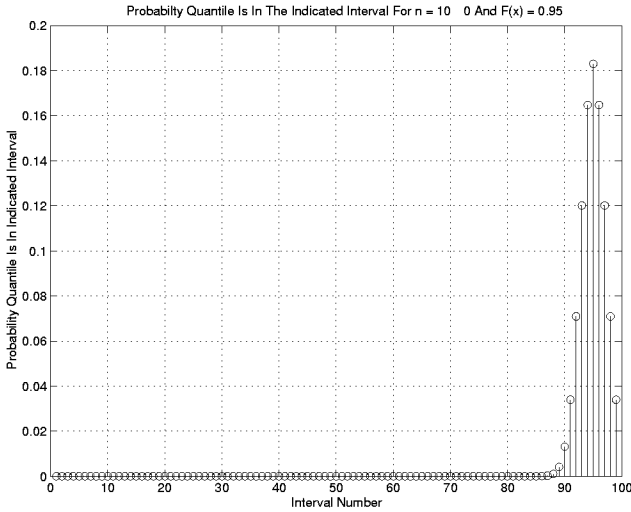


Figure 1: $P(X_{(r)} < \xi < X_{(v)})$ for $r = 1, 2, \dots, n-1$ and $v = r+1$ when $n = 100$ and $\alpha = 0.05$.

As indicated, the true, but unknown, threshold ξ is most likely in the interval between order statistics $X_{(95)}$ and $X_{(96)}$ with probability $P(X_{(95)} < \xi < X_{(96)}) = 0.1830$.

For this special case of computing the probabilities that the true threshold ξ lies between any two consecutive order statistics, we have a simple formula for determining the interval which has the largest probability [9]. This interval is computed as:

$$I = [(n+1)F(\xi)] = [(n+1)(1-\alpha)] \quad (4)$$

where $[x]$ is the integer portion of x . If $(n+1)(1-\alpha)$ is an integer then I identifies the second of two consecutive intervals which will have the largest (but equal) probability of containing ξ . That is, if $(n+1)F(\xi)$ is an integer then the intervals specified by $I-1$ and I will have equal probabilities and this probability will be larger than that of any other interval.

The steps required for estimating a threshold ξ may now be enumerated.

1. We make an initial determination of the number of independent noise or clutter samples required to estimate ξ for a given false alarm rate α as explained in section 2.
2. Given this estimate we select intervals defined by the order indices r and v and compute $P(X_{(r)} < \xi < X_{(v)})$ for every selected interval using equation (3). Note that if we choose the intervals defined by $r = 1, 2, \dots, n-1$ and $v = r+1$ as in the example above, we may then use equation (4) to determine, in one step, the interval which has the largest probability. If this strategy is used, then the quantity $(n+1)(1-\alpha)$ should be checked to determine if it is an integer and appropriate action should be taken in the following steps if it is.
3. Using either equation (3) or (4), we adopt a maximum likelihood point of view, and select the interval or intervals associated with the largest probability as being the most promising one to base our estimate of ξ on.
4. We collect n independent noise or clutter samples, order them, extract those ordered samples which constitute the selected interval and with them compute our estimate of ξ .

To illustrate, in the example given above we found that the maximum interval was bounded by $X_{(95)}$ and $X_{(96)}$. In this case we might estimate ξ by averaging $X_{(95)}$ and $X_{(96)}$ or by simply choosing one or the other of these order statistics as our estimate.

4. ESTIMATING A CFAR THRESHOLD

The method of estimating the false alarm rate threshold ξ outlined above assumes a stationary distribution. However, in many real world cases the distribution of the environmental noise or clutter is not only unknown, but non-stationary as well. Fortunately, the above technique is flexible enough to accommodate these cases as well. The steps in the algorithm are as follows.

1. We determine the number of independent samples and particular order statistics required to determine the desired threshold as outlined in section 3.
2. We estimate a sample interval (in time or space, depending on the application) such that samples separated by more than this amount can reasonably be assumed to be independent. For example, in typical time domain applications the CFAR threshold is applied after some filtering operation. In these cases one could estimate the equivalent noise bandwidth of the filter and down sample the filter output by the inverse of this bandwidth in samples.
3. At predetermined intervals of time or space n independent samples are collected, sorted, the order statistics are chosen, and a new threshold ξ is computed and applied to the input data stream.

An example of a sliding (lag) window version of this algorithm is show in figure (2).

Note that once a target signal is detected, the processing described above must be altered to account for the signal energy that is being added to the data stream. One approach is to simply hold the current threshold until the signal has passed. This approach assumes that the signal energy is reasonably constant over the duration of time the signal is present. However if the signal is present

for an extended period of time, and its energy is also changing over that time, we would like to be adaptive to that situation. In this case, one strategy is to construct a tolerance interval constant false *rejection* rate algorithm in a manner similar to the one described here and use it to determine when the signal has passed. In this case the threshold estimate would (most likely) be computed from the *smaller* order statistics and individual samples from the data stream would be tested to see if they fell *below* the estimated threshold.

Other applications, such as automatic gain control (AGC) problems, and other variations on the basic algorithm are easily imagined.

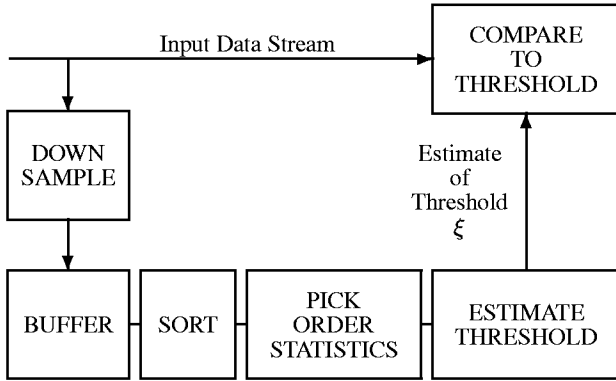


Fig 2. A Tolerance Interval CFAR Algorithm

5. AN EXAMPLE USING INDEPENDENT SAMPLES

In order to illustrate the performance of CFAR algorithms designed in this manner we select the algorithm shown in figure (2) as an example. The performance of this algorithm is demonstrated first in figure (3) by superimposing the computed time varying CFAR thresholds on top of the input data stream. For this example the input data are independent samples of simulated non-Gaussian clutter. The independence of the input samples eliminated the need for down sampling so this was not done. The design probability of false alarm was chosen to be five percent. A buffer length of 100 samples was used, and with no down sampling this became a sample by sample sliding window of that length. The order statistic intervals required in step (2) of the algorithm design sequence given in section 3 were chosen to be $r = 1, 2, \dots, 99$ and $v = r + 1$.

These selections result in the same quantile probability graph as that shown in figure (1). As noted earlier we see that the threshold ξ is most likely in the interval between order statistics $X_{(95)}$ and $X_{(96)}$ with probability $P(X_{(95)} < \xi < X_{(96)}) = 0.1830$ (continuous distribution assumption). For simplicity we choose the estimate of the threshold ξ to be the 96-th order statistic $X_{(96)}$. On average, a selection such as this should produce a slightly conservative estimate of ξ . That is one that produces a false alarm rate slightly less than the designed false alarm rate of five percent. The observed false alarm rate of the 10,000 samples of the clutter shown in figure (3) was 0.0508 (5.08 percent) in good agreement with the target design.

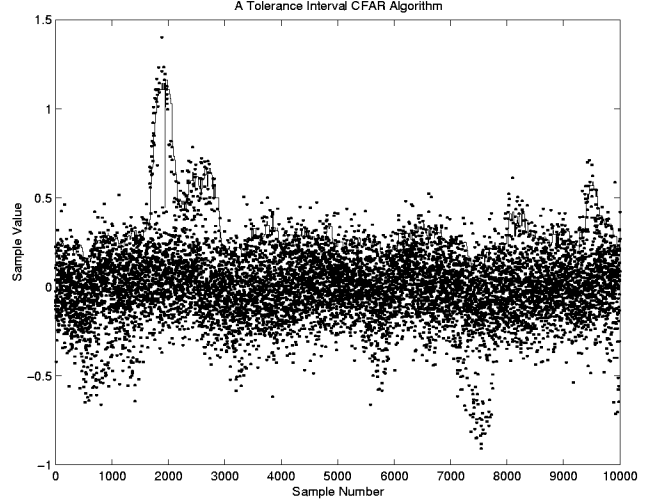


Figure 3: Typical performance of the tolerance interval CFAR algorithm of figure (2) in simulated clutter. The threshold value as a function of sample number is represented by a solid line while the data values are represented by dots.

6. AN EXAMPLE USING CORRELATED SAMPLES

In this section we attempt to demonstrate the utility of our approach when applied to a simulated sonar problem. In figure (4) we see a spectrogram of the input signal in Gaussian noise and non-Gaussian clutter. The signal is a two tone burst that begins at 5 seconds and has a duration of 0.125 seconds. The two component frequencies are at 100 and 300 Hz. The input signal to noise ratio is 0 dB. The sampling rate is 1000 Hz. For the purpose of demonstration only, we employ the MUSIC algorithm [10] as our detector. The subspace updates required by the MUSIC algorithm are performed by the FAST algorithm [11].

Figure (5) shows the adaptive threshold version of our algorithm applied to the output of the MUSIC algorithm. The false alarm value α that we designed for in this example is 0.001. The level of risk R was set to 0.1. From these values and equation (2) we compute the number of independent samples needed for each threshold update to be $n = 2302$. To be strictly true to our stated algorithm (section 4), we should down sample, or otherwise decorrelate, the output of the MUSIC algorithm in order to obtain these samples. However, in order to illustrate the effect of using correlated data with our algorithm, we apply it directly to the output of the MUSIC algorithm. We see from figure (5) that our measured false alarm rate in this case is 0.0008, slightly smaller than the designed for value of 0.001. Qualitatively, this is a common occurrence when processing correlated data with this algorithm; and is linked to the method by which the threshold is chosen.

Finally, we note that we have used a "split window" clutter buffer in this example. That is, we have introduced a processing delay (not shown) into the system so that the clutter buffer of figure (4) consisted of "past" and "future" clutter samples, relative to the test sample. We see this architecture reflected in the threshold plot in figure (5); i.e. the threshold rises in anticipation of upcoming target or clutter spikes in the data.

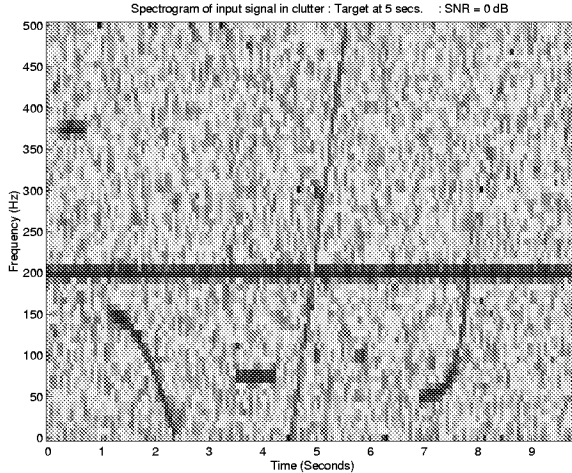


Figure 4: The input signal in clutter. Input SNR = 0 dB.

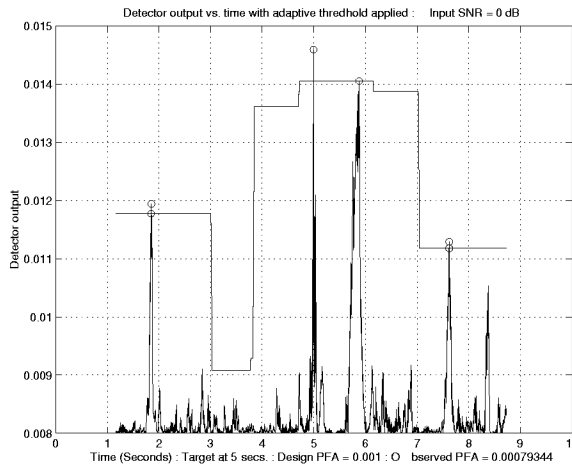


Figure 5: Adaptive threshold applied to output of MUSIC detector. Designed PFA = $1e-3$. Measured PFA = $7.93e-4$.

7. CONCLUSIONS

We have presented a simple, general and flexible method for computing adaptive and non-adaptive CFAR thresholds based on tolerance interval analysis. Implementation requires few or no arithmetic operations. The use of tolerance intervals eliminates the requirement that the distribution of the input data must be known. Further, the only requirements placed on the data are that the data be real valued and that the samples upon which the analysis is based be independent. Specific examples of the construction of an adaptive CFAR algorithm were given and its performance was demonstrated.

8. REFERENCES

- [1] John W. Pratt and Jean D. Gibbons, *Concepts of Nonparametric Theory*, Springer-Verlag, 1981.
- [2] S. S. Wilks, "Determination of Sample Sizes for Setting Tolerance Limits", *Annals of Mathematical Statistics*, Vol. 12, pp. 91-96, 1941.
- [3] Abraham Wald, "An Extension of Wilks' Method for Setting Tolerance Limits", *Annals of Mathematical Statistics*, Vol. 14, pp. 45-55, 1943.
- [4] H. Scheffé and J. W. Tukey, "Non-Parametric Estimation. I. Validation of Order Statistics", *Annals of Mathematical Statistics*, Vol. 16, pp. 187-192, 1945.
- [5] John W. Tukey, "Non-Parametric Estimation II. Statistically Equivalent Blocks and Tolerance Regions - The Continuous Case", *Annals of Mathematical Statistics*, Vol. 18, pp. 529-539, 1947.
- [6] John W. Tukey, "Non-Parametric Estimation, III. Statistically Equivalent Blocks and Multivariate Tolerance Regions - The Discontinuous Case", *Annals of Mathematical Statistics*, Vol. 19, pp. 30-39, 1948.
- [7] D. L. Hanson and D. B. Owen, "Distribution-Free Tolerance Limits : Elimination of the Requirement that Cumulative Distribution Functions be Continuous", *Technometrics*, Vol. 5, No. 4, pp. 518-522, November 1963.
- [8] Roy L. Streit and Tod E. Luginbuhl, "Maximum Likelihood Training of Probabilistic Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 5, No. 5, pp. 764-783, September 1994.
- [9] Athanasios Papoulis, *Probability, Random Variables, and Stochastic Processes - Second Edition*, McGraw-Hill, 1984.
- [10] Ralph Schmidt, "Multiple Emitter Location and Signal Parameter Estimation", *Proceedings of the RADCSpectrum Estimation Workshop*, pp. 243-258, 1979.
- [11] D. W. Tufts, E. C. Real and J. W. Cooley, "Fast Approximate Subspace Tracking (FAST)", *Proceedings of ICASSP97*, Vol. 1, pp. 547-550, April 21-24, 1997.