International Journal of Research in Advent Technology, Vol.7, No.5, May 2019
E-ISSN: 2321-9637
Available online at www.ijrat.org

Number Theoretic Transforms for Fast Digital Computation

Salila Hegde[1], RohiniNagapadma[2]
*Department of ECE[1,2], NIE Institute of Technology[1,]*National*Institute of Engineering[2]*
Email: salilahegde@nieit.ac.in[1], rohini.nagapadma@nie.ac.in[2]

Abstract- This paper examines the properties of Number Theoretic Transforms over FFT. The aim of this study is to show that Number Theoretic Transforms (NTTs) can be really beneficial in terms of error free and faster computation. One and two dimensional NTTs are implemented in MATLAB and properties are verified and as an example convolution is implemented using Fermat number transform(FNT) which is a variant of NTT.A study of comparison of NTT to FFT proves that NTTs are really beneficial in terms of computational complexity and error free computation.

Index Terms-Number theoretic transforms,FNT,Convolution,FFT

1. INTRODUCTION

Number theoretic transforms are developed in early seventies based on number theory. They are the discrete Fourier transforms defined over finite fields and rings.[1 − 5].As these transforms are carried out in modulo arithmetic -modulo prime number, they exhibit many interesting properties which are useful for digital signal processing.
NTTs are computed in integer domain and hence they are found to be superior compared to already existing FFTs. Several techniques and algorithms have been developed for signal and image processing in the previous decades following the footsteps of Fast Fourier Transforms.

There are variants in NTTs, namely Fermat number transform, Mersenne number transform(MNT), Pseudo Fermat and Mersenne number transforms, Complex number transform(CNT) and the New number transform(NMNT) [6 − 7].Of these FNT and MNT are found to be attractive and applications developed for fast digital convolution, digital filtering, data encryption, compression, digital watermarking etc. [8 − 12]

This paper studies different properties of NTTs and it is shown that NTT can be employed for carrying out digital convolution faster. Performance comparison is carried out between FFT and NTT.
Section II gives, modulo arithmetic fundamentals, basic definition of NTTs and NTT parameters. Section III explains application of NTT to implement fast convolution.

Section IV illustrates convolution operation using NTT with a numerical example and gives comparison between FFT and NTT.

2. NUMBER THEORETIC TRANSFORMS

2.1. *Modulo arithmetic fundamentals*

As stated earlier NTTs are the transforms based on modular arithmetic, in this section a brief explanation of modulo arithmetic operations and parameters are discussed.
Let n be a positive integer, The set $\{0, 1 \ldots . n - 1\}$ is denoted as Z_M, Z_M is called as set of integers $mod\ M$. Z_M isalso a ring of integers $mod\ M$. If multiplicative inverse exists for all nonzero integers inZ_M then Z_M becomes field. [4-5].
Let x and y differ by two integers, and if they differ by some multiple of M they are said to be same.

$$x = y + kM \quad (1)$$

Where x and y are said to be congruent mod M and expressed as

$$x \equiv y(mod\ M) \quad (2)$$

Following are the modulo arithmetic operations permissible on ring/field Z_M .

(1)Addition: 6+ 14 = 3 (mod 17)

(2)Negation: -4 = -4+17 = 13 (mod 17)

(3)Subtraction:4 – 12 = 4 +(– 12) = 4+5

=9(mod 17)

(4)Multiplication: 4 x 13 =52 =1(mod 17)

(5)Multiplicative inverse: Multiplicative inverse of x exists iff x and M are relatively prime such that $x\ x\ x - 1 = 1(mod\ M)$. Multiplicative inverse of 4 is 13,4 x 13 $= 1(mod\ 17)$

(6)Division: x/yexists iff$y$ has an inverse.
$x/y = x\ x\ y - 1$.

For example: $12 / 4 = 12\,x\,13 = 156 = 3(mod\ 17)$

It is the nature of modular arithmetic that numbers do not have magnitude. Two numbers are not close in any manner.

2.2.*Number theoretic transforms*

The use of transform methods is found useful when an application permits the signal to be processed in blocks, The most useful transform Discrete Fourier transform (DFT) is found beneficial in the field of digital signal processing. A general transform has following expression with DFT structure

$$X(k) = \sum_{n-0}^{N-1} x(n)\alpha^{kn} \quad (3)$$

And it must have Circular convolution property (CCP). To have CCP it is found that α is a root of unity of order N such that

$$\alpha^N = 1(mod\ M) \quad (4)$$

and$N$ is lest positive integer. α is said to be root of unity of order N or α is primitive N^{th} root of unity. Then α^k is of order N/k if k/N,$\alpha^k$is of order N if N and k are relatively prime.

NTTs are discrete Fourier transforms, defined over finite fields or rings. The NTT of a signal x (n) and its inverse INTT are defined as follows:

$$X(k) = \sum_{n-0}^{N-1} x(n)\alpha^{n/k}\, mod\ M, k = 0,1,\dots N-1 \quad (5)$$

$$x(n) = \left(N^{-1}\sum_{k=0}^{N-1} X(k)\, a^{nk}\right) mod\ M, k = 0$$
$$n = 0,1,2,\dots . N-1 \quad (6)$$

Where $M$is some integer taken as a modulus.$\alpha$is a root of unity of order N(i.e.$\alpha^N = 1\ mod\ F$). N is transform length. It is desirable but not necessary that the modulus $M$is a prime number.
The 2D NTT and INTT are given by "Eq.7" and "Eq.8"..

$$x(k,1) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} X(m,n)a_1^{mk}\, a_2^{nl} mod\ F \quad (7)$$

$$x(m,n) = \left(N^{-1}\sum_{k=0}^{N-1} X(k,1)a1^{mk}\, a2^{nk}\right) mod\ F$$

Type equation here.

$$n = 0,1,2,\dots\dots\dots . N-1, m = 0,1,2\_\_\_M - 1 (8)$$

2.3. *Fermat number transforms*

In this section one of the NTT variant Fermat number transforms are explained. FNT is the most promising variant of NTT and found several applications [9-10]. Here the modulus chosen is called as Fermat number Ft

$$Ft = 2b+1, where\ b = 2t., t = 0,1,2\dots . \quad (9)$$

Only F_0 to $F_4$are prime and next numbers are composite.

The first few Fermat numbers are..

$F0 = 3$

$F1 = 5$

$F2 = 17$

$F3 = 257$

$F4 = 65537\ and\ so\ on.$

Since Fermat numbers up to $F_4$are all prime, FNT for any length $N = 2^m$ where$m \leq b$ is possible. Referring to table 1, we can say that integer 3 is an α of order $N = 2^{b\cdot}$. Integer 2 is anα of order $N = 2b$.If α is taken as 2 or a power of 2 all the powers of α will be powers of 2, and for these cases FNT can be computed very efficiently.

Table 1: NTT Parameters

N	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2N	1	2	4	8	16	15	13	9	1	2	4	8	16	15	13	9	1
3N	1	3	9	10	13	5	15	11	16	14	8	7	4	12	2	6	1
4N	1	4	16	13	1	4	16	13	1	4	16	13	1	4	16	13	1

3. NTTs FOR FASTER CONVOLUTIN

In this section we consider convolution of two sequences as an example of NTT application. Convolution is heart of any transform based application. In the previous year's several methods have been shown to implement convolution that vary in terms of computations required, the amount of storage needed and round

off error effects. Reducing the number of multiplications in convolution is often necessary as the multiplication operation is time taking and complex.

3.1.*Methodology*

DFT is employed to find convolution as

DFT $[h * x] = DFT[h]$ X DFT [n] and this implies that convolution can be found by

$$y(n) = IDFT\{ DFT[h] \; X \; DFT[n] \} \qquad (10)$$

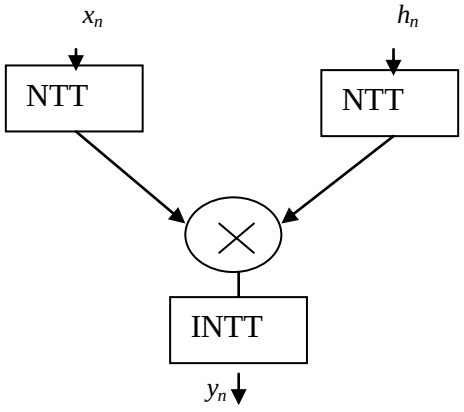
The disadvantage of this method is significant round off error, complex domain operations and considerable number of multiplications.

Number theoretic transforms are shown to possess property of CCP and have the similar structure of DFT as in "Eq.11".

$$X(k) = \left(\sum_{n=0}^{N-1} x(n)\, a^{n/k} \right) \qquad (11)$$

In Fourier transform the$e^{-2x/N}$is a complex number and transform operation is carried out in complex domain even if sequences are real. In NTT $e^{-2x/N}$term is replaced by an integer α of order N. When two sequences $x(n)$ and $h(n)$ are convolved using NTT , their output $y(n)$ is congruent to convolution of x and $h mod\ M$. By suitable choice of N, modulus M and value α it is possible to do the operation without any multiplications but using only shifts and additions. One such NTT is FNT as explained earlier. As the modulo arithmetic doesn't use approximations these transforms do not have round off errors.

So convolution of two sequences $x(n)$ and $h(n)$ is carried out using NTT as shown in fig 1.



4. RESULTS

4.1. *Illustration*

An implementation is carried out using MATLAB-13 software on 32 bit machine in windows 7 environment. Here circular convolution using FNT is explained taking a numerical example. In this we have taken care of treatment of negative values, finding transformation matrix t and inverse transform.

Let the two sequences be
x(n)= (1 , -2 , 3, 0) and *h(n)* = (1 1 0 0)
Let modulus be F_2= 17 and N =4;

From table 1 integer 2 is of order 8 and 2^2=4 is of order 4. So we choose α= 4.

Transformation matrix

$$T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 4 & 16 & 13 \\ 1 & 16 & 1 & 16 \\ 1 & 13 & 16 & 4 \end{bmatrix}$$

Inverse transformation matrix

$$T^{-1} = 13 \times \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 4 & 16 & 13 \\ 1 & 16 & 1 & 16 \\ 1 & 13 & 16 & 4 \end{bmatrix}$$

The forward transforms for x and h i.e. X and H are given by $X = T.x$ and $H = T.h$

$$X = T_{x=} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 4 & 16 & 13 \\ 1 & 16 & 1 & 16 \\ 1 & 13 & 16 & 4 \end{bmatrix} multiply \begin{bmatrix} 1 \\ 15 \\ 3 \\ 0 \end{bmatrix}$$

We get $X = (2 \quad 7 \quad 6 \quad 6)$

Similarly $H = T.h$ is given by $H = (2 \quad 5 \quad 0 \quad 14)$

and $Y = X.H = (4, 35, 0, 84) = (4, 1, 0, 16)\ mod\ 17$

$y(n)$is obtained by inverse transform on Y

$$y(n) = IFNT\,(Y) = Y\,.T^{-1}$$

$$= (1 \quad 3 \quad 1 \quad 16) = (1 \quad 3 \quad 1 \quad -1)\ mod\ 17$$

Integers are supposed to be in the range between$-8$and 8. Therefore in sequence $x(n), -2$ is represented as $-2 + 17 = 15$. In the answer 16 is to be represented as -1. i.e. $16 - 17 = -1$.This result matches with the answer obtained using DFT.

4.2. *Comparison with FFT*

NTT of a signal is carried out in the integer domain whereas DFT/FFT in complex domain.NTT coefficients do not have any physical meaning as the operations carried out modulo arithmetic. Because of modular arithmetic all the outputs are integers in the range 0 to $M-1$ where M is modulus, and no approximations take place. Hence NTTs do not have any round off errors as in FFT/DFT based algorithms. The special case of NTT, Fermat number transform uses $\alpha = 2$ or powers of 2 and hence all multiplications can be converted into shifts and additions. This is much simpler than complex domain multiplications; hence FNT is faster than FFT.

If two sequences $x(n)$ and $h(n)$ have $b1$ and $b2$ bit representations, and transform length is N, then convolution output $y(n)$ has maximum $(b1 + b2 + log_2 N)$ bit representation. In FFT every data sample has real and imaginary part, it requires two words one for real and one for imaginary. Hence storage and hardware requirements for both NTT and FFT remain almost same.

The convolution using FNT and DFT are implemented using Verilog HDL language on a DELL PC with Intel Pentium 64bit processor with 1.60GHz.. The comparative results are shown as in Table 2.

Table 2.Convolution timings in for length N real sequences.

N	$DFT\ (in\ ms)$	$FNT\ \ (in\ ms)$
16	4	0.69
32	8	1.5
64	17	3.3
128	31	7.4
256	60	16.6
512	113	40

In FNT method multiplication by power 2 are implemented as bit shift and add operations which are much simpler compared to complex multiplications required in DFT method. This offers significant savings in computation time and hence NTT algorithms are faster than FFT.

4.3. *Arithmetic issues*

In FNT algorithms all operations are done using modulo arithmetic over modulus Ft where $Ft = 2^b + 1$and $b = 2^t$. So all the coefficients in the range 0 to 2^b can be represented without any ambiguity. Negative numbers are represented in their two's complement form. But if the coefficient is 2^bit has to be approximated as 0 or 2^bwhich introduces error in the output. Otherwise one extra bit is required to represent 2^b, whichrequires more complicated hardware.

5. CONCLUSION

In this paper the properties of new transforms, NTTs are studied NTTs and its variants. NTTs have attractive features as the modulo arithmetic is carried out on integers and all representations are in integers. Because of this they do not have round off errors as DFT. They need very less or no multiplications as they are replaced by shift and additions by employing suitable NTT parameters as in the case of FNT.. This paper implements convolution using FNT which is a variant of NTT and gives faster result compared to convolution using FFT. So it is hereby forecasted that NTTs can be used for applications like faster digital filtering of images as filtering involves convolution of image with the filter mask. Further investigation and study of these transforms for the use in the field of image compression and encryption is also worth doing. The major disadvantage of NTT is the stringent relation between modulus M and sequence length N which can be overcome by block processing.

Acknowledgments

The authors would like to thank Department of ECE, NIE ,and Mysuru for providing resources to carry out this research study.

REFERENCES

[1] B.Gold; C.M.Rader(1969): Digital Processing of Signals,McGraw-Hill.
[2] R. Blahut (1985): Fast algorithms for digital signal processing, Addison-Wesley Publishing Company.
[3] Pollard J.M.(1971): The fast Fourier transform in a finite field, Math.Comput., 25, pp. 365-374.
D.burton(1980): Elementory number theory.
[4] P.J.Nicholson (1971):Algebraic theory of finite Fouriertransforms,J.Comput.Systems,Scl,vol.5,pp .524-547.
[5] M Bhattacharya: R Creutzburg: J Astol (2004): Some historical notes on number theoretic transform,Proc. 2004 Int. TICS Workshop on Spectral Methods and Multirate Signal
[6] B.Gold ; C.M.Rader(1969): Digital Processing of Signals.McGraw-Hill.
[7] S. Gudvangen; Hgskulen i Buskerud (1999):Practical applications of number theoretic transforms,Conference NORSIG'99,Norway

[8] Agarwal R.C;Burrus C.S.(1974): Fast convolution using Fermat number transform with application to digital filtering, ibid., ASSP-22, pp. 87-97.
[9] Tuukka Toivonen;Janne Heikkila (2006):Video filtering with fermat number theoretic transforms using residue number system, IEEE transactions on circuits and systems for video technology,vol.16,No1.
[10] Salila Hegde:Rohini Nagapadma (2017):Lossless compression scheme for regular images ,IJCA, volume 158(9),pp.7-12
[11] Hideaki Tamori (2009):Asymmetric fragile watermarking using a Number theoretic transform, IEICE Trans.fundamentas,vol. E92-A,No.3