Performance Case study of Grigoryan FFT over Cooley-Tukey FFT using TMS DSP Processors

Narayanam Ranganadh[1], Muni Guravaiah P[2]
Department of Electronics & Communications Engineering Bharat Institute of Engineering & Technology
Mangalpally (V), Ibrahimpatnam (M) Hyderabad, A.P., India – 501 510

Abstract

Frequency analysis plays vital role in the applications like cryptanalysis, steganalysis [6], system identification, controller tuning, speech recognition, noise filters, etc. Discrete Fourier Transform (DFT) is a principal mathematical method for the frequency analysis. The way of splitting the DFT gives out various fast algorithms. In this paper, we present the implementation of two fast algorithms for the DFT for evaluating their performance. One of them is the popular radix-2 Cooley-Tukey fast Fourier transform algorithm (FFT) [1] and the other one is the Grigoryan FFT based on the splitting by the paired transform [2]. We evaluate the performance of these algorithms by implementing them on the TMS320C6748 and TMS320C5416 DSPs. We developed C programming for these DSP processors. Finally we show that the paired-transform based algorithm of the FFT is faster than the radix-2 FFT, consequently it is useful for higher sampling rates. Working at higher data rates is a challenge in the applications of Digital Signal Processing.

Index Terms

frequency analysis, fast algorithms, DFT, FFT, paired transforms.

1. Introduction

In the recent decades, fast orthogonal transforms have been widely used in areas of data compression, pattern recognition and image reconstruction, interpolation, linear filtering, and spectral analysis. The suitability of unitary transforms in each of the above applications depends on the properties of their basis functions as well as on the existence of fast algorithms, including parallel ones. Since the introduction of the Fast Fourier Transform (FFT), Fourier analysis has become one of the most frequently used tool in signal/image processing and communication systems [6]; The main problem when calculating the transform relates to construction of the decomposition, namely, the transition to the short DFT's with minimal computational complexity [6]. The computation of unitary transforms are complicated and time consuming process. Since the decomposition of the DFT is not unique, it is natural to ask how to manage splittings and how to obtain the fastest algorithm of the DFT. The difference between the lower bound of arithmetical operations and the complexity of fast transform algorithms shows that it is possible to obtain FFT algorithms of various speeds [2]. One approach is to design efficient manageable split algorithms. Indeed, many algorithms make different assumptions about the transform length. The signal/image processing related to engineering research becomes increasingly dependent on the development and implementation of the algorithms of orthogonal or non-orthogonal transforms and convolution operations in modern computer systems. The increasing importance of processing large vectors and parallel computing in many scientific and engineering applications require new ideas for designing super-efficient algorithms of the transforms and their implementations [2].

In this paper we present the implementation techniques and their results for two different fast DFT algorithms. The difference between the algorithm development lies in the way the two algorithms use the splitting of the DFT. The two fast algorithms considered are Radix-2 Cooley-Tukey FFT and Grigoryan FFT [2] algorithms. The implementation of the algorithms is done both on the TMS320C5416 [4] and also on theTMS320C6748 [3] DSPs (fixed point operations). The performance of the two algorithms is compared in terms of their sampling rates. Section II presents the paired transform decomposition used in paired transform in the development of Grigoryan FFT. Section III presents the Implementation Techniques, Section IV presents implementation results. Finally with the Section V we conclude the work and put forward some suggestions for further sampling rate improvements.

2. Decomposition algorithm of the fast DFT using paired transform

In this algorithm the decomposition of the DFT is done by using the *paired transform[2]*. Let { $x(n)$ }, n = 0:(N-1) be an input signal, N>1. Then the DFT of the input sequence $x(n)$ is

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \text{ , k =0:(N-1)} \qquad (1)$$

which is in matrix form

$$X = [F_N]x \qquad (2)$$

where X(k) and $x(n)$ are column vectors, the matrix F_N = $\left\| W_N^{nk} \right\|_{n,k=(0:N-1)}$, is a permutation of X.

$$F_N = diag\{[F_{N1}],[F_{N2}],.....,[F_{Nk}]\}[\overline{W}][\chi'_N]$$

(3)

which shows the applying transform is decomposed into short transforms F_{Ni}, i = 1: k. Let S_F be the domain of the transform F the set of sequences f over which F is defined. Let (D; σ) be a class of unitary transforms revealed by a partition σ. For any transform $F \in$ (D; σ), the computation is performed by using paired transform in this particular algorithm. To denote this type of transform, we introduce *"paired functions [2]."*

Let $p, t \in$ period N, and let

$$\chi_{p,t}(n) = \begin{cases} 1; np = t(\mathrm{mod}\, N); \\ 0; otherwise. \end{cases} \text{n=0:(N-1)} \qquad (4)$$

Let L be a non trivial factor of the number N, and W_L = $e^{2\Pi/L}$, then the complex function

$$\chi'_{p,t} = \chi'_{p,t;L} = \sum_{k=0}^{L-1} W_L^k \chi_{p,t+kN/l} \qquad (5)$$

t = 0: $\left(N/L-1\right)$, p€ to the period 0: N-1

is called L-paired function [2]. Basing on this paired functions the complete system of paired functions can be constructed. The totality of the paired functions in the case of N = 2^r is

$$\{\{\chi_{2^n,2^n t}; t=0:(2^{r-n-1}-1), n=0:(r-1)\},1\}$$

(6)

Now considering the case of N = 2^r N_1, where N_1 is odd, r ≥ 1 for the application of the paired transform.

a) The totality of the partitions is

$$\sigma' = (T'_{1;2}, T'_{2;2}, T'_{4;2}, ..., T'_{2^r;2})$$

b) The splitting of F_N by the partition σ' is

{ $F_{N/2}, F_{N/4}, ..., F_{N/2^r}, F_{N1}$ }

c) The matrix of the transform can be represented as

$$[F_N] = \left(\bigoplus_{n=1}^{r} [F_{Ln}] \right) [\overline{W}][\chi'_n]$$

$$L_n = N/2^{n+1}, L_r = N_1.$$

Where the [$\overline{W}$] is diagonal matrix of the twiddle factors. The steps involved in finding the DFT using the paired transform are given below:

1) Perform the L-paired transform g = $\chi'_N(x)$ over the input x
2) Compose r vectors by dividing the set of outputs so that the first L^{r-1} elements $g_1, g_2, ..., g_{L^{r-1}}$ compose the first vector X_1, the next L^{r-2} elements $g_{Lr-1}+1, ..., g_{Lr-1+Lr-2}$ compose the second vector X_2, etc.
3) Calculate the new vectors Y_k, k=1:(r-1) by multiplying element-wise the vectors X_k by the corresponding turned factors $1, W_t, W_t^2, ..., W_t^{t/L-1}, (t = L^{r-k})$,(t=Lr-k). Take Y_r=X_r
4) Perform the L_{r-k}-point DFT's over Y_k, k=1: r
5) Make the permutation of outputs, if needed.

3. Implementation techniques

In the Fast Fourier Transform process the butterfly operation is the main unit on which the speed of the whole process of the FFT depends. So the faster the butterfly operation, the faster the FFT process. Coming to the implementation of the paired transform based DFT algorithm, there is no complete butterfly operation, as that in case of Cooley-Tukey algorithm. According to the mathematical description given in the Section II, the arithmetic unit is divided into two parts, addition part and multiplication part. This makes the main difference between the two algorithms, which causes the process of the DFT completes earlier than the Cooley-Tukey FFT algorithm. The addition part of the algorithm for 8-point transform is shown in Fig 1. The implementations are done for the 16-point, 32, 64, 128, 256, 512 and 1024 and point transforms for both Cooley-Tukey & Grigoryan FFTs. The paired Transform based Grigoryan FFT is working much higher sampling rates, in its implementation algorithms in C programming for both the two DSPs: TMS320C5416 & TMS320C6748 (fixed point). Grigoryan FFT is very efficient in case of higher sampling rate applications.

4. The implementation results

Results obtained on TMS320C6748 and TMS320C5416 digital signal processors: The software used for implementing on the DSP is Texas instruments code composer studio [5]. We used C programming language for implementation. Table 1, Table 2, Table 3 and Table 4 show the implementation results and the comparison between the two algorithms on the DSP processors.

Table 1: Performance comparison of the two algorithms on DSP processor TMS320C6748 (fixed point) processor.

No. of Samples	Radix-2 FFT	Paired transform based FFT	%improvement
16	100630	85601	18
32	274613	217498	26
64	388029	294799	32
128	1530693	1134670	35
256	1750023	1578720	48
512	4762375	2983408	60
1024	15123833	8987561	68

Table 2: Table showing the sampling rate of both the algorithms (starting form N = 16 to N = 1024) for TMS320C6748.

Number of CCs		Sample rate (MHz)	
radix-2 FFT	paired transform	radix-2 FFT	paired transform
100630	85601	19.75	21.76
274613	217498	12.63	18.41
488029	294799	15.08	19.35
1530693	1134670	10.29	14.54
1750023	1578720	12.923	18.81
4762375	2983408	13.56	19.40
15123833	8987561	10.45	14.44

Tables 1, 2 show the implementation results for the TMS320C6748 DSP and Tables 3, 4 show the implementation results for TMS320C5416. Results clearly show that the Grigoryan FFT is much faster than Cooley-Tukey FFT. Table 1, 3 show that going to higher transform lengths, paired algorithm gives the higher percentage improvement over the Cooley-Tukey algorithm. Table 2, 4 show the time required and the sampling rates that the two algorithms can be operated at, which shows that Grigoryan FFT is working at higher sampling rates than Cooley-Tukey FFT. The graphs in Fig 2 and Fig 3 show the graphical comparison of the number of Clock cycles taken by both the algorithms on both the DSPs. These graphs show that Grigoryan FFT takes less number of clock cycles than Cooley-Tukey FFT.

Table 3: Performance comparison of the two algorithms on DSP processor TMS320C5416 (fixed point) processor.

No. of Samples	Radix-2 FFT	Paired transform based FFT	%improvement
16	101580	92346	10
32	275363	222067	24
64	489229	391383	25
128	1542693	1224360	26
256	2580026	1686276	53
512	4807375	3004609	60
1024	15683833	9220491	70

Table 4: Table showing the sampling rate of both the algorithms (starting form N = 16 to N = 1024) for TMS320C5416.

Number of CCs		Sample rate (MHz)	
radix-2 FFT	paired transform	radix-2 FFT	paired transform
101580	92346	15.75	17.33
275363	222067	11.63	14.41
489229	391383	13.08	16.35
1542693	1224360	8.29	10.45
2580002	1686276	9.923	15.18
4807375	3004609	10.65	17.04
15683833	9220491	6.55	11.11

Form these results we can see that paired transform is always faster than the radix-2 algorithm. Thus paired-transform based algorithm can be used for higher sampling rate applications. In military applications, while doing the process, only some of the DFT coefficients are needed at a time. For this type of applications paired transform can be used as it generates some of the coefficients earlier, and also it is very fast.

5. Conclusion

In this paper we have shown that both on TMS320C6748 and TMS320C516 DSPs and the paired transform based Grigoryan FFT algorithm is faster and can be used at higher sampling rates than the Cooley-Tukey FFT. We are sure that this Grigoryan FFT is highly efficient than Cooley-Tukey FFT for some particular military applications.

1) In the implementations on the DSP if the MAC engines are used explicitly, then there may be a possibility of better comparison between the algorithms. Also one can see some more speed improvement in the DFT processes.
2) We would like to implement & apply this Grigoryan FFT to real time applications, even by using MAC engines, and would like to Standardize this Grigoryan FFT as an industrial standard.

Acknowledgment

Mr. Narayanam Ranganadh thanks Dr. Parimal A. Patel, Dr. Artyom M. Grigoryan of The University of Texas at San Antonio; regarding their valuable guidance to do the first part of this research series as his master’s thesis. Their support and help will always be with Mr. Narayanam to make Grigoryan FFT as a standarad FFT as their patent in the industrial market.

References

[1] James W. Cooley and John W. Tukey, “An algorithm for the machine calculation of complex Fourier Series”, Math. comput. 19, 297-301 (1965).

[2] Artyom M. Grigoryan and Sos S. Agaian, “Split Manageable Efficient Algorithm for Fourier and Hadamard transforms”, IEEE Transactions on Signal Processing, Volume: 48, Issue: 1, Pages: 172 – 183, Jan.2000.

[3] TMS320C6748 development kit & data sheet.

[4] TMS320C5416 development kit & data sheet.

[5] CC Studio getting started guide, for C/C++, IDE.

[6] The scientist and engineer’s guide to Digital Signal Processing, Dr. Steven W. Smith.

[7] N. Ranganadh, P. Patel, and A.M. Grigoryan, "Implementation of the DFT using radix-2 and paired transform algorithms," 17th International Conference on Computer Applications in Industry and Engineering, CAINE-2004, Orlando, Florida USA, Nov. 17-19, 2004.

[8] N. Ranganadh, P. Patel, and A.M. Grigoryan, “performances of Texas instruments DSP and Xilinx FPGAs for Cooley-Tukey and Grigoryan FFT algorithms”, Wolters Kluwer and Medknow International Journal of Engineering and Technology, Jul-Dec 2011,Vol 1,Issue 2.

[9] N. Ranganadh, P. Patel, and A.M. Grigoryan, “Case study of Grigoryan FFT onto FPGAs and DSPs” , IEEE proceedings, IEEE ICECT 2012, 2012 4th International Conference on Electronics Computer Technology (ICECT 2012), kanya kumari – 2012.

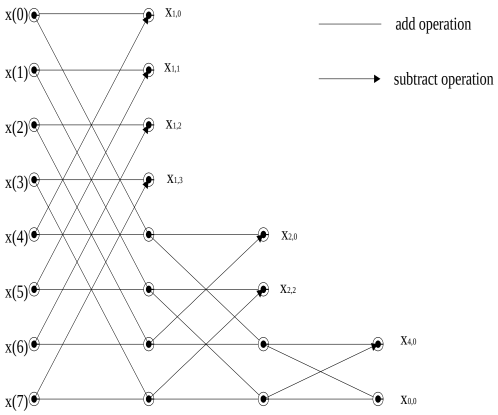

Fig 1: Showing the addition part of the 8-point paired transform based DFT

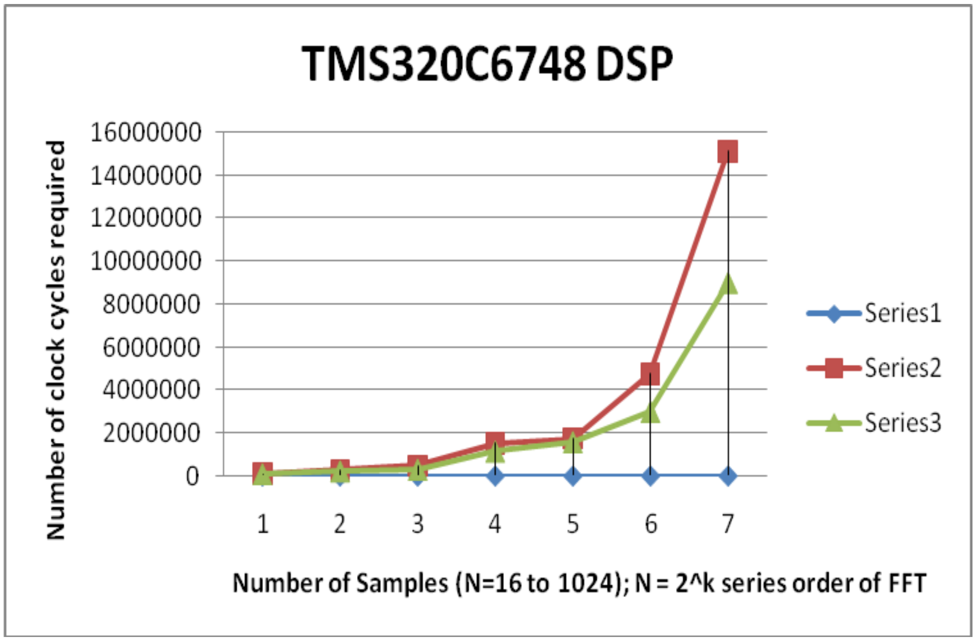

Fig 2: For TMS320C6748 DSP. On Y-axis the number of clock cycles taken to process the FFT of size N = 16,32,64,128,256,512,1024. N on X-axis. Series 2,3 are for the number of clock cycles taken for the Cooley-Tukey FFT and Grigoryan FFT respectively, it is clear that Grigoryan FFT is going faster than Cooley-Tukey FFT

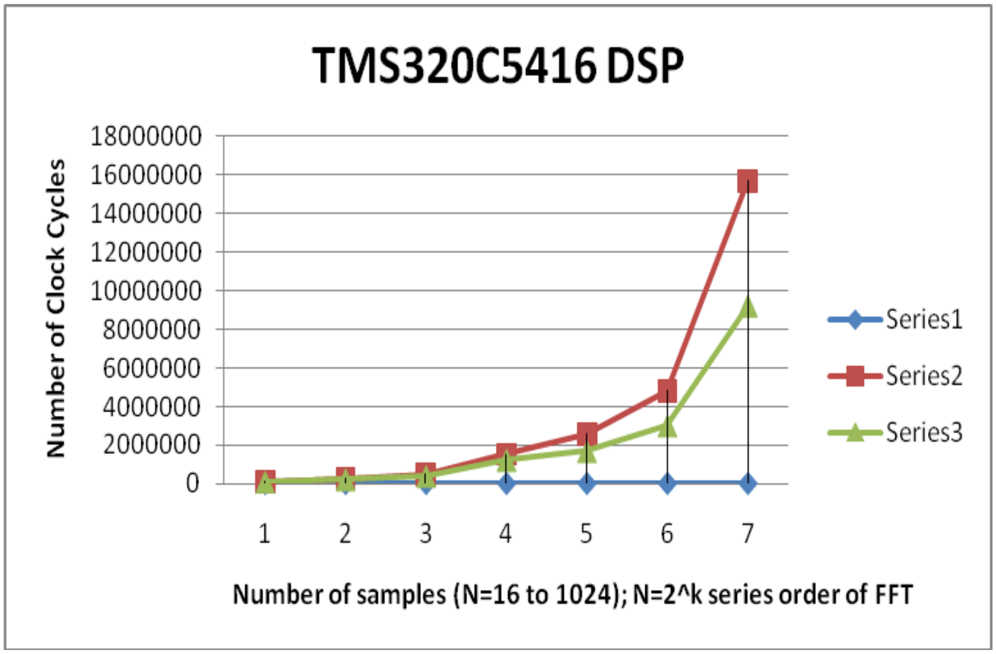

Fig 3: For TMS320C5416 DSP. On Y-axis the number of clock cycles taken to process the FFT of size N = 16,32,64,128,256,512,1024. N on X-axis. Series 2, 3 are for the number of clock cycles taken for the Cooley-Tukey FFT and Grigoryan FFT respectively, it is clear that Grigoryan FFT is taking less clock cycles than Cooley-Tukey FFT

Ranganadh Narayanam: Mr. Ranganadh Narayanam is an Assistant professor in the department of Electronics & Communications Engineering in Bharat Institute of Engineering & Technology (BIET). This research is continuation of the research done in the Univeristy of Texas at San Antonio under the guidance of Dr. Parimal A. Patel, Dr. Artyom M. Grigoryan, as my master's thesis. This current research was partly funded by BIET. Mr.Narayanam, a research student in the area of "Brain Stem Speech Evoked Potentials" under the guidance of Dr. Hilmi Dajnai of University of Ottawa,Canada. He was also a research student in The University of Texas at San Antonio under Dr. Parimal A Patel,Dr. Artyom M. Grigoryan, Dr Sos Again, Dr. CJ Qian,. in the areas of signal processing and digital systems, control systems. He worked in the area of Brain Imaging in University of California Berkeley. Mr. Narayanam is having around 5+2 years of full time teaching & research experiences respectively; and more than 5 years of entry level research experience and more than 10 publications. Mr. Narayanam's research interests include neurological Signal & Image processing, DSP software & Hardware design and implementations, neurotechnologies.

P. Muni Guravaiah: Mr. P. Muni Guravaiah is an Associate Professor & Admin. Incharge of Department of Electronics & Communications Engineering in BIET. He is currently pursuing Ph.D. in Jawaharlal Nehru Technological University Hyderabad, in the area of VLSI Design. His research interests include DSP; VLSI Design. He is having around 12 years of accumulated Teaching and Research experiences.