Design and implementation of polynomial matrix multiplication by using cyclic convolution technique

Parakala Sujitha
M.tech student
VLSI System Design
Kshatriya College of engineering
Chepoor village, Armoor, Nizamabad,
Telangana
sujitha.parakala21@gmail.com

Gopi Kondra
Assistant Professor, ECE dept
Vlsi & Embeded System
Kshatriya College of engineering
Chepoor village, Armoor, Nizamabad,
Telangana
Kopi707@gmail.com

Abstract--**This work presents a mathematical framework for the development of efficient algorithms for cyclic convolution computations. The framework is based on the Chinese Reminder Theorem (CRT) .In particularly, this work focuses on the arithmetic complexity of a matrix-vector product when this product represents a CC computational operation or it represents a polynomial multiplication modulo the polynomial z N-1, where N represents the maximum length of each polynomial factor and it is set to be a power of 2. The proposed algorithms are compared against existing algorithms developed making use of the CRT and it is shown that these proposed algorithms exhibit an advantage in computational efficiency. They are also compared against other algorithms that make use of the Fast Fourier Transform (FFT) to perform indirect CC operations, thus, demonstrating some of the advantages of the proposed development framework.**

INTRODUCTION

Direct computation of a matrix-vector product takes 2 N complex multiplications; however, by exploiting the special structure of a circulant matrix, the computational effort could be substantially decreased for large matrices. One approach is to use the fast Fourier transform (FFT), which makes possible to compute a matrix-vector product in (N)log (N) 2 complex multiplications for a matrix of order N . In this work algorithms are developed by means of a decimation in time approach, and the use of the roots of the unity (factoring the polynomial z N -1 in the complex field). They require only N multiplications, with some advantages over FFT such as memory use and addressing techniques.

Vector multiplications:

In mathematics, Vector multiplication refers to one of several techniques for the multiplication of two (or more) vectors with themselves. It may concern any of the following articles: Dot product — also known as the "scalar product", an operation that takes two vectors and returns a scalar quantity.

CHARACTERISTIC POLYNOMIALS

So far our discussion has dealt only theoretically with the existence of eigen values of an operator T ∈ L(V). From a practical standpoint, it is much more convenient to deal with the matrix representation of an operator. Recall that the definition of an eigen value λ∈ *F* and eigen vector $\mathrm{v} = \sum \mathrm{v}_i \mathrm{e}_i$ of a matrix $\mathrm{A} = (\mathrm{a}_{ij}) \in \mathrm{M}_n(\mathcal{F})$ is given in terms of components by $\sum_j \mathrm{a}_{ij}\mathrm{v}_j = \lambda \mathrm{v}_i$ for each i = 1, . . . , n. This may be written in the form

$$\sum_{j=1}^{n} a_{ij} v_j = \lambda \sum_{j=1}^{n} \delta_{ij} v_j$$

Alternatively it can be written as

$$\sum_{j=1}^{n} (\lambda\delta_{ij} - a_{ij}) v_j = 0$$

In matrix notation, this is

$$(\lambda \mathrm{I} - \mathrm{A})\mathrm{v} = 0$$

SINGULAR VALUE DECOMPOSITION

An efficient implementation for calculating the SVD of a constant matrix can be obtained for polynomial matrices. $\mathbf{A}(z) \in \mathbb{C}^{p\times q}$, a PSVD of $\mathbf{A}(z)$ is

$$\mathbf{A}(z) = \mathbf{U}(z)\mathbf{D}(z)\mathbf{V}^H(z^{-*})$$

Where

$\mathbf{U}(z) \in \mathbb{C}^{p\times p}$, $\mathbf{V}(z) \in \mathbb{C}^{q\times q}$ are approximately para unitary matrices and $\mathbf{D}(z) \in \mathbb{C}^{p\times q}$ is an approximately diagonal matrix [4] whose diagonal elements are called the singular values of A(z). As in the QRD case, the intuition is that the polynomial matrix A(z) is decomposed into its SVD over all frequencies. Note that in this definition there is no assumption of ordering of the singular values, as compared to the ordinary SVD .

A singular value and pair of singular vectors of a square or rectangular matrix Aare a non negative scalar σ and two nonzero vectors u and v so that

Av = σu,
AHu = σv.

The term “singular value” relates to the distance between a matrix and the set of singular matrices. Singular values play an important role where the matrix is a transformation from one vector space to a different vector space, possibly with a different dimension. Systems of over- or under determined algebraic equations are the primary examples. The definitions of eigenvectors and singular vectors do not specify their normalization.

An eigenvector *x*, or a pair of singular vectors *u* and *v*, can be scaled byany nonzero factor without changing any other important properties. Eigenvectorsof symmetric matrices are usually normalized to have Euclidean length equal to one, ‖ *x*‖ 2 = 1. On the other hand, the eigenvectors of nonsymmetrical matrices often havedifferent normalizations in different contexts. Singular vectors are almost alwaysnormalized to have Euclidean length equal to one, ‖ *u*‖ 2 = ‖ *v*‖ 2 = 1. You can stillmultiply eigenvectors, or pairs of singular vectors, by −1 without changing theirlengths.

Polynomial matrix multiplication:

Suppose we are given two polynomials:

p(x) = a0 + a1x + · · · + an−1xn−1,

q(x) = b0 + b1x + · · · + bn−1xn−1.

Their product is defined by

$$p(x) \cdot q(x) = c_0 + c_1 x + \cdots c_{2n-2} x$$

$$c_i = \sum_{\max\{0,i-(n-1)\} \le k \le \min\{i,n-1\}} a_k$$

In computing the product polynomial, every a_i is multiplied with every b_j , for 0 ≤ i, j ≤ n −1. So there are at most n^2 multiplications, given that some of the coefficients may be zero. Obtaining every c_i involves one fewer additions than multiplications. So there are at most n^2−2n+1 addition is involved. In short, the number of arithmetic operations is O(n^2). This is hardly efficient. But can we obtain the product more efficiently, by the use of a well-known method called fast Fourier transform, or simply, FFT.

Use FFT to compute the convolution of two vectors

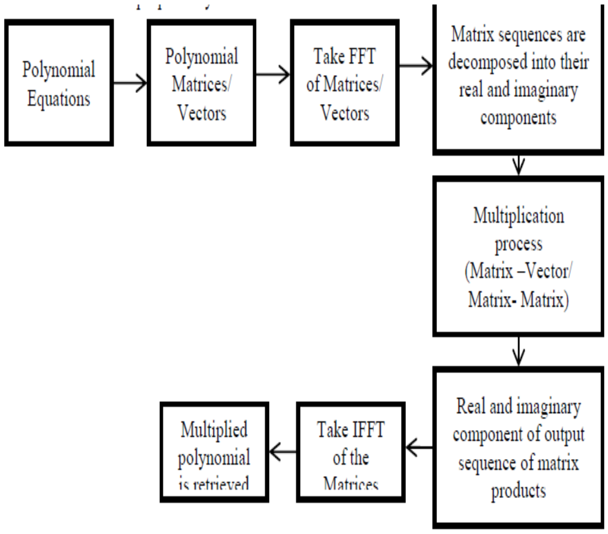
a = (a_0, . . . , a_{n-1}) and b = (b_0, . . . , b_{n-1}),

Which is defined as a vector c = (c_0, . . . , c_{n-1})

Where

$$c_j = \sum_{k=0}^{j} a_k b_{j-k}, \qquad j = 0, \ldots, n-1$$

.


Polynomial Equations
Polynomial Matrices/ Vectors
Take FFT of Matrices/ Vectors
Matrix sequences are decomposed into their real and imaginary components
Multiplication process (Matrix –Vector/ Matrix- Matrix)
Real and imaginary component of output sequence of matrix products
Take IFFT of the Matrices
Multiplied polynomial is retrieved

The Multiplication of Polynomials

Let $\alpha(z) = \alpha0+\alpha1z+\alpha2z2+\cdot \quad \cdot \quad \cdot$

αpzp and $y(z) = y0+y1z+y2z2+\cdot \quad \cdot \quad \cdot$

$ynzn$ betwo polynomials of degrees p and n respectively. Then, their product $\gamma(z) =\alpha(z)y(z)$ is a polynomial of degree $p + n$ of which the coefficients comprise combinations of the coefficient of $\alpha(z)$ and $y(z)$.

A simple way of performing the multiplication is via a table of which the margins contain the elements of the two polynomials and in which the cells contain their products.

Polynomial decomposition:

The decompositions generated by the algorithms will be approximations, because as shown by an exact FIR decomposition of a FIR matrix is impossible to achieve. In this thesis, approximate polynomial decomposition algorithms will be used for the channel diagonalization problem of spatial multiplexing in wireless communications. There, a sequential best rotation algorithm is introduced using generalized Kogbetliantz transformations. The algorithm, which is not studied in this thesis, is shown to perform better than previous sequential best rotation procedures. The _rst section of this chapter will describe the performance measures employed in the study of the algorithms. As these are defined, the following sections will investigate one algorithm at a time, with respect to function, convergence and complex it.

The PSVD of a matrix has an interesting application in spatial multiplexing for wideband wireless channels. By pre coding and receive filtering by the obtained para unitary matrices, a channel matrix can be diagonalized over all frequencies, so that signaling can be performed
over a set of frequency-selective spatial modes.

Types of Convolution

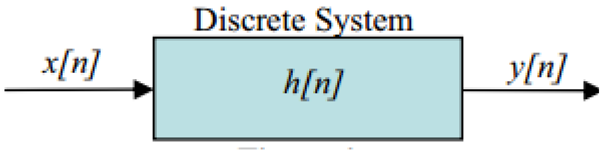
There are two types of convolution. They are

1. Linear convolution
2. Circular convolution

Discrete Linear Convolution A block diagram representing a basic discrete system is depicted in Figure. A discrete system is defined here as an entity which acts, transforms, or operates on a input signal, termed the input signal in order to produce another signal, termed the output signal. An important class of discrete systems is linear and shift-invariant systems known as discrete filters. A discrete filter is uniquely described by its impulse response signal, denoted by h[n], n∈ Z, where Z is the set of integers. A discrete filter's impulse response is obtained as a resulting output signal when the input to the filter is a delta function $\delta(n)$, n∈Z ⍰ , with $\delta[n] = 1$ for n= 0 and $\delta[n]$ =0 n for n ≠ 0 . Consider an arbitrary input signal x[n], to a discrete filter with an associated impulse response signal equal to h[n]. The output signal, say y[n] of the discrete filter is given by the following general expression

$$y[n] = \sum_{m=-\infty}^{m=+\infty} x[m]h[n-m] = \sum_{m=-\infty}^{m=+\infty} h[m]x[n-m], \quad n \in Z$$

This operation is commonly known as the linear convolution sum operation of the input signal x[n], with the impulse response signal h[n]and it is commutative operation.



The set of all discrete complex signals of the type x[n] becomes a linear space denoted by l [Z]. A subspace of this linear space, denoted by 2 l [Z] , is the set of all discrete signals with finite energy. discrete signal, say x[n] , is said to have finite energy if the condition is satisfied.

$$\sum_{n=-\infty}^{n=+\infty} x[n]x^*[n] = \langle x, x \rangle < \infty$$

The symbol "*" in the expression above denotes complex conjugation. The expression x, y is termed an inner product of x and y , with, both, x,y∈ l^2. The norm or length of a finite energy discrete signal 2 x l Z( ) is denoted by

$$\|x\| = \langle x, x \rangle^{\frac{1}{2}}$$

PERIODIC OR CYCLIC CONVOLUTION

Let x,h∈ l^2be two arbitrary sequences, each of length N . The periodic or cyclic convolution modulo N of these two signals is denoted by the expression $x \otimes_N h$ h and it is a new signal, say y , also of length N , defined by the following expression for any N ∈ z_n

$$y[n] = \left(x \otimes_N h\right)[n] = \sum_{m=0}^{N-1} x[m]h[\langle n-m \rangle_N$$

or

$$y[n] = \left(h \otimes_N x\right)[n] = \sum_{m=0}^{N-1} h[m]x[\langle n-m \rangle_N$$

The focus of this work is to develop fast and efficient algorithms for the computation of the circular or cyclic convolution operation, reaching the minimal number of multiplications according to the Winograd's theorem. Normally, two approaches are utilized to compute the cyclic convolution operation, namely, the direct approach and the transform approach. The direct approach evaluates the equation for the cyclic convolution of two N - point signals for each value N ∈ z_n resulting in a system of equations. The transform approach establishes a discrete Fourier transform (DFT) isomorphism between the cyclic convolution operation two signals in the object domain and the point-by-point multiplication operation or Hadamard product of each of the transformed signals.

This section describes a cyclic convolution operator as a linear shift invariant (LSI) operator acting on the finite dimensional linear space $l^2 \in z_n$. In addition, the cyclic convolution operator is also described as a cyclic finite impulse response (FIR) system. Combining these two attributes allows for a deeper study of the properties of the cyclic convolution operator. A formal discussion follows, arriving at a matrix representation of a cyclic convolution operation.

FPGA FLOW

The basic implementation of design on FPGA has the following steps.

- Design Entry
- Logic Optimization
- Technology Mapping
- Placement
- Routing
- Programming Unit
- Configured FPGA

Above shows the basic steps involved in implementation. The initial design entry of may be VHDL, schematic or Boolean expression. The optimization of the Boolean expression will be carried out by considering area or speed.

In technology mapping, the transformation of optimized Boolean expression to FPGA logic blocks, that is said to be as Slices. Here area and delay optimization will be taken place. During placement the algorithms are used to place each block in FPGA array. Assigning the FPGA wire segments, which are programmable, to establish connections among FPGA blocks through routing. The configuration of final chip is made in programming unit.

Advantages

The following are the advantages of the FPGA technology.

- ➢ Reduced time to market.
- ➢ Lower non-recurring engineering costs.
- ➢ Reprogrammable.

APPLICATIONS

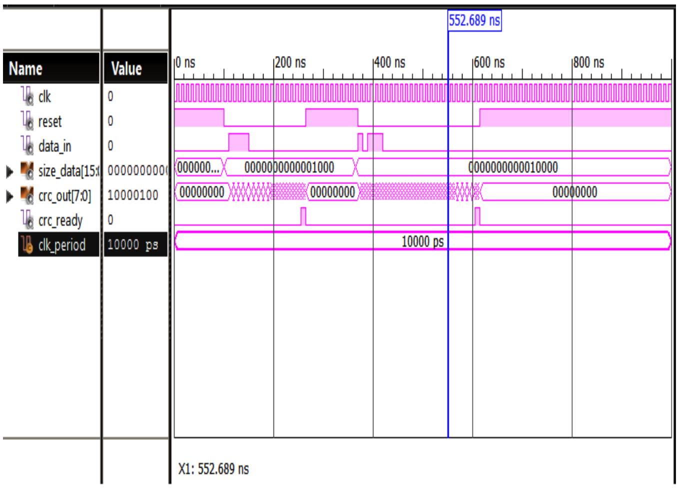
The following are the applications of the FPGA technology.

- ➢ FPGA can be applied to a very wide range of applications including: random logic, integrating multiple SPLDs, device controllers, communication encoding and filtering, small to medium sized with SRAM blocks.
- ➢ Prototyping of designs later to be implemented in gate arrays. Prototyping might be possible using only a single large FPGA (which corresponds to a small gate array in terms of capacity).
- ➢ Emulation of entire hardware systems.

Future Scope

- ➢ In this the 4x4 convolution and the data width is 4 bits shown. The convolution can be done for NXN and the width can be extended to N bits.

Simulation Results:



CONCLUSION

Here with polynomial matrix multiplication will be performed by using cyclic convolution with reduced power and efficiency than conventional ones. Cyclic convolution is most efficient technique for reducing the power and number of iterations.

REFERENCES

[1] G. H. Golub and C. F. Van Loan, Matrix Computations, John Hopkins University Press, Baltimore, MD, USA, 1996.

[2] R. H. Lambert, M. Joho, and H. Mathis, "Polynomial Singular Values for Number of Wideband Source Estimation and Principal Components Analysis," in International Conference on Independent Component Analysis, 2001, pp. 379–383.

[3] S. Redif, J. G. McWhirter, P. Baxter, and T. Cooper, "Robust Broadband Adaptive Beamforming via Polynomial Eigenvalues," in IEEE OCEANS Conference, 2006, pp. 1–6.

[4] P. P. Vaidyanathan, "Theory of Optimal Orthonormal Subband Coders," IEEE Transactions on Signal Processing, vol. 46, no. 6, pp. 1528–1543, June 1998.

[5] S. Redif, S. Weiss, and J. G. McWhirter, "An Approximate Polynomial Matrix Eigenvalue Decomposition Algorithm for Para-Hermitian Matrices," in IEEE International Symposium on Signal Processing and Information Technologies, 2011, pp. 421–425.

[6] S. Y. Kung, Y. Wu, and X. Zhang, "Bezout Space-Time Precoders and Equalizers for MIMO Channels," IEEE Transactions on Signal Processing, vol. 50, no. 10, pp. 2499–2541, October 2002.

[7] J. Foster, J. G. McWhirter, S. Lambotharan, I. Proudler, M. Davies, and J. Chambers, "Polynomial Matrix QR Decomposition and Iterative Decoding of Frequency Selective MIMO Channels," IET Signal Processing, vol. 6, no. 7, pp. 704–712, September 2012.

[8] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD Algorithm for Para-Hermitian Polynomial Matrices," IEEE Transactions on Signal Processing, vol. 55, no. 5, pp. 2158–2169, May 2007.

[9] S. Redif, S. Weiss, and J. G. McWhirter, "Design of FIR Paraunitary Filter Banks for Subband Coding using a Polynomial Eigenvalue Decomposition," IEEE Transactions on Signal Processing, vol. 59, no. 11, pp. 5253–5264, December 2011.

[10] S. Kasap and S. Redif, "Novel Field-Programmable Gate Array Architecture for Computing the Eigenvalue Decomposition of Para-Hermitian Polynomial Matrices," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2013, doi:10.1109/TVLSI.2013.2248069.

[11] R. Bracewell, The Fourier Transform and Its Applications, McGraw– Hill Higher Education, New York, USA, 1999.

[12] P. P. Vaidyanathan, Multirate Systems and Filter Banks, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[13] T. Kailath, Linear Systems, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1980.

[14] P. Duhamel and M. Vetterli, "Fast Fourier transforms: a Tutorial Review and a State of the Art," IEEE Signal Processing Society, vol. 19, no. 4, pp. 259–299, April 1990.

[15] V. K. Prasanna Kumar and Y. C. Tsai, "On Synthesizing Optimal Family of Linear Systolic Arrays for Matrix Multiplication," IEEE Transactions on Computers, vol. 40, no. 6, pp. 770–774, June 1991.

[16] Digilent inc., "Virtex-5 OpenSPARC Evaluation Platform," http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400, 795&Prod=XUPV5, Jan. 2012, Accessed 12-July-2013.