A SHORT SURVEY ON ARITHMETIC TRANSFORMS AND THE ARITHMETIC HARTLEY TRANSFORM

Renato José de Sobral Cintra and Hélio Magalhães de Oliveira

Abstract - Arithmetic complexity has a main role in the performance of algorithms for spectrum evaluation. Arithmetic transform theory offers a method for computing trigonometrical transforms with minimal number of multiplications. In this paper, the proposed algorithms for the arithmetic Fourier transform are surveyed. A new arithmetic transform for computing the discrete Hartley transform is introduced: the Arithmetic Hartley transform. The interpolation process as the key to the arithmetic transform theory is also examined.

Keywords: Arithmetic transforms, discrete transforms, Fourier series, VLSI implementations.

Resumo - A complexidade aritmética ocupa um papel de destaque no desempenho de algoritmos para o cálculo de espectros. As transformadas aritméticas proporcionam um método para o cálculo de transformadas trigonométricas, minimizando-se o número de operações de multiplicação. Neste artigo, os algoritmos existentes para a transformada aritmética de Fourier são discutidos. Uma nova transformada aritmética para o cálculo da transformada discreta de Hartley é introduzida: a transformada aritmética de Hartley. O processo de interpolação é examinado com o ponto crucial das transformadas aritméticas.

Palavras-chave: Transformadas aritméticas, transformadas discretas, série de Fourier, implementações em VLSI.

1. INTRODUCTION AND HISTORICAL BACKGROUND

Despite the existence of fast algorithms for discrete transforms (e.g., fast Fourier transform, FFT), it is well known that the number of multiplications can significantly increase their computational (arithmetic) complexity. Even today, the multiplication operation consumes much more time than addition or subtraction. Table 1 brings the clock count of some mathematical operations as implemented for the Intel Pentium™ processor. Observe that multiplications and divisions can be by far more time demanding than additions, for instance. Sine and cosine function costs are also shown.

This fact stimulated the research on discrete transform algorithms that minimize the number of multiplications. The Bhatnagar's algorithm [1a], which uses Ramanujan numbers to eliminate multiplications (however, the choice of the transform blocklength is rather limited), is an example. Parallel to this, approximation approaches, which perform a trade-off between accuracy and computational complexity, have been proposed [2a, 3a, 4a]

Operation	Clock count
`add`	1–3
`sub`	1–3
`fadd`	1–7
`fsub`	1–7
`mul` (unsigned)	10–11
`mul` (signed)	10–11
`div` (unsigned)	17–41
`div` (signed)	22–46
`fdiv`	39
`sin`, `cos`	17–137

Table 1. Clock count for some arithmetic instructions carried on a Pentium™ processor. See "The Pentium Processor" by J. L. Antonakos for detailed data.

Arithmetic transforms emerged in this framework as an algorithm for spectrum evaluation, aiming the elimination of multiplications. Thus, it would offer a lower computational complexity. The theory of arithmetic transform is essentially based on Möbius function theorems [5a], offering only trivial multiplications, i.e., multiplications by $\{-1, 0, 1\}$. Therefore, only addition operations (except for multiplications by scale factors) are left to computation. Beyond the computational attractiveness, arithmetic transforms turned out to be naturally suited for parallel processing and VLSI design [6, 18].

The very beginning of research on arithmetic transforms dates back to 1903 when the German mathematician Ernest Heinrich Bruns[1] published the *Grundlinien des wissenschaftlichnen Rechnens* [3], the seminal work in this field. In spite of that, the technique remained unnoticed even among mathematicians for a long time. Forty-two years later, in Baltimore, U.S.A., the Hungarian Aurel Freidrich Wintner[2], privately published a monograph entitled *An Arithmetical Approach to Ordinary Fourier Series*. This monograph presented an arithmetic method using Möbius function to calculate the Fourier series of even periodic functions.

After Wintner's monograph, the theory entered again in "hibernation" state. Not before 1988, Dr. Donald W. Tufts and Dr. Angaraih G. Sadasiv, independently, had reinvented

The authors are with the Communications Research Group – CODEC, Department of Electronics and Systems, Federal University of Pernambuco, Recife, Brazil. Emails: rjsc@ee.ufpe.br, hmo@ufpe.br. Review coordinated by Renato Baldini Filho and Jaime Portugheis (Area Editors). Manuscript received May/19/2003, reviewed Aug/26/2003, Mar/18/2004, accepted Aug/04/2004.

[1]Bruns (1848-1919) got a doctorate in 1871 under supervision of Weierstrass and Kummer.

[2]A curious fact: Wintner was born in April 8th 1903 in Budapest, the same year Bruns had published the *Grundlinien*. Wintner died on January 15th 1958 in Baltimore.

Wintner's arithmetical procedure, reawaking the arithmetic transform.

In the quest to implement it, two other researchers played an important role: Dr. Oved Shisha of the U.R.I. Department of Mathematics and Dr. Charles Rader of Lincoln Laboratories. They were aware of Wintner's monograph and helped Tufts in many discussions. In 1988 *The Arithmetic Fourier Transform* by Tufts and Sadasiv was published in IEEE Acoustic, Speech, and Signal Processing (ASSP) Magazine [6].

Another breakthrough came in early 1990s when Emeritus Professor Dr. Irving S. Reed entered in scene. Although Dr. Reed is better recognized for his work on coding theory — since he is the main originator of the widely used Reed-Muller (1954) and Reed-Solomon (1964) codes — his interests were definitely not limited to codes. Author of hundreds of publications, Dr. Reed made important contributions to the area of signal processing. Specifically on arithmetic transforms, in 1990 Reed, Tufts and co-workers provided two fundamental contributions [15, 26].

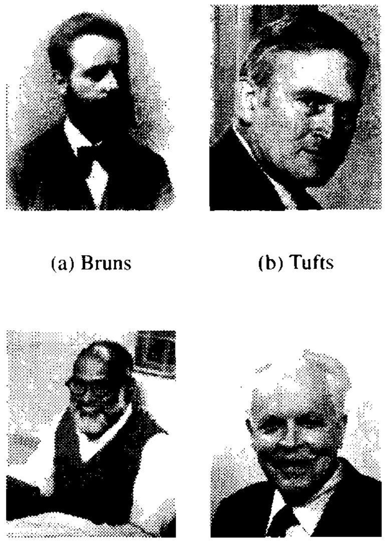
In the light of [15], a reformulated version of Tufts-Sadasiv approach, the arithmetic Fourier transform (AFT) algorithm was able to encompass a larger class of signals and to compute Fourier series of odd periodic functions as well as even periodic ones.

The publication of the 1992 *A VLSI Architecture for Simplified Arithmetic Fourier Transform Algorithm* by Dr. Reed and collaborators in the IEEE Transactions on ASSP [26] was another crucial slash on the subject. Indeed, that paper was previously presented at the *International Conference on Application Specific Array Processors* held in Princeton. However, the 1992 publication reached a vastly larger public, since it was published in a major journal. The new method, an enhancement of the last proposed algorithm [15], was redesigned to have a more balanced and computationally efficient performance. As a matter of fact, Reed *et al.* proved that the newly proposed algorithm was identical to Bruns' original method.

When the AFT was introduced, some concerns on the feasibility of the AFT were pointed out [10]. The main issue dealt with the number of samples required by the algorithm. However, later studies showed that the use of interpolation techniques on a sub-sampled set (e.g., zero- and first-order interpolation) could overcome these difficulties [11].

The conversion of the standard 1-D AFT into 2-D versions was just a matter of time. Many variants were proposed following the same guidelines of the 1-D case [8, 20, 39, 43, 44, 12]. Further research was carried out seeking different implementations of the AFT. An alternative method [32] proposed a "Möbius-function-free AFT". Iterative [30] and adaptative approaches [16] were also examined. In spite of that, the most popular presentations of the AFT are still those found in [15, 26].

Although the main and original motivation of the arithmetic algorithm was the computation of the Fourier Transform, further generalizations were performed and the arithmetic approach was utilized to calculate other transforms. Dr. Luc Knockaert of Department of Information Technology at Ghent University, Belgium, amplified the Bruns procedure,



(a) Bruns (b) Tufts

(c) Sadasiv (d) Reed

Figure 1. Some important people in the history of the arithmetic transform algorithm (see the text).

defining a generalized Möbius transform [35, 38]. Moreover, four versions of the cosine transform was shaped in the arithmetic transform formalism [40].

Further generalization came in early 2000s with the definition of the Arithmetic Hartley Transform (AHT) [48, 47]. These works constituted an effort to make arithmetical procedure applicable for the computation of trigonometrical transforms, other than Fourier transform. In particular the AHT computes the discrete Hartley transform[3]: the real, symmetric, Fourier-like discrete transform defined in 1983 by Emeritus Professor Ronald Newbold Bracewell in *The Discrete Hartley Transform*, an article published in the Journal of Optical Society of America.

In 1988 and then the technological state-of-art was dramatically different from that Bruns and Wintner found. Computational facilities and digital signal processing integrated circuits made possible AFT to leave theoretical constructs and reach practical implementations. Since its inception in engineering, the AFT was recognized as tool to be implemented with VLSI techniques. Tufts himself had observed that AFT could be naturally implemented in VLSI architectures [6]. Implementations were proposed in [17, 21, 27, 22, 24, 29, 31, 18, 26, 43, 14, 19, 23, 36]. Initial applications of the AFT took place in several areas: pattern matching techniques [28], measurement and instrumentation [37, 41], auxiliary tool for computation of z-transform [34, 33], and imaging [13].

This paper is organized in two parts. In section 2, the

[3]Ralph Vinton Lyon Hartley (1888-1970) introduced his real integral transform in a 1942 paper published in the *Proceedings of I.R.E.* The Hartley transform relates a pair of signals $f(t) \longleftrightarrow F(\nu)$ by

$$F(\nu) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)(\cos(\nu t) + \sin(\nu t))dt,$$

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\nu)(\cos(\nu t) + \sin(\nu t))d\nu.$$

mathematical evolution of the Arithmetic Fourier Transform is outlined. In section 3, a summary of the major results on the Arithmetic Hartley Transform is shown. Interpolation issues are addressed and many points of the AFT were clarified, particularly the zero-order approximation.

2. THE ARITHMETIC FOURIER TRANSFORM

Throughout this section, the three major breakthroughs of the arithmetic Fourier transform technique are presented. With emphasis on the theoretical groundwork, the AFT algorithms devised by Tufts, Sadasiv, Reed *et alli* are briefly surveyed.

Before describing the algorithms, it is convenient to call attention to some useful preliminary results. In this work, $k_1|k_2$ denotes that k_1 is a divisor of k_2; $\lfloor\cdot\rfloor$ is the floor function and $[\cdot]$ is the nearest integer function.

Lemma 2.1 *Let k, k' and m be integers.*

$$\sum_{m=0}^{k-1} \cos\left(2\pi m \frac{k'}{k}\right) = \begin{cases} k & \text{if } k|k', \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

and

$$\sum_{m=0}^{k-1} \sin\left(2\pi m \frac{k'}{k}\right) = 0. \tag{2}$$

Proof: Consider the expression $\sum_{m=0}^{k-1}\left(e^{2\pi j \frac{k'}{k}}\right)^m$. When $k|k'$, yields

$$\sum_{m=0}^{k-1}\left(e^{2\pi j \frac{k'}{k}}\right)^m = \sum_{m=0}^{k-1} 1 = k.$$

Otherwise,

$$\sum_{m=0}^{k-1}\left(e^{2\pi j \frac{k'}{k}}\right)^m = \frac{1-e^{j2\pi k'}}{1-e^{j2\pi \frac{k'}{k}}} = 0.$$

Therefore,

$$\sum_{m=0}^{k-1} e^{2\pi j m \frac{k'}{k}} = \begin{cases} k & \text{if } k|k', \\ 0 & \text{otherwise.} \end{cases}$$

Taking real and imaginary parts ends the proof. ∎

Definition 2.1 (Möbius μ-function) *For a positive integer n,*

$$\mu(n) \triangleq \begin{cases} 1 & \text{if } n=1, \\ (-1)^r & \text{if } n=\prod_{i=1}^{r} p_i,\ p_i \text{ distinct primes}, \\ 0 & \text{if } p^2|n \text{ for some prime } p. \end{cases} \tag{3}$$

An interesting lemma using the μ-function is stated below.

Lemma 2.2

$$\sum_{d|n} \mu(d) = \begin{cases} 1 & \text{if } n=1, \\ 0 & \text{if } n>1. \end{cases} \tag{4}$$

Theorem 2.1 (Möbius Inversion Formula for Finite Series) *Let n be a positive integer and f_n a non-null sequence for $1 \le n \le N$ and null for $n > N$. If*

$$g_n = \sum_{k=1}^{\lfloor N/n \rfloor} f_{kn}, \tag{5}$$

then

$$f_n = \sum_{m=1}^{\lfloor N/n \rfloor} \mu(m) g_{mn}. \tag{6}$$

∎

This is the finite version of the Möbius inversion formula [5a]. A proof can be found in [15].

2.1 TUFTS-SADASIV APPROACH

Consider a real even periodic function expressed by its Fourier series, as seen below:

$$v(t) = \sum_{k=1}^{\infty} v_k(t). \tag{7}$$

The components $v_k(t)$ represent the harmonics of $v(t)$, given by:

$$v_k(t) = a_k \cdot \cos(2\pi k t), \tag{8}$$

where a_k is the amplitude of the kth harmonic.

It was assumed, without loss of generality, that $v(t)$ had unitary period and null mean ($a_0 = 0$). Furthermore, consider the N first harmonics as the only significant ones, in such a way that $v_k(t) = 0$, for $k > N$ (bandlimited approximation). Thus the summation of Equation (7) might be constrained to N terms.

Definition 2.2 *The nth average is defined by*

$$S_n(t) \triangleq \frac{1}{n}\sum_{m=0}^{n-1} v\left(t - \frac{m}{n}\right), \tag{9}$$

for $n = 1, 2, \ldots, N$. $S_n(t)$ is null for $n > N$. ∎

After an application of equations (7) and (8) into 9, it yielded:

$$\begin{aligned} S_n(t) &= \frac{1}{n}\sum_{m=0}^{n-1} v\left(t-\frac{m}{n}\right) \\ &= \frac{1}{n}\sum_{m=0}^{n-1}\sum_{k=1}^{\infty} a_k \cos\left(2\pi k t - 2\pi k \frac{m}{n}\right) \\ &= \frac{1}{n}\sum_{k=1}^{\infty} a_k \sum_{m=0}^{n-1}\left(\cos(2\pi k t)\cos\left(2\pi k\frac{m}{n}\right)\right. \\ &\qquad \left. - \sin(2\pi k t)\sin\left(2\pi k \frac{m}{n}\right)\right) \\ &= \frac{1}{n}\sum_{k=1}^{\infty} a_k \cos(2\pi k t)\cdot \begin{Bmatrix} n & \text{if } n|k, \\ 0 & \text{otherwise} \end{Bmatrix} \\ &= \sum_{n|k}^{\infty} v_k(t) = \sum_{m=1}^{\infty} v_{mn}(t), \quad n=1,\ldots,N. \end{aligned} \tag{10}$$

Proceeding that way, the nth average could be written in terms of the harmonics of $v(t)$, instead of its samples (Definition 2.2). Since we assumed $v_n(t) = 0, n > N$, only the first $\lfloor N/n \rfloor$ terms of Equation (10) might possibly be nonnull.

As a consequence the task was to invert Equation (10). Doing so, the harmonics could be expressed in terms of the averages, $S_n(t)$, which were derived from the samples of the signal $v(t)$. The inversion was accomplished by invoking the Möbius inversion formula.

Theorem 2.2 *The harmonics of $v(t)$ can be obtained by:*

$$v_k(t) = \sum_{m=1}^{\infty} \mu(m) S_{mk}(t), \quad \forall k = 1, \ldots, N. \tag{11}$$

Proof: Some manipulation is needed. Substituting Equation (10) into Equation (11), it yields

$$\sum_{m=1}^{\infty} \mu(m) S_{mk}(t) = \sum_{m=1}^{\infty} \mu(m) \sum_{n=1}^{\infty} v_{kmn}(t). \tag{12}$$

Now it is the tricky part of the proof.

$$\begin{aligned} \sum_{m=1}^{\infty} \mu(m) \sum_{n=1}^{\infty} v_{kmn}(t) &= \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu(m) v_{kmn}(t) \\ &= \sum_{j=1}^{\infty} v_j(t) \left(\sum_{m | \frac{j}{k}}^{\infty} \mu(m) \right). \end{aligned} \tag{13}$$

According to Lemma 2.2, the inner summation can only be null if $j/k = 1$. In other words, the term $v_k(t)$ is the only survivor of the outer summation and the proof is completed. ■

The following aspects of the Reed-Tufts algorithm could be highlighted [6]:

- This initial version of the AFT had a strong constraint: it could only handle even signals;
- All computations were performed using only additions (except for few multiplications due to scaling);
- The algorithm architecture was suitable for parallel processing, since each average was computed independently from the others;
- The arithmetic transform theory was based on Fourier series, instead of the discrete transform itself.

2.2 REED-TUFTS APPROACH

Presented by Reed *et al.* in 1990 [15], this algorithm is a generalization of Tuft-Sadasiv method. The main constraint of the latter procedure (handling only with even signals) was removed, opening path for the computation of all Fourier series coefficient of periodic functions.

Let $v(t)$ be a real T-periodic function, whose N-term finite Fourier series is given by

$$\begin{aligned} v(t) = a_0 &+ \sum_{n=1}^{N} a_n \cos\left(\frac{2\pi n t}{T}\right) \\ &+ \sum_{n=1}^{N} b_n \sin\left(\frac{2\pi n t}{T}\right), \end{aligned} \tag{14}$$

where a_0 is the mean value of $v(t)$. The even and odd coefficients of the Fourier series are a_n and b_n, respectively.

Let $\bar{v}(t)$ denote the signal $v(t)$ removed of its mean value a_0. Consequently,

$$\begin{aligned} \bar{v}(t) &= v(t) - a_0 \\ &= \sum_{n=1}^{N} a_n \cos\left(\frac{2\pi n t}{T}\right) + \sum_{n=1}^{N} b_n \sin\left(\frac{2\pi n t}{T}\right). \end{aligned} \tag{15}$$

A delay (shift) of αT in $\bar{v}(t)$ leaded to the following:

$$\begin{aligned} \bar{v}(t + \alpha T) = &\sum_{n=1}^{N} a_n \cos\left(2\pi n (\frac{t}{T} + \alpha)\right) \\ &+ \sum_{n=1}^{N} b_n \sin\left(2\pi n (\frac{t}{T} + \alpha)\right) \\ = &\sum_{n=1}^{N} c_n(\alpha) \cos\left(2\pi n \frac{t}{T}\right) \\ &+ \sum_{n=1}^{N} d_n(\alpha) \sin\left(2\pi n \frac{t}{T}\right), \end{aligned} \tag{16}$$

where $-1 < \alpha < 1$ and

$$c_n(\alpha) = a_n \cos(2\pi n \alpha) + b_n \sin(2\pi n \alpha), \tag{17}$$

$$d_n(\alpha) = -a_n \sin(2\pi n \alpha) + b_n \cos(2\pi n \alpha). \tag{18}$$

In the sequel, the computation of the Fourier coefficients a_n and b_n based on $c_n(\alpha)$ is outlined. Meanwhile, the formula for the nth average (Tufts-Sadasiv) was updated by the next definition.

Definition 2.3 *The nth average is given by*

$$S_n(\alpha) \triangleq \frac{1}{n} \sum_{m=0}^{n-1} \bar{v}\left(\frac{m}{n} T + \alpha T\right), \tag{19}$$

where $-1 < \alpha < 1$.

Now the quantities $c_n(\alpha)$ could be related to the averages, according to the following Theorem.

Theorem 2.3 *The coefficients $c_n(\alpha)$ are computed via Möbius inversion formula for finite series and are expressed by*

$$c_n(\alpha) = \sum_{l=1}^{\lfloor N/n \rfloor} \mu(l) S_{ln}(\alpha). \tag{20}$$

Proof: Substituting the result of Equation (16) into Equation (19):

$$\begin{aligned} S_n(\alpha) = &\sum_{k=1}^{N} c_k(\alpha) \frac{1}{n} \sum_{m=0}^{n-1} \cos\left(\frac{2\pi k m}{n}\right) \\ &+ \sum_{k=1}^{N} d_k(\alpha) \frac{1}{n} \sum_{m=0}^{n-1} \sin\left(\frac{2\pi k m}{n}\right). \end{aligned} \tag{21}$$

A direct application of Lemma 2.1 yields

$$S_n(\alpha) = \sum_{l=1}^{\lfloor N/n \rfloor} c_{ln}(\alpha). \tag{22}$$

Invoking the Möbius inversion formula for finite series, the theorem is proved. ■

Finally, the main result could be derived.

Theorem 2.4 (Reed-Tufts) *The Fourier series coefficients a_n and b_n are computed by*

$$a_n = c_n(0), \tag{23}$$

$$b_n = (-1)^m c_n\left(\frac{1}{2^{k+2}}\right) \quad n = 1, \ldots, N, \tag{24}$$

where k and m are determined by the factorization $n = 2^k(2m+1)$.

Proof: For $\alpha = 0$, using Equation (17), it is straightforward to show that $a_n = c_n(0)$. For $\alpha = \frac{1}{2^{k+2}}$ and $n = 2^k(2m+1)$, there are two sub-cases: m even or odd.

- For $m = 2q$, $n = 2^k(4q+1)$. Therefore,

$$2\pi n\alpha = 2\pi\frac{2^k(4q+1)}{2^{k+2}} = 2\pi q + \frac{\pi}{2}. \tag{25}$$

Consequently, substituting this quantity into Equation (17), yields

$$\begin{aligned} c_n\left(\frac{1}{2^{k+2}}\right) &= a_n \cos\left(2\pi q + \frac{\pi}{2}\right) \\ &\quad + b_n \sin\left(2\pi q + \frac{\pi}{2}\right) \\ &= b_n. \end{aligned} \tag{26}$$

- For $m = 2q+1$, $n = 2^k(4q+3)$. It follows that

$$2\pi n\alpha = 2\pi\frac{2^k(4q+3)}{2^{k+2}} = 2\pi q + \frac{3\pi}{2}. \tag{27}$$

Again invoking the Equation (17), the following expression is derived.

$$\begin{aligned} c_n\left(\frac{1}{2^{k+2}}\right) &= a_n \cos\left(2\pi q + \frac{3\pi}{2}\right) \\ &\quad + b_n \sin\left(2\pi q + \frac{3\pi}{2}\right) \\ &= -b_n. \end{aligned} \tag{28}$$

Joining these two sub-cases, it is easy to verify that

$$b_n = (-1)^m c_n\left(\frac{1}{2^{k+2}}\right). \tag{29}$$

■

The number of real multiplications and additions of this algorithm were given by [15]

$$M_R(N) = \frac{3}{2}N, \tag{30}$$

and

$$A_R(N) = \frac{3}{8}N^2, \tag{31}$$

respectively, where N is the blocklength of the transform.

2.3 REED-SHIH (SIMPLIFIED AFT)

Introduced by Reed *et al.* [18], this algorithm is an evolution of that one developed by Reed and Tufts. Surprisingly, in this new method, the averages were re-defined in accordance to the theory created by H. Bruns [3] in 1903.

Definition 2.4 (Bruns Alternating Average) *The $2n$th Bruns alternating average, $B_{2n}(\alpha)$, is defined by*

$$B_{2n}(\alpha) \triangleq \frac{1}{2n}\sum_{m=0}^{2n-1}(-1)^m \cdot v\left(\frac{m}{2n}T + \alpha T\right). \tag{32}$$

Invoking the definition of c_n, applying Theorem 2.3 and Definition 2.3, the following theorem was derived.

Theorem 2.5 *The coefficients $c_n(\alpha)$ are given by the Möbius inversion formula for finite series as*

$$c_n(\alpha) = \sum_{l=1,3,\ldots}^{\lfloor \frac{N}{n} \rfloor} \mu(l) \cdot B_{2nl}(\alpha). \tag{33}$$

Proof: See [26]. ■

Since a relation between the signal samples and the Brun alternating averages was obtained, as well as an expression connecting the Bruns alternating averages to the c_n coefficients, was available, few points were missing to compute the Fourier series coefficients. Actually, it remained to derive an expression that could relate the Fourier series coefficients (a_n and b_n) and the coefficients c_n. Examining Equation (17), two conditions were distinguishable:

- $a_n = c_n(0)$;
- $b_n = c_n\left(\frac{1}{4n}\right)$.

Those were the final relations. Calling Theorem 2.5, the next result was obtained.

Theorem 2.6 (Reed-Shih) *The Fourier series coefficients a_n and b_n are computed by*

$$a_0 = \frac{1}{T}\int_0^T v(t)\mathrm{d}t, \tag{34}$$

$$a_n = \sum_{l=1,3,5,\ldots}^{\lfloor \frac{N}{n} \rfloor} \mu(l) B_{2nl}(0), \tag{35}$$

$$b_n = \sum_{l=1,3,5,\ldots}^{\lfloor \frac{N}{n} \rfloor} \mu(l)(-1)^{\frac{l-1}{2}} B_{2nl}\left(\frac{1}{4nl}\right), \tag{36}$$

for $n = 1, \ldots, N$.

Proof: The proof is similar to the proof of Theorem 2.4. ■

For a blocklength N, the multiplicative and additive complexities were given by

$$M_R(N) = N, \tag{37}$$

and

$$A_R(N) = \frac{1}{2}N^2, \tag{38}$$

respectively.

The AFT algorithm proposed by Reed-Shih presented some advancements:

- The computation of both a_n and b_n had been readjusted, having roughly the same computational effort. The algorithm became even more balanced than Reed-Tufts algorithm;
- The algorithm was naturally suited to a parallel processing implementation;
- It was computationally less complex that Reed-Tufts algorithm.

2.4 AN EXAMPLE

In this subsection, some comments to an example of the Reed-Shih algorithm are done. Let $v(t)$ be a signal with period $T = 1$ s. Consider the computation of the Fourier series coefficients up to the 5th harmonic.

According to Reed-Shih algorithm, the coefficients a_n and b_n of the Fourier series of $v(t)$ were expressed by

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} B_2(0) \\ B_4(0) \\ B_6(0) \\ B_8(0) \\ B_{10}(0) \end{bmatrix} \tag{39}$$

and

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} B_2(\frac{1}{4}) \\ B_4(\frac{1}{8}) \\ B_6(\frac{1}{12}) \\ B_8(\frac{1}{16}) \\ B_{10}(\frac{1}{20}) \end{bmatrix}. \tag{40}$$

Comparing these formulations with the ones of Reed-Tufts algorithm, one may note the balance in the computation of a_n and b_n. Both coefficients were obtained through similar matrices. A table relating Bruns alternative averages, $B_n(\alpha)$, with the necessary time samples to compute it, could be constructed.

Bruns averages	Sample time (s)
$B_2(0)$	$0, \frac{1}{2}$
$B_4(0)$	$0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$
$B_6(0)$	$0, \frac{1}{6}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{5}{6}$
$B_8(0)$	$0, \frac{1}{8}, \frac{1}{4}, \frac{3}{8}, \frac{1}{2}, \frac{5}{8}, \frac{3}{4}, \frac{7}{8}$
$B_{10}(0)$	$0, \frac{1}{10}, \frac{1}{5}, \frac{3}{10}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{7}{10}, \frac{4}{5}, \frac{9}{10}$
$B_2(\frac{1}{4})$	$\frac{1}{4}, \frac{3}{4}$
$B_4(\frac{1}{8})$	$\frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}$
$B_6(\frac{1}{12})$	$\frac{1}{12}, \frac{1}{4}, \frac{5}{12}, \frac{7}{12}, \frac{3}{4}, \frac{11}{12}$
$B_8(\frac{1}{16})$	$\frac{1}{16}, \frac{3}{16}, \frac{5}{16}, \frac{7}{16}, \frac{9}{16}, \frac{11}{16}, \frac{13}{16}, \frac{15}{16}$
$B_{10}(\frac{1}{20})$	$\frac{1}{20}, \frac{3}{20}, \frac{1}{4}, \frac{7}{20}, \frac{9}{20}, \frac{11}{20}, \frac{13}{20}, \frac{3}{4}, \frac{17}{20}, \frac{19}{20}$

Table 2. Necessary samples for the Bruns alternating averages.

Table 2 shows that at least 40 non-uniform time samples of $v(t)$ were necessary to exactly compute the Bruns alternating averages, and then the Fourier series coefficients.

At this point, some observations were relevant:

- This algorithm is not naturally suited for uniform sampling.
- A uniform sampler utilized to obtain all the necessary samples would need a sampling rate too high. In the example illustrated here, a 120 Hz clock should be required to sample the necessary points for the computation of the Fourier series of a 1 Hz bandlimited signal.

Certainly these observations appear to be disturbing and seems to jeopardize the feasibility of the whole procedure. However, it is important to stress that this procedure furnishes the *exact* computation of the Fourier series coefficients.

An empirical solution to circumvent this problem is to interpolate. An interpolation based on uniformly sampled points could be used to estimate the sample values required by AFT. Of course, this procedure inherently introduces computation errors.

For example, assuming that the 1Hz signal $v(t)$ was sampled by a clock with period $T_0 = \frac{1}{10}$ s. Hence, the following sample points were available:

$$v(0), v\left(\frac{1}{10}\right), v\left(\frac{2}{10}\right), v\left(\frac{3}{10}\right), v\left(\frac{4}{10}\right),$$
$$v\left(\frac{5}{10}\right), v\left(\frac{6}{10}\right), v\left(\frac{7}{10}\right), v\left(\frac{8}{10}\right), v\left(\frac{9}{10}\right).$$

Table 2 shows, for example, that the computation of $B_4(0)$ requires — among other samples — $v\left(\frac{1}{4}\right)$, which is clearly not available. To overcome this difficulty, a rounding operation could be introduced. Thus, the sample $v\left(\frac{3}{10}\right)$ could be used whenever the algorithm called $v\left(\frac{1}{4}\right)$ ($\left[10\frac{1}{4}\right]/10 = 3/10$). This rounding operation is also known as zero-order interpolation.

The accuracy of the AFT algorithm is deeply associated with the sampling period T_0. If more precision is required, then one should expect to increase sampling rate, resulting in the introduction of smaller errors due to the interpolation scheme. Higher order of interpolation (e.g. first-order interpolation) could also be used to obtain more accurate estimations of the Fourier series coefficients. The following trade-off is quite clear accuracy versus order of interpolation.

However, for signals sampled at Nyquist rate (or close to), zero-order interpolation already leads to good results [15]. A detailed error analysis of interpolation schemes can be found in [9, 15, 40, 34]. Further comments can be found in [18].

3. A NEW ARITHMETIC TRANSFORM

Besides its numerical appropriateness [6a], the discrete Hartley transform (DHT) has proved along the years to be an important tool with several applications, such as biomedical image compression, OFDM/CDMA systems, and ADSL transceivers. Searching the literature, no mention about a possible "Arithmetic Hartley Transform" to compute the DHT was found.

In this section, a condensation of the main results of the Arithmetic Hartley Transform is outlined. The method used to define the AHT turned out to furnish a new insight into the arithmetic transform. In particular, the role of interpolation was clarified. Additionally, it was mathematically shown that interpolation is a pivotal issue in arithmetic transforms. Indeed it determines the transform.

A new approach to arithmetic transform is adopted. Instead of considering uniformly sampled points extracted from a continuous signal $v(t)$, the AHT is based on the purely discrete signal. Thus, the starting point of the development is the discrete transform definition, not the series expansion, as it was done in the AFT algorithm. This approach is philosophically appealing, since in a final analysis a discrete transform relates two set of points, not continuous functions.

Let $\mathbf{v}$ be an N-dimensional vector with real elements. The DHT establishes a pair denoted by $\mathbf{v} = [v_0, v_1, \ldots, v_{N-1}]^T \leftrightarrow \mathbf{V} = [V_0, V_1, \ldots, V_{N-1}]^T$, where the elements of the transformed vector $\mathbf{V}$ (i.e., Hartley spectrum) are defined by [6a]

$$V_k \triangleq \frac{1}{N}\sum_{i=0}^{N-1} v_i \cdot \text{cas}\left(\frac{2\pi k i}{N}\right), \quad k = 0, 1, \ldots, N-1, \quad (41)$$

where $\text{cas}\, x \triangleq \cos x + \sin x$ is Hartley's "cosine and sine" kernel. The inverse discrete Hartley transform is then [6a]

$$v_i = \sum_{k=0}^{N-1} V_k \cdot \text{cas}\left(\frac{2\pi k i}{N}\right), \quad i = 0, 1, \ldots, N-1. \quad (42)$$

Lemma 3.1 (Fundamental Property) *The function* $\text{cas}(\cdot)$ *satisfies*

$$\sum_{m=0}^{k-1} \text{cas}\left(2\pi m \frac{k'}{k}\right) = \begin{cases} k & \text{if } k|k', \\ 0 & \text{otherwise.} \end{cases} \quad (43)$$

Proof: It follows directly from Lemma 2.1. ■

Similarly to the AFT theory, it was necessary to define averages S_k, calculated from the time-domain elements. The averages were computed by

$$S_k \triangleq \frac{1}{k}\sum_{m=0}^{k-1} v_{m\frac{N}{k}}, \qquad k = 1, \ldots, N-1. \quad (44)$$

It is interesting to note that this definition required fractional index sampling (!). Analogously to the AFT methods, this fact seemed to make further considerations impracticable, since only integer index samples were available. This subtle question is to be treated in the sequel. Meanwhile, the fractional indexes will be treated in mathematical development without concernments.

An application of the inverse Hartley transform (Equation (42)) in Equation (44) offered:

$$S_k = \frac{1}{k}\sum_{m=0}^{k-1}\sum_{k'=0}^{N-1} V_{k'} \,\text{cas}\left(\frac{2\pi k' \frac{mN}{k}}{N}\right). \quad (45)$$

Rearranging the summation order, simplifying, and calling Lemma 3.1, it yielded:

$$\begin{aligned} S_k &= \frac{1}{k}\sum_{k'=0}^{N-1} V_{k'} \sum_{m=0}^{k-1} \text{cas}\left(2\pi m \frac{k'}{k}\right) \\ &= \sum_{s=0}^{\lfloor (N-1)/k \rfloor} V_{sk}. \end{aligned} \quad (46)$$

For simplicity and without loss of generality, consider a signal $\mathbf{v}$ with zero mean value, i.e., $\frac{1}{N}\sum_{i=0}^{N-1} v_i = 0$. Clearly, this consideration has no influence on the values of V_k, $k \neq 0$. An application of the modified Möbius inversion formula for finite series [15] was sufficient to obtain the final theorem to derive the Arithmetic Hartley Transform. According to Theorem 2.1, the following result could be stated:

Theorem 3.1 (Reed *et alli*) *If*

$$S_k = \sum_{s=1}^{\lfloor (N-1)/k \rfloor} V_{sk}, \quad 1 \leq k \leq N-1, \quad (47)$$

then

$$V_k = \sum_{l=1}^{\lfloor (N-1)/k \rfloor} \mu(l) S_{kl}, \quad (48)$$

where $\mu(\cdot)$ *is Möbius function.* ■

To illustrate its usage, consider an 8-point DHT. Using Theorem 3.1, Equation (48), the spectral analysis is given by:

$$\begin{aligned} V_1 &= S_1 - S_2 - S_3 - S_5 + S_6 - S_7, \\ V_2 &= S_2 - S_4 - S_6, \\ V_3 &= S_3 - S_6, \\ V_4 &= S_4, \\ V_5 &= S_5, \\ V_6 &= S_6, \\ V_7 &= S_7. \end{aligned}$$

The component $V_0 = V_8$ can be computed directly from the given samples, since it represents the mean value of the signal $V_0 = \frac{1}{8}\sum_{m=0}^{7} v_m$. In Figure 2, a diagram of this computation is shown.

The above theorem and equations completely specified how to compute the discrete Hartley spectrum. Additionally, the inverse transformation could also be established. The following was straightforward.

Corollary 1 *Inverse discrete Hartley transform components can be computed by*

$$v_i = \sum_{l=1}^{\lfloor (N-1)/i \rfloor} \mu(l) \sigma_{il}, \quad (49)$$

where $\sigma_i \triangleq \frac{1}{i}\sum_{m=0}^{i-1} V_{m\frac{N}{i}}$, $i = 1, \ldots, N-1$. ■

The original Arithmetic *Fourier* Transform had identical equations to those just derived for the *Hartley* transform (compare Equation (10) and Equation (47)). A question arises: since the equations were the same, which spectrum was actually being evaluated? Fourier or Hartley spectrum?

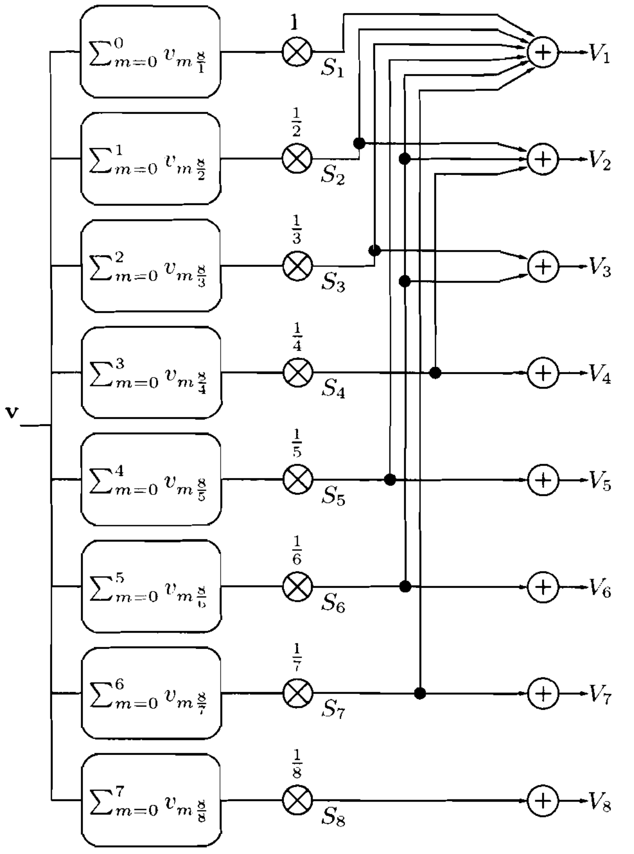


Figure 2. Diagram for computing the AHT for $N = 8$. The boxes compute the averages and the multipliers implement the scaling operation. The third layer accounts for the arithmetic computation based on Möbius functions.

A clear understanding of underlying arithmetic transform mechanisms will be possible in the next section. Once more the reader is asked to put this question aside for a while, allowing further developments to be derived.

To sum it up, at this point two major questions were accumulated: (i) How to handle with fractional indexes? and (ii) How could same formulae result in different spectra? Interestingly, both questions had the same answer.

The arithmetic transform algorithm could be summarized in four major steps:

1. Index generation, i.e., calculating the indexes of necessary samples ($m\frac{N}{k}$);

2. Fractional index samples handling, which requires interpolation;

3. Computation of averages: $S_k \triangleq \frac{1}{k}\sum_{m=0}^{k-1} v_{m\frac{N}{k}}$;

4. Computation of spectrum by Möbius Inversion Formula: $V_k = \sum_{l=1}^{\lfloor (N-1)/k \rfloor} \mu(l) S_{kl}$.

In the rest of this paper, the step two was addressed. In the sequel, a mathematical method, explaining the importance of the interpolation process in the arithmetic algorithms, was derived.

3.1 INTERPOLATION

Usual arithmetic theory deals with spectrum approximations via zero- or first-order interpolation [15, 26, 33]. In this section, it is shown that an interpolation process based on the known components (integer index samples) characterizes the definition of the fractional index components, v_r, $r \notin \mathbb{N}$. This analysis allows a more encompassing perception of the interpolation mechanisms and gives mathematical tools for establishing validation constraints to such interpolation process. In addition, brief comments on the trade-off between accuracy and computational cost required by interpolation process close the section.

3.1.1 IDEAL INTERPOLATION

What does a fractional index discrete signal component really mean? The value of v_r for a noninteger value r, $r \notin \mathbb{N}$, could be computed by

$$\begin{aligned} v_r &= \sum_{k=0}^{N-1} V_k \operatorname{cas}\left(\frac{2\pi k r}{N}\right) \\ &= \sum_{i=0}^{N-1} v_i \sum_{k=0}^{N-1} \operatorname{cas}\left(\frac{2\pi k i}{N}\right) \operatorname{cas}\left(\frac{2\pi k r}{N}\right). \end{aligned} \tag{50}$$

Defining the *Hartley weighting function* by

$$w_i(r) \triangleq \sum_{k=0}^{N-1} \operatorname{cas}\left(\frac{2\pi k i}{N}\right) \operatorname{cas}\left(\frac{2\pi k r}{N}\right), \tag{51}$$

the value of the signal at fractional indexes could be found utilizing an N-order interpolation expressed by:

$$v_r \triangleq \sum_{i=0}^{N-1} w_i(r) \cdot v_i. \tag{52}$$

It is clear that each transform kernel could be associated to a different weighting function. Consequently, a different interpolation process for each weighting function is required. In the arithmetic transform formalism, the difference from one transform to another resides in its interpolation process.

It can be shown that weighting functions make the Equation $\sum_{i=0}^{N-1} w_i(r) = 1$ to hold. If r is an integer number, then the orthogonality properties of $\operatorname{cas}(\cdot)$ function [6a] make $w_r(r) = 1$ and $w_i(r) = 0$ ($\forall i \neq r$). Therefore, no interpolation is needed.

After some trigonometrical manipulation, the interpolation weights for several kernels could be expressed by closed formulae. As stated before, there is a weighting function for each transform. Let $\operatorname{Sa}(\cdot)$ be the sampling function,

$$\operatorname{Sa}(x) \triangleq \begin{cases} \sin(x)/x, & x \neq 0, \\ 1, & x = 0. \end{cases} \tag{53}$$