H. J. Nussbaumer

Digital Filtering Using Complex Mersenne Transforms

Abstract: Complex Mersenne Transforms are defined in a ring of integers modulo a Mersenne or pseudo-Mersenne number and can be computed without multiplications. It is shown that under certain conditions, these transforms can be computed by means of fast transform algorithms and permit the evaluation of digital convolutions with better efficiency and accuracy than does the Fast Fourier Transform.

Introduction

With the rapid advances in large scale integration, a growing number of complex digital signal processing applications are becoming economically feasible. In most cases the bulk of the processing workload appears to consist of digital filter computation. Future progress in digital signal processing, either towards high speed, real time operation or increased sophistication, thus largely depends on increased efficiency in digital filter computation. This can be achieved not only by implementing improved filter circuits but also by using better computation algorithms, as will be discussed in this paper.

Rader [1] and Agarwal and Burrus [2, 3] have recently introduced Mersenne Transforms and Fermat Number Transforms. These two transforms have the circular convolution property and therefore can be used for evaluating digital filters in the same way as Discrete Fourier Transforms (DFT) [4, 5].

Mersenne Transforms and Fermat Number Transforms are very promising for digital filter computation because they can be calculated without multiplications. Their main drawback is a rigid relationship between transform length and word length, caused by the fact that all operations are performed in a finite ring with arithmetic carried out modulo an integer p. Another difficulty arises because it is not possible to achieve simultaneously optimum efficiency in reducing the number of operations and in implementing arithmetic operations. This is so because Fermat Number Transforms are amenable to a fast transform algorithm, and Mersenne Transforms are not, whereas arithmetic operations can be implemented more efficiently modulo a Mersenne number than modulo a Fermat number.

In this paper we consider complex transforms defined in the ring of integers modulo a Mersenne number. We show that these Complex Mersenne Transforms can be partly computed by a fast transform technique and have a maximum length up to four times that of conventional Mersenne Transforms. We discuss the use of such transforms for filtering complex signals and show that calculations in the first stages of the direct transforms can be performed on words of reduced length. We then extend these results to cover the case of pseudo-Mersenne numbers. We show that this leads to a definition of pseudo-Mersenne Transforms, which can be completely computed by means of a fast transform algorithm but in which the advantage of performing arithmetic operations modulo a Mersenne number is retained.

Complex Mersenne Transform

Let q be a prime and $p = 2^q - 1$. Mersenne numbers p are primes for $q = 2, 3, 5, 7, 13, 17, 19, 31, 61 \cdots$ [6].

The Mersenne Transform of an integer sequence $\{a_n\}$ having q terms is defined by

$$A_k = \left(\left(\sum_{n=0}^{q-1} a_n 2^{nk}\right)\right) \qquad k = 0, 1 \cdots q-1, \tag{1}$$

where any quantity enclosed by superfluous double parentheses is to be replaced by its value modulo p.

The Inverse Mersenne Transform is defined by:

$$a_m = \left(\left(R \sum_{k=0}^{q-1} A_k\, 2^{-mk}\right)\right) \qquad m = 0, 1 \cdots q-1, \tag{2}$$

where R is such that $((R \cdot q)) = 1$, all exponents and indices being taken modulo q and all operations being performed modulo p.

It can be demonstrated easily [1] that the Mersenne Transform satisfies the convolution theorem; that is to say, if $\{X_k\}$ is the Mersenne Transform of $\{x_n\}$, then with $Z_k = ((A_k \cdot X_k))$, the Inverse Mersenne Transform $\{z_m\}$ of $\{Z_k\}$ is given by

$$z_m = \left(\left(\sum_{n=0}^{q-1} a_n\, x_{m-n}\right)\right). \qquad (3)$$

If $\{a_n\}$ and $\{x_n\}$ are properly bounded [1], z_m is equal to the output of the ordinary cyclic convolution with

$$z_m = \sum_{n=0}^{q-1} a_n\, x_{m-n}. \qquad (4)$$

Under these conditions, digital filtering of real integer sequences can be performed by dividing the sequences into blocks, padding the blocks with zeros [4] to prevent folding, and aliasing and computing the cyclic convolutions by means of Mersenne Transforms. This provides a very efficient way of computing digital filters because calculations of Mersenne Transforms reduce to one's complement additions and circular shifts. Because Mersenne Transforms are evaluated without multiplications, computation of a time-invariant circular convolution having q points reduces to one multiplication per output sample, as opposed to q multiplications with direct calculation. This corresponds, in the case of a nonrecursive digital filter having N taps, to a number of multiplications per output sample that is small and essentially independent of the number of taps. Direct computation would require N multiplications in the general case and implementation by means of the Fast Fourier Transform (FFT) would require $K \log_2 N$ multiplications.

The main limitations of the Mersenne Transform approach are related to the fact that the number of transform terms q is a prime. This means that calculations of the transforms cannot be simplified by an FFT-type algorithm and that the number of transform terms is equal to the word size. These limitations can be slightly alleviated [1] by using a root -2 instead of 2 in (1) and (2). The maximum transform length then becomes $2q$. It is also possible to increase the maximum convolution size by resorting to multidimensional convolutions [1, 2, 7]. Unfortunately, this result is achieved at the expense of increased requirements for computation and storage.

We show now that by defining Complex Mersenne Transforms, it is possible to achieve higher computation efficiency while increasing both maximum transform length and convolution length.

In many instances of digital signal processing, digital filtering of complex signals is required. Modem equalizers for phase modulated signals are a good example of such applications [8]. In that case, a complex integer sequence $\{x_n = y_n + j\hat{y}_n\}$ is to be filtered by a complex sequence $\{a_n = b_n + j\hat{b}_n\}$, having N terms, to produce a complex output sequence $\{z_m = u_m + j\hat{u}_m\}$ with

$$z_m = \sum_{n=0}^{N-1} a_n x_{m-n} = \sum_{n=0}^{N-1} \left(b_n y_{m-n} - \hat{b}_n \hat{y}_{m-n}\right) + j \sum_{n=0}^{N-1} \left(\hat{b}_n y_{m-n} + b_n \hat{y}_{m-n}\right) \qquad j = \sqrt{-1}. \qquad (5)$$

The complex convolution (5) can be calculated by means of real transforms provided that computation on real and imaginary numbers is carried out in separate transforms. It is even possible, with Fermat Number Transforms, to reduce the number of multiplications by taking advantage of the real number representation of $j = \sqrt{-1}$ in a Fermat number system [9].

It is more natural, however, to compute complex convolutions with complex transforms such as the Fourier transform or Complex Number Theoretic Transforms [3, 10, 11]. In particular, complex transforms can be defined in a Mersenne ring. Along these lines, we have proposed a digital filter based on Complex Mersenne Transforms [12]. The author thanks the referees for bringing to his attention the independent work of Vegh and Leibowitz on the same subject [13].

In the following we restrict our discussion to Complex Mersenne Transforms that have simple roots and can be computed without multiplications. In a Mersenne ring, with $p = 2^q - 1$, 2 and -2 are respectively roots of unity of orders q and $2q$, corresponding to transforms of lengths q and $2q$, respectively. Since q is a prime, 2^d and -2^d are also roots of orders q and $2q$, provided d is not a multiple of q. This implies that $2j$ is a root of order $4q$ and $1 + j$ or $1 - j$ are roots of order $8q$. Higher-order complex roots do not have a simple structure and therefore will generally not be of practical interest.

Under these conditions, a Complex Mersenne Transform having $4q$ terms can be defined by

$$A_k = \left(\left(\sum_{n=0}^{4q-1} a_n j^{nk} 2^{nk}\right)\right) \qquad j = \sqrt{-1},\ k = 0, 1 \cdots 4q - 1. \qquad (6)$$

Because q has an inverse modulo p and the inverse of 4 is 2^{q-2}, $4q$ has an inverse R such that $((4qR)) = 1$ and the inverse transform of A_k is

$$a_m = \left(\left(R \sum_{k=0}^{4q-1} A_k j^{-mk} 2^{-mk}\right)\right) \qquad m = 0, 1 \cdots 4q - 1, \quad (7)$$

where all exponents and indices are taken modulo $4q$.

We demonstrate that this Complex Mersenne Transform satisfies the convolution theorem. Taking the complex transform $\{X_k\}$ of the complex sequence $\{x_l\}$, performing the term-by-term complex multiplications $X_k \cdot A_k$ and taking the inverse transform yields

$$z_m = \left(\left(\sum_{n=0}^{4q-1} \sum_{l=0}^{4q-1} a_n x_l R \left(\sum_{s=0}^{3} j^{rs} 2^{rs}\right) \sum_{k=0}^{q-1} 2^{4rk}\right)\right) \qquad (8)$$

with $l + n - m = r$ and k replaced by $4k + s$.

As in the case of conventional Mersenne Transforms, the sum $\Sigma_{k=0}^{q-1} 2^{4rk}$ is equal to zero for $(r)_{\mathrm{mod}\ q} \neq 0$ and equal to q for $(r)_{\mathrm{mod}\ q} = 0$. For $(r)_{\mathrm{mod}\ q} = 0$, $r = q\,t$ and $((\Sigma_{s=0}^{3} j^{rs} 2^{rs}))$ becomes $((\Sigma_{s=0}^{3} j^{qts}))$. Because q is an odd prime, this sum is different from zero only when t is a multiple of 4 and is equal to 4 when t is a multiple of 4. Under these conditions, the product of the two sums is different from zero only for $(r)_{\mathrm{mod}\ 4q} = 0$ and z_m becomes

$$z_m = \left(\left(\sum_{n=0}^{4q-1} a_n\, x_{(m-n)_{\mathrm{mod}\ 4q}}\right)\right), \qquad (9)$$

which shows that two complex sequences of length $4q$ can be cyclically convolved by means of Complex Mersenne Transforms modulo $p = 2^q - 1$. In such an approach, all arithmetic operations are performed as in normal, complex arithmetic with $j^2 = -1$ and real and imaginary parts treated separately modulo p.

Using a root $j + 1$ leads to a definition of a Complex Mersenne Transform having $8q$ terms with

$$A_k = \left(\left(\sum_{n=0}^{8q-1} a_n (j+1)^{nk}\right)\right) \qquad k = 0, 1 \cdots 8q - 1, \quad (10)$$

and, with R such that $((8qR)) = 1$, an inverse transform

$$a_m = \left(\left(R \sum_{k=0}^{8q-1} A_k (j+1)^{-mk}\right)\right)$$

$$m = 0, 1 \cdots 8q - 1, \qquad (11)$$

with all exponents and indices taken modulo $8q$. It can be demonstrated easily, by using the same development as given above, that this transform permits evaluation of convolutions of length $8q$ with word lengths of q bits.

Fast computation of Complex Mersenne Transforms

We have so far defined Complex Mersenne Transforms that can be computed without multiplications and have a length up to four times that of conventional Mersenne Transforms. The calculation of these transforms can be partly simplified by an FFT-type algorithm because the number of terms is no longer a prime.

Using either decimation in time or decimation in frequency [14] allows a decomposition into transforms having four q terms and transforms having eight q terms in the case of roots $2j$ and $j + 1$, respectively. These FFT decompositions reduce the number of real operations to $8q(q + 1)$ additions and $8q(q - 1)$ shifts for a complex transform having $4q$ points, and to $16\ q(q - 1) + 52q$ additions and $16(q - 1)^2 + 24(q - 1)$ shifts for a complex transform having $8q$ points. If we assume, for the sake of comparison, the existence of hypothetical real Mersenne Transforms having $4q$ points and $8q$ points, computing the transforms of the real and imaginary parts of a complex sequence would require $4q(4q - 1)$ additions and $(4q - 1)^2$ shifts for $4q$-point transforms and $8q(8q - 1)$ additions and $(8q - 1)^2$ shifts for $8q$-point transforms. This means that in the practical range of interest for q (where $q = 31$), substituting Complex Mersenne Transforms for conventional Mersenne Transforms results approximately in an eightfold reduction in the number of operations.

Computation of a complex convolution by means of Complex Mersenne Transforms is carried out as with DFT, with real and complex parts being evaluated modulo p separately. In order to avoid errors due to overflow, the amplitudes of real and imaginary outputs must be bounded to $(p - 1)/2$. This means that usually the word length of real and imaginary parts of input sequences $\{a_n\}$ and $\{x_n\}$ is less than half that of output sequences. In other words, all computations are carried out modulo p on q-bit words, yielding q-bit word outputs, and the input sequences are represented by words of less than $(q - 1)/2$ bits. It is possible to take advantage of the input word length limitations to further reduce the processing workload. This is done by selecting a fast transform decomposition such that the first two transform stages can be computed without multiplication by powers of two. In such an approach, a Complex Mersenne Transform having $4q$ points, with root $2j$, is decomposed into q four-point transforms with

$$A_k = \left(\left(\sum_{n=0}^{3} a_{qn} j^{qnk} + j^k 2^k \sum_{n=0}^{3} a_{qn+1} j^{qnk} + \cdots\right.\right.$$
$$\left.\left. + j^{(q-1)k}\, 2^{(q-1)k} \sum_{n=0}^{3} a_{qn+q-1} j^{qnk}\right)\right). \qquad (12)$$

The q four-point transforms corresponding to the first two stages of the decomposition require only multiplications by ±1 and $\pm j$ and can therefore be computed in normal, complex arithmetic, on words of length approximately half that of the final result.

If the two sequences $\{y_n\}$ and $\{a_n\}$ to be convolved are real, the full benefit of using Complex Mersenne Transforms can be retained by processing simultaneously two successive blocks of the sequence $\{y_n\}$ by means of the same Complex Mersenne Transforms. This is done by computing the complex convolution $\{z_m\}$ of the sequence $\{a_n\}$ with the auxiliary complex sequence $\{x_n = y_n + jy_{n+8q}\}$. The real part $\{u_m\}$ and the imaginary part $\{\hat{u}_m\}$ of $\{z_m\}$ yield respectively the convolutions of $\{y_n\}$ and of the next block $\{y_{n+8q}\}$ with $\{a_n\}$.

Complex pseudo-Mersenne Transforms

We have seen above that a conventional q-point Mersenne Transform could be extended to $8q$ points by oper-

Table 1 Maximum odd length and corresponding power-of-2 roots for real transforms modulo $p = 2^q - 1$ with q odd and p composite.

q	*Prime factorization of* $p = 2^q - 1$ $p_1, p_2 \cdots p_i$	*Prime factorization of* $p_i - 1$	*Max. odd length*	*Power of 2 roots*
15	$7 \cdot 31 \cdot 151$	$2 \cdot 3$ $2 \cdot 3 \cdot 5$ $2 \cdot 3 \cdot 5^2$	3	/
21	$7^2 \cdot 127 \cdot 337$	$2 \cdot 3$ $2 \cdot 3^2 \cdot 7$ $2^4 \cdot 3 \cdot 7$	3	/
23	$47 \cdot 178481$	$2 \cdot 23$ $2^4 \cdot 5 \cdot 23 \cdot 97$	23	2
25	$31 \cdot 601 \cdot 1801$	$2 \cdot 3 \cdot 5$ $2^3 \cdot 3 \cdot 5^2$ $2^3 \cdot 3^2 \cdot 5^2$	15	/
27	$7 \cdot 73 \cdot 262657$	$2 \cdot 3$ $2^3 \cdot 3^2$ $2^9 \cdot 3^3 \cdot 19$	3	/
29	$233 \cdot 1103 \cdot 2089$	$2^3 \cdot 29$ $2 \cdot 19 \cdot 29$ $2^3 \cdot 3^2 \cdot 29$	29	2
33	$7 \cdot 23 \cdot 89 \cdot 599479$	$2 \cdot 3$ $2 \cdot 11$ $2^3 \cdot 11$ $2 \cdot 3 \cdot 11 \cdot 31 \cdot 293$	/	/
35	$31 \cdot 71 \cdot 127 \cdot 122921$	$2 \cdot 3 \cdot 5$ $2 \cdot 5 \cdot 7$ $2 \cdot 3^2 \cdot 7$ $2^3 \cdot 5 \cdot 7 \cdot 439$	/	/
37	$223 \cdot 616318177$	$2 \cdot 3 \cdot 37$ $2^5 \cdot 3 \cdot 37 \cdot 167 \cdot 1039$	37	2
			111	/
39	$7 \cdot 79 \cdot 8191 \cdot 121369$	$2 \cdot 3$ $2 \cdot 3 \cdot 13$ $2 \cdot 3^2 \cdot 5 \cdot 7 \cdot 13$ $2^3 \cdot 3 \cdot 13 \cdot 389$	3	/
41	$13367 \cdot 164511353$	$2 \cdot 41 \cdot 163$ $2^3 \cdot 41 \cdot 59 \cdot 8501$	41	2
43	$431 \cdot 9719 \cdot 2099863$	$2 \cdot 5 \cdot 43$ $2 \cdot 43 \cdot 113$ $2 \cdot 43 \cdot 3^2 \cdot 2713$	43	2
45	$7 \cdot 31 \cdot 73 \cdot 151 \cdot 631 \cdot 23311$	$2 \cdot 3$ $2 \cdot 3 \cdot 5$ $2^3 \cdot 3^2$ $2 \cdot 3 \cdot 5^2$ $2 \cdot 3^2 \cdot 5 \cdot 7$ $2 \cdot 3^2 \cdot 5 \cdot 7 \cdot 37$	3	/
47	$2351 \cdot 4513 \cdot 13264529$	$2 \cdot 5^2 \cdot 47$ $2^5 \cdot 3 \cdot 47$ $2^4 \cdot 31 \cdot 47 \cdot 569$	47	2
49	$127 \cdot 4432676798593$	$2 \cdot 3^2 \cdot 7$ $2^7 \cdot 3^2 \cdot 7^2 \cdot 43 \cdot 337 \cdot 5419$	63	/

ating in the complex number field and replacing the root 2 of order q, by a complex root $(1 + j)$ of order $8q$. Up to now, the discussion has been restricted to Mersenne numbers, that is, to numbers $p = 2^q - 1$ such that q is a prime.

If p is not a prime, its prime factorization is given by

$$p = p_1^{d_1} \cdot p_2^{d_2} \cdots p_i^{d_i}. \tag{13}$$

An M-point real transform having the circular convolution property can be defined in the ring of integers modulo p provided M-point transforms can be defined separately in the fields $p_1, p_2 \cdots p_i$. This follows directly from the Chinese remainder theorem [3, 15], and leads to the condition for the existence of an M-point transform in the ring p that M must simultaneously divide $p_1 - 1$, $p_2 - 1, \cdots p_i - 1$. When p is a prime, the maximum length of the transform is $M = p - 1$. Transforms in a ring p, with p nonprime, are therefore proportionally shorter than transforms defined modulo a prime number. If p and q are composites with $q = q_1 \cdot q_2$ and q_1 prime, $2^{q_1} - 1$ divides p and the maximum transform length is governed by that possible for $2^{q_1} - 1$. This has led us to consider that the only transforms of interest in a ring modulo $2^q - 1$ are Mersenne transforms.

The situation, however, changes noticeably if we consider Complex Mersenne Transforms. In that case, the transform length becomes $8M$ if 2 does not divide M. This means that even if M is too small to yield a real transform of practical interest, useful Complex Mersenne Transforms of length $8M$ can still be defined.

Let us define these points precisely. Assuming we restrict our discussion to transforms that can be computed without multiplications, and therefore have roots 2^W of order M, we have by definition $2^{WM} \equiv 1$, with 2^{nW} taking M distinct values for $n = 0, 1 \cdots M - 1$. The exponents Wn are taken modulo MW with $(MW)_{\text{mod } q} = 0$. If we consider now a root $(j + 1)^W$, this root will be of order $8M$ in the ring p if $(j + 1)^{8MW} \equiv 1$ and if $(j + 1)^{Wn}$ takes $8M$ distinct values for n varying from 0 up to $8M - 1$. We first notice that as $(j + 1)^8 = 2^4$, $(j + 1)^{8MW} = 2^{4MW} \equiv 1$. If W is odd, $(j + 1)^{nW}$ takes eight distinct complex values for n varying from $n = 0$ to $n = 7$ and if $q = WM$ is odd, $(((j + 1)^{8nW})) = ((2^{4nW}))$ takes M distinct values for n varying from 0 to $M - 1$.

Under these conditions, $(j + 1)^W$ is a root of order $8M$ in the ring modulo $p = 2^q - 1$ if q, M, and W are odd. Moreover, the existence of an M-point real transform in the ring modulo p implies that M has an inverse, R modulo p. As the inverse of 8 modulo p is 2^{q-3}, $8M$ always has an inverse, $2^{q-3}R$. This means that, given an M-point real transform of root 2^w with p composite and q, w, M odd integers, we can define $8M$-point complex transforms in the ring modulo p with

$$A_k = \left(\left(\sum_{n=0}^{8M-1} a_n (j + 1)^{wnk}\right)\right)$$
$$j = \sqrt{-1}$$
$$k = 0, 1 \cdots 8M - 1, \tag{14}$$

$$a_m = \left(\left(2^{q-3} R \sum_{k=0}^{8M-1} A_k (j + 1)^{-wnk}\right)\right)$$
$$m = 0, 1 \cdots 8M - 1. \tag{15}$$

Table 2 Length and roots for real and complex transforms in the ring $(2^q - 1)/p_i^{d_i}$.

	Transform	*Real transform*		*Complex transform*		*Approximate word length*
q	*ring*	*Length*	*Root*	*Length*	*Root*	*Nb of bits*
15	$\frac{2^{15}-1}{7}$	5	2^3	40 $(2^3 \cdot 5)$	$2(j-1)$	12
21	$\frac{2^{21}-1}{7^2}$	7	2^3	56 $(2^3 \cdot 7)$	$2(j-1)$	15
25	$\frac{2^{25}-1}{31}$	25	2	200 $(2^3 \cdot 5^2)$	$j+1$	20
27	$\frac{2^{27}-1}{7.73}$	27	2	216 $(2^3 \cdot 3^3)$	$j+1$	18
35	$\frac{2^{35}-1}{31.127}$	35	2	280 $(2^3 \cdot 5 \cdot 7)$	$j+1$	23
35	$\frac{2^{35}-1}{127}$	5	2^7	40 $(2^3 \cdot 5)$	$2^3(1-j)$	28
35	$\frac{2^{35}-1}{31}$	7	2^5	56 $(2^3 \cdot 7)$	$-2^2(1+j)$	30
45	$\frac{2^{45}-1}{7.73}$	5	2^9	40 $(2^3 \cdot 5)$	$2^4(1+j)$	36
49	$\frac{2^{49}-1}{127}$	7	2^7	56 $(2^3 \cdot 7)$	$2^3(1-j)$	42
49	$\frac{2^{49}-1}{127}$	49	2	392 $(2^3 \cdot 7^2)$	$j+1$	42

In most practical digital filtering applications, the input signal samples are defined by a number of significant bits comprised between seven bits and 20 bits. This means that, in order to avoid overflow, the transforms must operate on word lengths approximately double that of input words, comprising between 15 bits and about 50 bits.

The various possibilities for p nonprime and odd q are listed in Table 1. The case of q prime ($q = 23, 29, 37, 41, 43, 47$) corresponds to conventional Mersenne Transforms. When q is not a prime, the corresponding transforms have a very short length and their roots are not powers of 2.

In order to achieve maximum effectiveness in computing convolutions by means of pseudo-Mersenne Transforms, it would be desirable to have relatively long transforms with a number of terms highly factorizable. This does not seem possible with transforms computed modulo $p = 2^q - 1$. We note, however, that when p is not a prime, with the prime factorization of p defined by (13), we can define transforms modulo $p/p_i^{d_i}$ having power-of-2 roots and such that the number of terms is large and highly factorizable. These transforms can be defined by

$$A_k = \left(\sum_{n=0}^{8M-1} a_n (j+1)^{wnk}\right)_{\text{mod } p/p_i^{d_i}}$$

$$k = 0, 1 \cdots 8M-1$$

$$j = \sqrt{-1}. \qquad (16)$$

Various possibilities for such transforms are listed in Table 2. It can be seen that the maximum number of terms is both large (40 to 392 terms) and highly factorable, thereby leading to efficient FFT-type computation with a minimum number of operations.

It would seem, however, that these advantages are offset by the fact that the various operations are performed modulo $(2^q - 1)/(p_i^{d_i})$. The corresponding arithmetic circuits are obviously much more complex than arithmetic circuits modulo $2^q - 1$.

This difficulty can be circumvented by noticing that as $p = p_i^{d_1} \cdot p_i^{d_2} \cdots p_i^{d_i}$, we can compute the convolution modulo $p = 2^q - 1$ as with conventional Mersenne Transforms and obtain the final result by performing a last operation modulo $p/p_i^{d_i}$ on the convolutions computed modulo p,

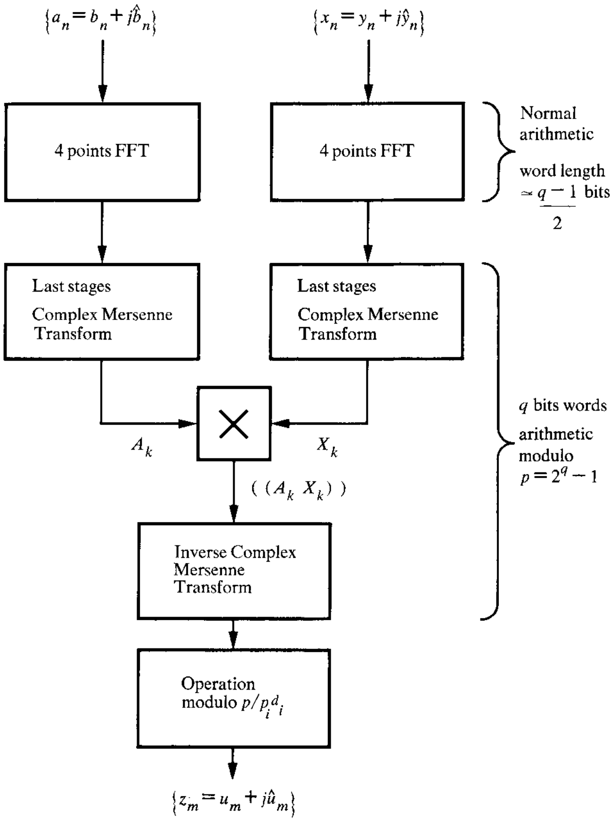
$$z_m \bmod p/p_i^{d_i} = (z_m \bmod p) \bmod p/p_i^{d_i}. \qquad (17)$$

By proceeding in this fashion, relatively long convolutions can be computed efficiently by means of FFT-type algorithms with all but the last operation performed with easily implemented arithmetic circuits operating modulo $(2^q - 1)$. This technique is similar to what was proposed in [1] to compute Fermat Number Transforms with Mersenne arithmetic. The only drawback of this approach is that all the operations modulo $(2^q - 1)$ are done on words longer than that of the final result. It can be seen however, from Table 2 that many transforms with a highly factorable number of terms can be defined for which $p_i^{d_i}$ is small compared to p, so that the penalty in word length increase incurred when operating modulo p instead of modulo $p/p_i^{d_i}$ is only of the order of 20 percent.

The most interesting transforms are those which have a large, highly factorable number of terms combined with a useful word length as close as possible to q. Among these, the 200-point, 56-point, and 392-point Complex pseudo-Mersenne Transforms (corresponding, respectively, to q equal to 25, 35, and 49) seem particularly well adapted for digital filtering applications. Taking as an example the case of transforms defined by $q = 25$, one can see from Table 1 that the maximum odd length for real transforms computed modulo $p = 2^{25} - 1$ is 15 terms and that the corresponding roots are not powers of two. By operating modulo $(2^{25} - 1)/31$, it is possible to define real transforms having power-of-two roots with a maximum odd length increased to 25 terms. The maximum length is then expanded to 200 terms by using complex roots. Such a transform can be computed very efficiently by means of an FFT-type algorithm with a three-stage radix 2 decomposition, followed by a two-stage radix 5 decomposition.

One limitation of conventional Mersenne Transforms is the rigid relationship between word length and transform length. In this respect, pseudo-Mersenne Transforms provide a significant improvement, because their maximum number of terms $M_{\max}$ is highly composite and any transform length submultiple of $M_{\max}$ can be selected. It is even possible to have several transforms of identical length and defined modulo integers $P_1, \cdots P_i$ that are relatively prime. The convolution can than be computed separately modulo $P_1, \cdots P_i$ and the final result obtained modulo $(P_1 \cdot P_2 \cdots P_i)$ by the Chinese remainder theorem. This approach could, for instance, be used to compute a 40-term convolution with an approximate word length of 32 bits by means of transforms defined by modulo $(2^{15} - 1)/7$ and $(2^{25} - 1)/31$.

The computation of a time-variant convolution by means of Complex pseudo-Mersenne Transforms is shown in Fig. 1. When a time-invariant convolution is to be evaluated, the various samples A_k of the transform of $\{a_n\}$ are constant. They are usually precomputed and stored in a memory. In this case, minor additional savings on multiplication cost can be achieved by storing A_k Modulo $p/p_i^{d_i}$ instead of A_k modulo p.



$\{a_n = b_n + j\hat{b}_n\}$
$\{x_n = y_n + j\hat{y}_n\}$
4 points FFT
4 points FFT
Normal arithmetic
word length $\simeq \frac{q-1}{2}$ bits
Last stages Complex Mersenne Transform
Last stages Complex Mersenne Transform
A_k
X_k
$((A_k\ X_k))$
q bits words arithmetic modulo $p = 2^q - 1$
Inverse Complex Mersenne Transform
Operation modulo $p/p_i^{d_i}$
$\{z_m = u_m + j\hat{u}_m\}$

Figure 1 Computation of complex circular convolutions by means of Complex pseudo-Mersenne Transforms.

Concluding remarks

We have discussed Complex Mersenne Transforms that can be computed without multiplications. These transforms are very promising for computing convolutions because they can be partly computed with FFT-type algorithms and some of the operations can be performed on words of reduced length. Complex Mersenne Transforms also have the advantage of permitting operation on transform length and convolution lengths up to four times longer than is possible with conventional Mersenne Transforms.

These results have then been extended to cover the case of transforms operating in a ring modulo a pseudo-Mersenne number or a submultiple of such a number. It has been shown that some of these transforms have a highly composite transform length and therefore can be computed with an efficient FFT-type algorithm.

Complex Mersenne Transforms can be used for implementing digital filters in the same manner as Discrete Fourier Transforms. They have the advantage over DFT of permitting exact calculations, without round-off errors, and of being computationally much more efficient because they can be calculated without multiplications.

References

1. C. M. Rader, "Discrete Convolutions via Mersenne Transforms," *IEEE Trans. Computers* **C-21,** 1269 (1972).
2. R. C. Agarwal and C. S. Burrus, "Fast Convolution using Fermat Number Transforms with Applications to Digital Filtering," *IEEE Trans. Acoustics, Speech and Signal Processing* **ASSP-22,** 87 (1974).

3. R. C. Agarwal and C. S. Burrus, "Number Theoretic Transforms to Implement Fast Digital Convolution," *Proc. IEEE* **63,** 550 (1975).
4. B. Gold, C. M. Rader, A. V. Oppenheim, and T. G. Stockham, *Digital Processing of Signals*, McGraw-Hill Book Company, Inc., New York, 1969, Ch. 7, pp. 203-213.
5. J. W. Cooley and J. W. Tukey, "An Algorithm for Machine Calculation of Complex Fourier Series," *Math. Comp.* **19,** 297 (1966).
6. D. E. Knuth, *The Art of Computer Programming: Seminumerical Algorithms*, Addison-Wesley Publishing, Co., Inc., Reading, MA, 1969, Vol. 2, p. 356.
7. R. C. Agarwal and C. S. Burrus, "Fast One-Dimensional Digital Convolution by Multidimensional Techniques," *IEEE Trans. Acoustics, Speech and Signal Processing* **ASSP-22,** 1 (1974).
8. R. D. Gitlin, E. Y. Ho, and J. E. Mazo, "Passband Equalization of Differentially Phase-Modulated Data Signals," *Bell System Tech. J.* **52,** 219 (1973).
9. H. J. Nussbaumer, "Complex Convolutions via Fermat Number Transforms," *IBM J. Res. Develop.* **20,** 282 (1976).
10. I. S. Reed and T. K. Truong, "The Use of Finite Fields to Compute Convolutions," *IEEE Trans. Inf. Theory* **IT-21,** 208 (1975).
11. I. S. Reed and T. K. Truong, "Complex Integer Convolutions Over a Direct Sum of Galois Fields," *IEEE Trans. Inf. Theory* **IT-21,** 657 (1975).
12. H. J. Nussbaumer, "Dispositif Générateur de Fonction de Convolution Discrète et Filtre Numérique Incorporant Ledit Dispositif," French Patent Application No. 7,512,557, April 1975.
13. L. M. Leibowitz, "Fast Convolution by Number Theoretic Transforms," NRL Report 7924, Naval Research Laboratory, Washington, D. C., September 1975.
14. Gold, Rader, Oppenheim, and Stockham, *op. cit.*, Ch. 6, pp. 174-196.
15. J. M. Pollard, "The Fast Fourier Transform in a finite Field," *Math. Comp.* **25,** 365 (1971).

Received December 22, 1975

The author is located at Compagnie IBM, France, Centre d'Etudes et Recherches, 06610 La Gaude, France.