Pulse Extraction for Radar Emitter Location

Mark L. Fowler, Zhen Zhou, and Anupama Shivaprasad
Department of Electrical Engineering
State University of New York at Binghamton
Binghamton, NY 13902
Email: mfowler@binghamton.edu

Abstract **– Two related data compression methods for radar signals are described and analyzed. The methods use the singular value decomposition (SVD) of a data matrix containing one pulse in each row to exploit pulse-to-pulse redundancy. By using a rank-one approximation to the data matrix it is possible to achieve compression ratios typically of the order of several 10's and sometimes over 150:1 for typical radar types while maintaining accurate location; the amount of compression depends on the radar's parameters. Alternatively, by retaining a single singular vector as an extracted prototype pulse, it is possible to use it for matched filter processing, thus providing an improved method for doing noncoherent time-of-arrival (TOA) processing for a slight cost of a small amount of additional data to be sent.**

I. INTRODUCTION

A common way to locate electromagnetic emitters is to measure the time-difference-of-arrival (TDOA) and the frequency-difference-of-arrival (FDOA) between pairs of signals received at geographically separated sites [1]. For radar emitters there are (at least) two alternatives for measuring the TDOA/FDOA values. The coherent method measures the TDOA/FDOA by coherently cross-correlating the signal pairs [2], and the noncoherent method measures the time-of-arrival (TOA) of the pulses (and possibly frequency-of-arrival (FOA) also) at each platform and then combines the TOA (and possibly the FOA) measurements made at two platforms into TDOA (and possibly FDOA) estimates [8]. From a theoretical viewpoint, the coherent method has a clear advantage because it more completely utilizes the information embedded in the received signals. Furthermore, the noncoherent approach requires that the SNR at each platform be high enough that the pulses can be detected using simple thresholding of the leading edge because no matched filter is available. After pulse detection the TOA/FOA can be measured (TOA is usually measured using leading edge methods or pulse centroid methods, while various frequency estimation techniques are used to measure FOA). On the other hand, coherent methods exploit the time-bandwidth processing gain [2] to allow, in principle, operation at much lower SNRs at all platforms; although in practice at least one platform generally needs to be at an SNR high enough to detect the pulses to allow signal acquisition processing such as identifying the presence of a signal of interest.

However, coherent processing does have a serious drawback: signal samples received at one platform must be transmitted over a data link to another platform in order to perform cross correlation. Because these links are rarely completely allocated to the sole task of transferring data for location processing, the allocated link rate usually is insufficient to accomplish this in a timely manner – especially for the radar location case with its wide bandwidths. To mitigate this, various data compression approaches have been proposed [3] – [7], although they have been designed for the generic signal case and can't fully exploit the characteristics of radar signals. For the radar case, the fact that (at least) one platform will be operating at an SNR high enough to detect pulses can be exploited as a first step towards reducing the transferal by not sending the samples between detected pulses. However, even with this reduction the transferal time is still excessive given allocated rates for current and projected data links.

In this paper, rather than approaching the problem directly from the perspective of data compression, we will propose a way to extract a prototype pulse from the high-SNR platform and then send that pulse to the other platforms (possibly together with some side information), where it can be used for measuring the TDOA/FDOA values. As a result we get a significant amount of data compression, but viewing the scheme more as a prototype pulse extraction leads to some alternative viewpoints. We discuss two different ways to use the extracted prototype: one based on noncoherent TOA methods (which we will call "semi-coherent") and one based on coherent TDOA/FDOA methods. The semi-coherent method uses the extracted pulse as a pulse-matched filter allowing improved pulse detection and TOA measurement; this allows all but one platform to be at low SNR, although not as low as for coherent processing. On the other hand, the coherent method uses the prototype pulse together with some side information (a sequence of pulse phases, magnitudes, and times) to reconstruct a complete pulse train at the other platform that is suitable for cross correlating with that platform's locally received pulse train. We will show how to use the singular value decomposition (SVD) together with some pulse alignment processing to effectively extract the prototype pulse. We then will discuss how to exploit the prototype pulse in the semi-coherent and coherent approaches.

II. EXTRACTIING THE PULSE

The proposed extraction approach is based on the fact that a radar emits a train of similar pulses. Because modern radars can change modes we assume that preliminary subtrain-extraction processing has grouped the signal of interest into one or more subtrains, each having pulses from the same mode of operation – such processing is a standard part of any electronic warfare system (this processing also removes pulses from other emitters) [9]. We extract a prototype pulse for each subtrain identified at the high-SNR platform, although for simplicity of discussion we will assume here that there is a single subtrain. The pulses in a subtrain look very much alike – except that they may have random phase shifts if the radar is not pulse-to-pulse phase coherent. Subtrain extraction provides a series of similar pulses separated by their original pulse repetition interval (PRI); the separation is indicated in the figure in terms of number of samples N_1, N_2, etc., which will be nearly-equal integers. The subtrain pulses are then gated and the inter-pulse samples are removed and the numbers of samples removed between the pulses (N_1, N_2, etc.) are extracted as side information; the resulting series of pulses can be thought of as a pulse train having an artificially short PRI. (This gating can be viewed as preliminary compression – but note that all compression ratios stated in this paper do not include this gating-based compression.) Finally, the gated pulse train is processed to extract the prototype pulse and any other required side information, which is then sent over the data link along with the gating-extracted pulse separations.

The mathematical basis of the prototype extraction processing is the singular value decomposition (SVD) [10]. If we put the gated pulses of the baseband equivalent signal for the received pulse train into a matrix – with one pulse per row – we would have a rank one matrix if (i) there were no noise or propagation effects, (ii) the pulses were perfectly time aligned – i.e. perfect leading-edge detection & gating, and (iii) the radar's PRI were an integer multiple of the sampling interval T. All but the first of these causes of increased rank can be mitigated by performing time alignment on the pulses in the pulse matrix. The needed time alignment values can be determined by computing pulse-to-pulse cross correlations. The pulses are then time aligned using a fractional delay FIR filter [11] although we are currently investigating the use of DFT based alignment. By using this alignment step the resulting pulse matrix is closer to being a rank-one matrix than it was before the alignment.

Let p denote the number of pulses in the pulse subtrain and let n denote the number of samples per pulse kept after gating; then the total number of samples is pn. If we denote the $p \times n$ (aligned) pulse matrix by $\mathbf{P}$, its SVD is

$$\mathbf{P} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^H \ , \tag{1}$$

where r is the rank of $\mathbf{P}$, $\mathbf{u}_i$ is the i^{th} left singular vector, $\mathbf{v}_i^H$ is Hermitian transpose of the i^{th} right singular vector, and σ_i is the i^{th} singular value, ordered such that $\sigma_i \geq \sigma_{i+1}$. Each term in the sum in (1) is a rank-one matrix. If we truncate this sum to only $k < r$ terms we get the rank-k matrix

$$\mathbf{P}_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^H \tag{2}$$

that best approximates $\mathbf{P}$ in the sense that the sum of the squares of the elements of $\mathbf{P} - \mathbf{P}_k$ is smaller than for any other rank-k matrix. Note that in our case the matrix contains the pulses and therefore this approximation gives the smallest mean square error (MSE) between the original pulse train and the approximate pulse train formed by concatenating the de-aligned rows of $\mathbf{P}_k$. As we will see later, this minimum MSE property is the basis for using the SVD for both of the two viewpoints considered here.

In the perfect-signal scenario, where there is no noise and no pulse misalignment, $\mathbf{P}$ *is* rank one and $\sigma_i = 0$ for $i \geq 2$. However, when signal perturbations are present, $\mathbf{P}$ has higher rank, but still has a few dominant singular values. Therefore, to get maximum compression we strive to approximate $\mathbf{P}$ by a matrix having a low rank while having a small MSE – in fact we will approximate it with a rank-one matrix.

The effect of the time alignment is to concentrate the energy of the pulse matrix into the first singular value, producing a matrix that is closer to a rank one matrix. The effect of the noise on the singular values is uniformly spread across all the singular values – this is in fact a known result that is exploited by previous applications of the SVD to signal processing problems. Thus, when we truncate the SVD to k terms as in (2) we are throwing away all the noise that exists in the thrown away singular values, and – if we've done our job right – we have thrown away very little of the signal because it is mostly concentrated in the singular values that we keep. The effect of this is to ***increase*** the SNR of the reconstructed signal found by concatenating the rows of $\mathbf{P}_k$ into a pulse train; thus, not only do we compress the signal but we get an improvement in SNR rather than a degradation due to compression! This simultaneous compression and noise reduction will be demonstrated in the next section after further discussing the compression/decompression processing.

To extract a prototype pulse we consider the case where we truncate the SVD to a single term (k = 1) to get $\mathbf{P}_1$; we'll demonstrate later that the accuracy achieved with k = 1 is excellent. To specify $\mathbf{P}_1$ we need the $p \times 1$ vector $\mathbf{u}_1$ (i.e., the 1st *left* singular vector), the $n \times 1$ vector $\mathbf{v}_1$ (i.e., the 1st *right* singular vector), and the scalar σ_1 (i.e., the 1st singular value). Note, then, that $\mathbf{v}_1^H$ is the same size as a pulse (n samples). Thus, we can interpret vector $\mathbf{v}_1^H$ as a single prototype pulse that has been extracted from the original pulse train, which is a nice viewpoint given that the *radar's* receiver would process *its* received pulse train using a pulse template as a matched filter; this leads to what we call a semi-coherent approach. Alternatively, we can view the concatenation of the rows of $\mathbf{P}_1$ as forming an approximation to gated and aligned pulse train, which together with the alignment and gating side info can be used to create an approximation of the original pulse train. These two viewpoints will be explored in the next section.

III. Exploiting The Pulse

A. Semi-Coherent Method

For simplicity here we consider the TDOA-only case, where we assume that there is no relative motion between emitter and platforms. In the semi-coherent method we use the prototype pulse $p_{pp}(k)$ as a matched filter to detect the pulses at the low SNR platform and to measure the TOA values for each pulse in the received pulse stream. The prototype pulse is also used to measure the TOA values at the platform where it was extracted. Then the corresponding TOA values from the two platforms are subtracted to form a sequence of TDOA values that are then averaged to give the desired TDOA estimate.

To do this processing we need to extract a prototype pulse from the subtrain received at the high SNR platform. Simply choosing one of the detected pulses as the prototype pulse *is* possible, although some rules would have to be used to select the best candidate – that would likely be difficult to do in the presence of fading, emitter scanning, and multipath. However, we'll see that the SVD method gives a mathematically-based rule to generate a prototype pulse by exploiting all the data in the pulse subtrain. We seek a prototype pulse that is highly correlated (positive or negative – since we check the magnitude of the matched filter output because we are not assured that the emitter is pulse-to-pulse coherent) with all the pulses in the received pulse subtrain. One criteria that could be used to find such a pulse is to seek a prototype pulse that maximizes the sum of the magnitude squares of the correlations. In other words, we seek $p_{pp}(k)$ such that if $p_i(k)$ are the pulses in the received pulse subtrain, then we maximize

$$C = \sum_i |< \mathbf{p}_i, \mathbf{p}_{pp} >|^2 , \tag{3}$$

where we have used vectors to represent the corresponding pulses. However, we now show that this is equivalent to choosing the prototype pulse vector such that it has unit norm and minimizes

$$E^2 = \sum_i \left\| \mathbf{p}_i - \alpha_i \mathbf{p}_{pp} \right\|^2 \tag{4}$$

for appropriately chosen α_i. Minimizing this over the choice of $\mathbf{p}_{pp}$ and α_i can be done in two steps: find the minimizing α_i for a given $\mathbf{p}_{pp}$ and then find the minimizing unit-norm $\mathbf{p}_{pp}$. The minimizing α_i are $\alpha_i =< \mathbf{p}_i, \mathbf{p}_{pp} >^*$, for which we get

$$E^2 = \sum_i \left\| \mathbf{p}_i \right\|^2 - \sum_i \left| < \mathbf{p}_i, \mathbf{p}_{pp} > \right|^2 , \tag{5}$$

which is minimized when C in (3) is maximized. However, recalling that the SVD approximate matrix minimizes the MSE between the elements of the matrix and the approximation, from (4) it is clear that E^2 is minimized when we choose $\mathbf{p}_{pp} = \mathbf{v}_1^H$ and $\boldsymbol{\alpha} = \sigma_1 \mathbf{u}_1$ where α_i is the i^{th} element of $\boldsymbol{\alpha}$. Thus, we see that extracting a prototype pulse using the SVD-based method described above satisfies our criteria of maximizing (3).

In the semi-coherent method, the side information needed in addition to the prototype pulse is the TOA measurements made at the high SNR platform using the pulse-matched filter. Of course, this side information is not truly side information in this case because they have to be sent for the noncoherent method, too. After the prototype pulse is received at the other platform it is used as a matched filter to detect subtrain pulses from the received signal and to measure their TOA values, which are then combined with the TOA values sent from the extracting platform.

The major advantage of doing this is that unlike noncoherent TOA-based processing, this method uses matched filter processing rather than leading-edge processing to measure the TOA. This enables operation at a lower SNR at the nonextracting platform than is possible with conventional noncoherent methods. In addition it gives a compression ratio on the order of

$$CR_{semi} = \frac{np}{n} = p \tag{6}$$

if we ignore any possibility of further coding the samples of the prototype pulse. This compares the amount of data needed for the coherent method with no compression to the amount needed for the proposed semi-coherent method. Thus, we see that the amount of compression is the number of pulses to be used in the processing. Because the number of pulses needed to achieve sufficient TDOA accuracy can range between several tens of pulses to several thousands of pulses (depending on the radar type as well as other system considerations), this method can give extremely large compression ratios.

B. Coherent Method

In the coherent method we seek a reconstructed pulse *train* that would minimize the MSE between it and the original pulse train before it was compressed. For a given compression ratio (e.g., for a given number of terms retained in (2)) this clearly is the pulse train formed from the de-aligned rows of the approximating pulse matrix $\mathbf{P}_k$, due to the SVD minimizing MSE. For $k = 1$, this is equivalent to minimizing E^2 in (4). Thus, we see that the semi-coherent method and the coherent method have the same minimization requirement and it is met through finding the SVD-based rank-one approximate matrix $\mathbf{P}_1$.

In the coherent method, in addition to the prototype pulse we need side information consisting of the values of the left-singular vector $\mathbf{u}_1$, the fractional time alignments, and the number of samples between adjacent pulses that were removed by gating. The left-singular vector $\mathbf{u}_1$ is used to reassemble the truncated SVD form of the pulse matrix (up to the scaling factor of σ_1), after which the time alignment information is used to undo the time alignment and, finally, zeros are inserted in place of the gating-removed signal samples. Thus, for the cost of a modest amount of side information it is possible to reconstruct a MSE-minimizing pulse train suitable for coherent cross-correlation.

In particular, the approximating matrix $\mathbf{P}_1$ is formed from

$$\mathbf{P}_1 = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^H , \tag{7}$$

from which it is clear that each row in $\mathbf{P}_1$ is a complex-valued scalar multiple of $\mathbf{v}_1^T$, where the complex scalar for the i^{th}

row is the i^{th} element in $\mathbf{u}_1$ times σ_1; it is also clear that σ_1 does nothing more than amplitude scale the entire reconstructed pulse train and can therefore be omitted. Thus, we can change (7) to

$$\tilde{\mathbf{P}}_1 = \mathbf{u}_1 \mathbf{v}_1^H , \tag{8}$$

from which we see that $\mathbf{u}_1$ holds the reconstruction magnitudes and phases. Finally, the rows of $\tilde{\mathbf{P}}_1$ have to be time shifted to undo the alignment processing time shifts, after which the results are assembled into a pulse train (with zeros inserted between pulses to undo the effect of pulse gating) that is cross-correlated with the pulse train received at the other platform. Thus, the information that is needed to reconstruct the signal is:

1. The $n\times 1$ right singular vector (RSV) $\mathbf{v}_1$ (i.e., the prototype pulse)
2. The $p\times 1$ left singular vector (LSV) $\mathbf{u}_1$ (i.e., the reconstruction magnitudes and phases)
3. The p-1 time shifts
4. The p-1 numbers of inserted zeros (N_1, N_2, … $N_{p\text{-}1}$)

Using this data at the other platform the reconstructed pulse train can be formed and then cross-correlated with the signal data received locally at that platform to estimate the TDOA/FDOA.

How much compression can we get from this scheme? We first consider the case where a single pulse is put into each row of **P**, but we will see that it is often better to put multiple pulses per row. Thus we have that **P** is $p\times n$, where p is the number of pulses (i.e. rows) in the pulse matrix and n is the number of samples per pulse. Thus, if no compression is used there are np samples to be sent. To send $\mathbf{P}_k$ in (2) we only need to send: (i) k singular values, which requires k values, (ii) k left singular vectors each having p elements, which requires kp values, and (iii) k right singular vectors each having n elements, which requires kn values. The total number of values needed to specify the reduced-rank SVD approximate $\mathbf{P}_k$ is $k(1+p+n)$ values; if k = 1 we don't need the singular value so this becomes $p+n$. Assuming that we use the same number of bits for each element as we did for the signal samples, the compression ratio is

$$CR_{coho} = \frac{np}{k(1+p+n)} , \tag{9}$$

where k is the # of singular vectors retained, p is the number of pulses to be processed, and n is the number of samples per pulse. Results on how to code the SVD extracted information is given in [12] as well as results that show for the coherent approach that the CR is maximized when the pulse matrix is made square. Thus, it may be desirable in many cases to put more than one pulse per row when using the coherent approach. When that is done the compression ratio for the coherent method becomes

$$CR_{coho} = \frac{np}{k(1+2\sqrt{np})} \approx \frac{\sqrt{np}}{2k} , \tag{10}$$

An additional advantage of the coherent approach is that the SVD's noise reducing property provides a reconstructed signal that is an improved version of the original rather than a compression-degraded version. Thus, a large compression ratio is achieved without the usual degradation associated with lossy compression methods. Furthermore, the coherent method allows the low-SNR platforms to operate at lower SNR values than for the semi-coherent method because the coherent processing provides the maximum time-bandwidth gain.

C. Performance Results

Monte Carlo simulations were performed to demonstrate the capability of the semi-coherent processing method. Two variations of the semi-coherent method were simulated to provide insight into the performance: (i) using a noise-free prototype and (ii) using the SVD-extracted prototype. Obviously the first variation is not possible in practice but is included here solely for comparison purposes. The simulation results are shown in Figure 1 for the case of 102 pulses of 40 samples each. The signal-to-noise ratio for the signal being compressed (denoted as DNR) was taken as 20 dB, the signal-to-noise ratio for the signal not being compressed (denoted SNR) varied from –2 dB to 30 dB; 100 Monte Carlo simulation runs were performed. The figure shows results for the semi-coherent method using (i) the SVD-prototype, (ii) the noise-free (NF) prototype taken directly from the noise-free pulse train, and (iii) the coherent cross-correlation method without compression (for reference). From the figure it is seen that as long as SNR ≥ 0 dB (for this case) there is no difference between the three methods.

To see why, consider the NF-prototype case where n is the number of samples in a pulse. When the NF-prototype is cross-correlated with pulse train #1 to estimate the i^{th} TOA, the variance is given by [1]

$$\sigma^2_{TOA_1,i} = \frac{1}{(2\pi\beta_{rms})^2 2nSNR} , \tag{11}$$

where β_{rms} is the rms (or Gabor) bandwidth of the signal. At the other platform we can write $DNR = \alpha SNR$ for some α, so the corresponding TOA estimate on the other pulse train (#2) has variance

$$\sigma^2_{TOA_2,i} = \frac{1}{(2\pi\beta_{rms})^2 2n\alpha SNR} . \tag{12}$$

The resulting TDOA estimate (assuming that the two TOA's are uncorrelated, which is only an approximation) has variance given by

$$\sigma^2_{TDOA,i} = \sigma^2_{TOA_1,i} + \sigma^2_{TOA_2,i} = (1 + 1/\alpha)\sigma^2_{TOA_1,i} \tag{13}$$

After averaging these we get that

$$\sigma^2_{TDOA,semi} = \sum_{i=1}^{p} \sigma^2_{TDOA,i}$$

$$= \frac{1}{(2\pi\beta_{rms})^2 2npSNR/(1+1/\alpha)} \quad (14)$$

$$= \frac{1}{(2\pi\beta_{rms})^2 2np(\frac{1}{SNR}+\frac{1}{DNR})}$$

On the other hand, if we compute the TDOA using the coherent method (without compression) the variance is given by [1]

$$\sigma^2_{TDOA,coho} = \frac{1}{(2\pi\beta_{rms})^2 2np(\frac{1}{SNR}+\frac{1}{DNR}+\frac{1}{SNR\cdot DNR})} \quad (15)$$

Recall that *DNR* must be high enough to detect the pulses prior to gating, say $DNR \geq 10$ dB, for which (15) can be shown to be approximately equal to the variance in (14).

However, there is an additional constraint that must be met for (14) and (15) to be valid: the cross-correlation output SNR must exceed about 15 dB. For semi-coherent processing, (14) is valid if the output SNR in (11) must exceed 15 dB: this requires that $nSNR \geq 15$ dB. However, for coherent processing, (15) is valid if

$$np(\frac{1}{SNR}+\frac{1}{DNR}+\frac{1}{SNR\cdot DNR}) \geq 15\,\text{dB} \quad (16)$$

For the SNR/DNR scenarios here this reduces to $npSNR \geq 15$ dB which shows that the coherent method can work at an SNR that is p times lower than for semi-coherent. In other words, the coherent method has a processing gain of np whereas the semi-coherent method has a processing gain of only n. For the example given in Figure 1 (n = 40, p = 102) the semi-coherent method should break down when SNR < –1 dB while the coherent method doesn't break down until SNR < –21 dB. This explains why the semi-coherent simulation result deviates from the coherent result at SNR = –2 dB as shown in Figure 1.

To give a rough idea of how the accuracy of noncoherent processing compares we use [8]

$$\sigma^2_{TDOA,nonco} = \frac{t_R^2}{SNR}, \quad (17)$$

where t_R is the rise time of the pulse. For the simulation in Figure 1 the rise time was about 2 μs, which at an SNR of 15 dB gives a TDOA rms error of about 350 ns, which is extremely large compared to the results obtained using the semi-coherent and coherent methods.

Monte Carlo simulations were also performed to demonstrate the capability of the coherent processing method. Due to space limitations we show only the TDOA results; the FDOA results are similar in nature. The TDOA accuracy results are shown in Figure 2, where it is seen that using the SVD method actually improves the accuracy despite the fact that it requires much less data transferal; at moderate SNR values the improvement is on the order of a 3 dB improvement in effective SNR. The case considered here had p = 50 pulses and n = 43 samples/pulse giving a compression ratio of 23:1.

Table 1 gives some idea of the large compression ratios achievable for typical scenarios. Four emitter scenarios are listed with typical pulse widths (PW), pulse repetition intervals (PRI), bandwidths (BW), number of complex samples per pulse, and the number of pulses that would typically be processed in order to get desirable accuracies. In addition, the resulting compression ratios (after gating) are computed using the equations given above. From these results it is clear that these methods can give very large compression ratios. Furthermore, if SNR is high enough then the semi-coherent and coherent methods have the same TDOA accuracy; however, the coherent method can operate at lower SNR values due to its higher time-bandwidth gain.

REFERENCES

[1] P. C. Chestnut, "Emitter location accuracy using TDOA and differential doppler ," *IEEE Trans. Aero. and Electr. Systems*, vol. AES-18, pp. 214-218, March 1982.

[2] S. Stein, "Algorithms for ambiguity function processing," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-29, pp. 588 - 599, June 1981.

[3] M. L. Fowler, "Data compression for TDOA/DD-based location system," US Patent #5,991,454 issued Nov. 23, 1999, Lockheed Martin Federal Systems.

[4] G. Desjardins, "TDOA/FDOA technique for locating a transmitter," US Patent #5,570,099 issued Oct. 29, 1996, Lockheed Martin Federal Systems.

[5] M. L. Fowler, "Coarse quantization for data compression in coherent location systems," *IEEE Trans. Aero. and Electr. Systems*, vol. 36, no. 4, pp. 1269 – 1278, Oct. 2000.

[6] M. L. Fowler, "Data compression for emitter location systems," Conference on Information Sciences and Systems, Princeton University, March 15-17, 2000, pp. WA7b-14 – WA7b-19.

[7] M. L. Fowler, "Decimation vs. quantization for data compression in TDOA systems," in *Mathematics and Applications of Data/Image Coding, Compression, and Encryption III*, Mark S. Schmalz, Editor, Proceedings of SPIE Vol. 4122, pp. 56 – 67, San Diego, CA, July 30 – August 4, 2000.

[8] R. G. Wiley, *Electronic Intelligence: The Interception of Radar Signals*. Artech House, 1985.

[9] R. G. Wiley, *Electronic Intelligence: The Analysis of Radar Signals*, 2nd Edition. Artech House, 1993.

[10] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, 2000.

[11] T. I. Laakso, V. Valimaki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay," *IEEE Sig. Proc. Mag.*, vol. 13, no. 1, pp. 30 - 60, January 1996.

[12] M. L. Fowler and Z. Zhou, "Data compression via pulse-to-pulse redundancy for radar emitter location," submitted to *Mathematics and Applications of Data/Image Coding, Compression, and Encryption IV*, SPIE Annual Symposium, August 1,2001.

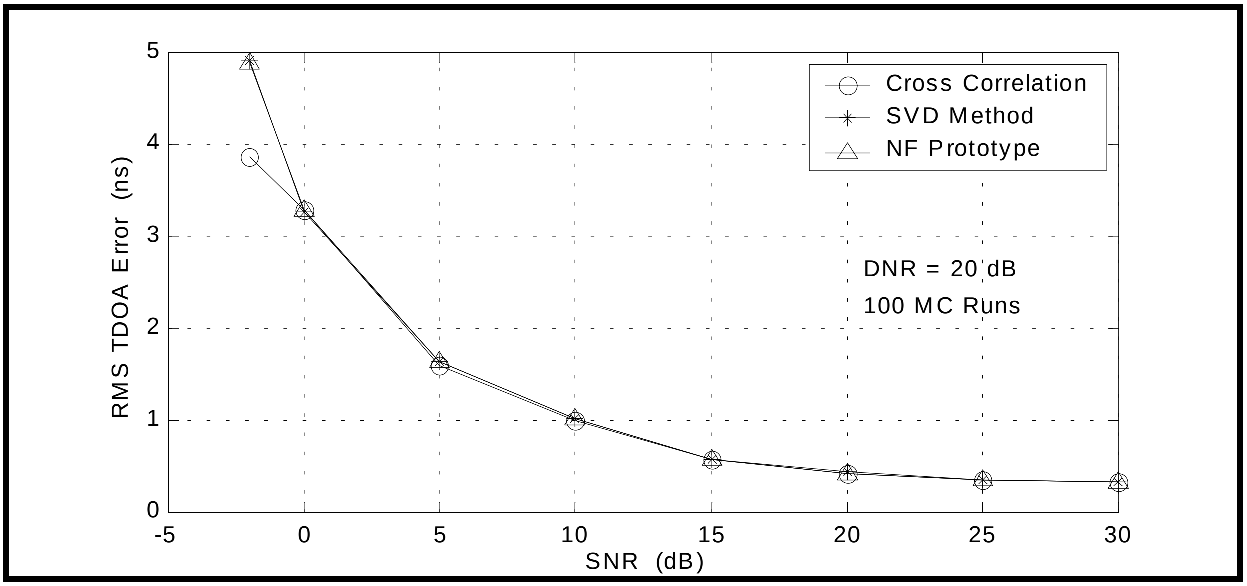

Figure 1

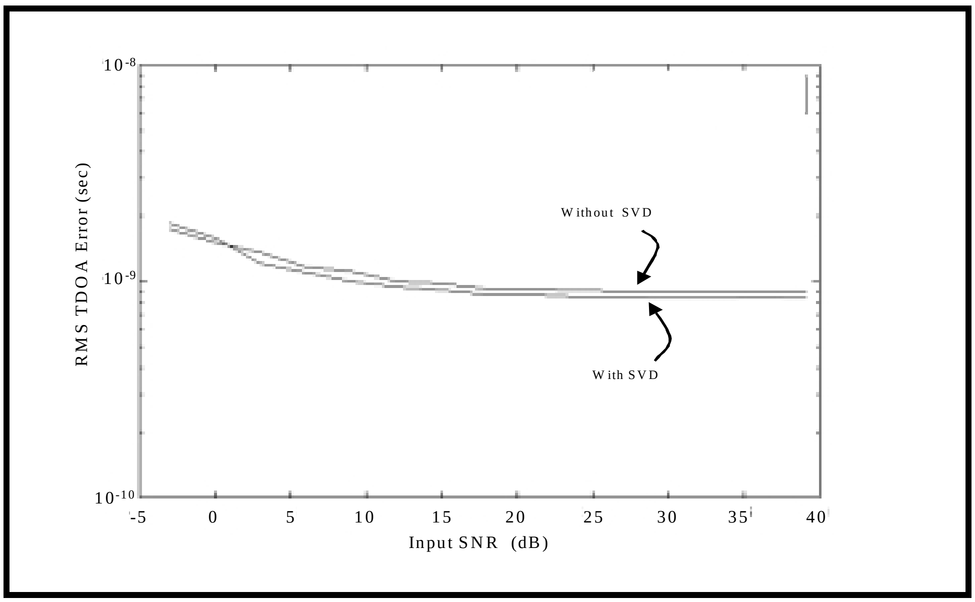

Figure 2

Table 1

PW (µs)	**PRI** (µs)	**BW** (MHz)	**# Pulses** *p*	**Samples/ pulse** *n*	***CR**$_{semi}$* **Eq. (6)**	***CR**$_{coho}$* **Eq. (10) w/ *k*=1**
0.5	600	4.0	80 – 300	6	80 – 300	10.7 – 21.0
1.5	10	2.0	1,500 – 14,000	8	1,500 – 14,000	54.5 – 167.1
6.5	70	2.5	250 – 1,500	24	250 – 1,500	38.5 – 94.6
9.0	240	2.8	60 – 400	38	60 – 400	23.6 – 61.4