where $0 < \omega G_1 < 2N$ since Var $[\phi_{ss}]$ must be positive. One can note that the steady-state mean and variance of phase error of the DTL are functions of M and N. The larger the values of M and N (under the constraint of $N/M < 1$), the better the DTL performance in locking range, and the steady-state behavior. However, one cannot increase M indefinitely because of limitation in real-time processing of the DTL.

In conclusion, in the improved DTL the locking range can be extended considerably by increasing M and N ($N < M$) without affecting the stability of the loop. Moreover, its steady-state phase error can be reduced significantly by a factor of N/M.

References

[1] J. C. Lee and C. K. Un, "Performance analysis of digital tanlock loop," *IEEE Trans. Commun.*, vol. COM-30, pp. 2398–2411, Oct. 1982.

[2] H. J. Kim, C. K. Un, and J. C. Lee, "The N-phase digital tanlock loop for tracking suppressed-carrier N-ary PSK signals," *IEEE Trans. Commun.*, vol., COM-33, pp. 904–910, Sept. 1985.

[3] M. Hagiwara and M. Nakagawa, "Digital tanlock loop with multisampling," submitted for publication.

A Remark on the Definition of Costas Arrays

WEITA CHANG

A weaker definition of Costas arrays is shown to be equivalent to the standard one. A Costas array is a diagram whose corresponding frequency-hopping pattern has good range-doppler ambiguity properties. Our result implies that even when we demand ostensibly less restrictive ambiguities in the doppler direction, the resulting waveform becomes a Costas waveform.

Let us consider the set of all $n \times n$ arrays consisting of n rows and columns, having exactly one dot on each row and each column. The standard definition of a Costas array is as follows (cf. [1], [2]):

Definition: An $n \times n$ array is called a Costas array if it satisfies:

$$C(r, s) \le 1, \quad \text{for all integer pairs } (r, s) \ne (0, 0) \tag{1}$$

with

$$|r| \le n \quad |s| \le n$$

where the coincidence function C is defined as $C(r, s)$ = the number of coincidences of dots between the original array and its translated version, the one shifted to the right by r, up by s. (When r is negative, this means the shift to the left by $-r$.)

Such arrays are important because they correspond to a class of radar/sonar frequency-hopping waveforms having good ambiguity properties ([1]). In this context, the horizontal coordinate r represents the time delay and the vertical coordinate s the doppler shift. The coincidence function C above represents a discrete version of the unnormalized ambiguity function.

The main purpose of this letter is to point out that the condition (1) is equivalent to the following seemingly weaker condition:

$$C(r, s) \le 1, \quad \text{for all integer pairs } (r, s) \ne (0, 0) \tag{2}$$

with

$$|r| \le n \quad |s| \le \left[\frac{n-1}{2}\right]$$

where $[x]$ is the largest integer that does not exceed x. (For example, the range of s becomes

$$|s| \le 2, \quad \text{if } n = 5, 6$$
$$|s| \le 3, \quad \text{if } n = 7, 8.$$

Before proving the equivalence of (1) and (2), it is convenient to recall the concept of the difference vectors. For each pair of dots, A, B of the array, the vector $\vec{AB}$ is called the difference vector. There are

$$\binom{n}{2}$$

difference vectors, or

$$2 \cdot \binom{n}{2}$$

if one counts both $\vec{AB}$ and $\vec{BA}$. From the standard definition (1) it follows easily that (cf. [2], [3])

an $n \times n$ array is Costas if all the difference vectors are distinct (c.f. Fig. 1). (3)

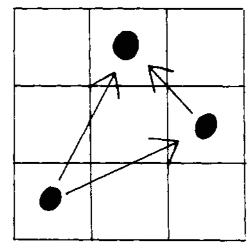

Fig. 1. A 3 × 3 Costas array.

More in detail, we have

for each (r, s) the condition $C(r, s) \le 1$ is equivalent to saying that there is at most one difference vector with coordinates (r, s). (4)

Now we are ready to show that (2) implies (1). Assume (2) holds but (1) does not. Then there must be two identical difference vectors $\vec{AB} - \vec{CD}$ where A and C are the initial dots, B and D are the terminal dots. Without loss of generality, we assume that the vertical component of $\vec{AB}$ to be positive. Since (2) is assumed, in view of (4), the dots A and C must be located in the lower half of the array and B and D in the upper half. (Otherwise the vector $\vec{AB}$ would have a vertical component less than or equal to

$$\left[\frac{n-1}{2}\right]$$

and this would contradict (2).) Trivially, $\vec{AC} = \vec{BD}$ (c.f., Fig. 2). Since

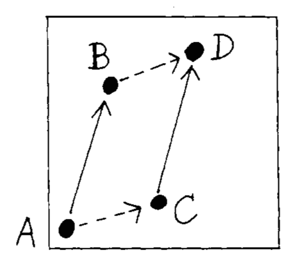

Fig. 2.

A and C are in the lower half, the vertical component of AC is less than or equal to

$$\left[\frac{n-1}{2}\right]$$

in magnitude. Thus we have two identical difference vectors $\vec{AC} = \vec{BD}$ which contradicts (2).

Thus the equivalence between (1) and (2) is proved. Now one can define a Costas array by

Definition: An $n \times n$ Costas array is one that satisfies the condition

$$C(r, s) \le 1, \quad \text{for all integer pairs } (r, s) \ne (0, 0) \tag{2}$$

Manuscript received August 21, 1986.

The author is with the Naval Underwater Systems Center, New London, CT 06320, USA.

with

$$|r| \leq n \qquad |s| \leq \left[\frac{n-1}{2}\right].$$

Summary

Our remark implies, for example, that in order to find a 6 × 6 Costas array, it suffices to find an array whose coincidence function is less than or equal to one only for the vertical shifts 1 and 2, instead of all vertical shifts from 1 up to 5. This may help in search for higher dimensional Costas arrays.

Acknowledgment

The author wishes to thank Dr. H. Taylor of the University of Southern California at Los Angeles for helpful information on Costas arrays.

References

[1] J. Costas, "A study of a class of detection waveform having nearly ideal range-doppler ambiguity properties," *Proc. IEEE*, vol. 72, no. 8, pp. 996–1009, Aug. 1984.
[2] S. W. Golomb and H. Taylor, "Construction and properties of Costas arrays," *Proc. IEEE*, vol. 72, no. 9, pp. 1143–1163, Sept. 1984.
[3] H. Taylor, "Non-attacking rooks with distinct differences," Communication Science Institute Tech. Rep. 84-03-02, University of Southern California, Los Angels, CA 90089-0272.
[4] A. Freedman and N. Levanon, "Any N × N Costas signals must have at least one common ambiguity sidelobe if N > 3—A proof," *Proc. IEEE*, vol. 73, no. 10, pp. 1530–1531, Oct. 1985.

Fast Interpolation Algorithm Using Fast Hartley Transform

JOHNSON I. AGBINYA

The use of fast Hartley transform for fast discrete interpolation is considered. The computational method uses the split-radix algorithm which requires the least number of operations compared with other Hartley algorithms. Results from this method are compared with those using the fast Fourier transform.

Introduction

Discrete interpolation using the fast Fourier Transform (FFT) is discussed in [1]–[3]. This method requires complex additions and multiplications of a sequence of n samples and takes appreciable computer time. The inverse FFT also differs from the forward transform, which adds to the complexity of the method. Bracewell [4], [5] proposed a fast Hartley transform (FHT) which represents both the forward and inverse transform in the same way, and involves only real-valued data. Computational time savings is apparent by the use of this method.

Since the work by Bracewell, interests on discrete Hartley transforms has been renewed [6], [7]. It has been established in [6] and [8] independently that the split-radix FHT is twice faster than the split-radix FFT. Emphasis has thus only been put on the speed of usage.

The split-radix FHT was used in [8] to implement a discrete Hartley transform (DHT). It is shown to be faster than other currently known FHT algorithms. An area of equal interest in using FHT is the accuracy of usage.

In this letter, we show that using the FHT instead of FFT for discrete signal processing leads not only to increased computational speed but to increased accuracy, as well. The split-radix FHT is used to implement fast interpolation on a finite sequence of data. The sampling rate of FHT is also higher than the Nyquist rate. Comparison of mean square error and computational time using both the FHT and FFT with the classical algorithm is given.

Manuscript received July 16, 1986.
The author is with the Department of Electronics and Communication, Faculty of Engineering, University of Jos, Makurdi Campus, Makurdi, Nigeria.

Fast Hartley Transform

The problem of finding values of a function between two samples is known as interpolation. When an analog band-limited function is sampled at a uniform rate $1/T$ greater than $2W$ (W being the highest frequency of the spectrum of $f(t)$) exceeding the Nyquist rate, the function itself can be reconstructed using the formula [1]

$$y(t) = \sum_{n=-\infty}^{\infty} f(nT) \cdot \frac{\sin(\pi/T(t-nT))}{\pi/T(t-nT)}. \tag{1}$$

For a practical case it is physically impossible to compute infinite terms, and if the function is causal, then (1) is truncated as

$$y'(t) = \sum_{n=0}^{N-1} f(nT) \cdot \frac{\sin(\pi/T(t-nT))}{\pi/T(t-nT)}. \tag{2}$$

We take S new samples in between adjacent pairs of samples of the original finite-length sequence of length N samples, using a new sampling period T_f such that

$$T_f = T/(S+1) \tag{3}$$

and the total number of samples P in the new sequence is given by

$$P = N + (N-1)\,S. \tag{4}$$

Each period T of the original sequence is divided into S new uniform periods. Equations (3) and (4) transform (2)into

$$y(mT_f) = \sum_{n=0}^{N-1} f(nT) \cdot \frac{\sin(\pi(m/(S+1)-n))}{\pi(m/(S+1)-n))}, \quad m = 0, 1, 2, \cdots, P-1. \tag{5}$$

That is the discrete interpolation formula which may be evaluated directly. Such a method requires divisions (which take long computer times) and computing the sinc ratio N times which is time consuming. Usually an FFT algorithm is used instead of the direct method to evaluate (5) [1] to increase speed. Recently the DHT has been considered as a good alternative for FFT since no complex values are involved. The DHT is defined as

$$F_k = \sum_{n=0}^{N-1} f(nT) \cdot \mathrm{cas}\,(a/N) \tag{6}$$

where $\mathrm{cas}\,(a/N) = \cos(a/N) + \sin(a/N)$ and $a = 2\pi nk$. If F_k is the DHT of the original sequence $\{f(nT)\}$ and F'_k is the DHT of the new sequence $\{f(mT_f)\}$; we may then write

$$F'_k = H(mT_f) \cdot F_k$$

where $H(mT_f)$ acts as a transform transfer function that changes the first short sequence $\{f(nT)\}$ of N samples into the long sequence $\{f(mT_f)\}$ with P samples. Define the filtering action of $H(mT_f)$ as a low-pass filter of gain given by

$$H(mT_f) = \mathrm{rect}\,(w) = \begin{cases} P+1, & |w| < \pi/T \\ 0, & \pi/T < |w| < \pi/T_f \end{cases}$$

and the sequence $\{f(nT)\} = 1, 2, \cdots, N$. Using the techniques of split-radix FHT [8], we first decompose (6) into even radix-2 and odd-term radix-4 simultaneously and compute F_k as

$$F_{2k} = \sum_{n=0}^{N/2-1} (f_n + f_{(N/2)+n}) \cdot \mathrm{cas}\,(a/(N/2))$$

$$F_{4k+1} = \sum_{n=0}^{N/4-1} [(A_n + A_{(N/4)-n}) \cdot \cos(2\pi n/N) - (B_n - B_{(N/4)-n}) \cdot \sin(2\pi n/N)] \cdot (\mathrm{cas}\,(a/(N/4))$$

$$F_{4k+3} = \sum_{n=0}^{N/4-1} \left[(A_n - A_{(N/4)-n}) \cdot \cos\left(\frac{2\pi}{N} \cdot 3n\right) + (B_n + B_{(N/4)-n}) \cdot \sin\left(\frac{2\pi}{N} \cdot 3n\right)\right] \cdot \mathrm{cas}\,(a/(N/4)) \tag{7}$$

0018-9219/87/0400-0523$01.00 © 1987 IEEE