Order Number 9207129

New multiplicative 2-dimensional FFT algorithms

Tian, Hongjiang, Ph.D.

City University of New York, 1991

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

NEW MULTIPLICATIVE 2-DIMENSIONAL FFT ALGORITHMS

by

HONGJIANG TIAN

A dissertation submitted to the Graduate Faculty in Engineering in partial fulfillment of the requirements of the degree of Doctor of Philosophy, The City University of New York.

1991

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

5-31-91
Date

Richard Tolmach
Chair of Examining Committee

May 31, 1991
Date

Gerard Y. Power
Executive Officer

Michael Anshel

Joseph Barba

Michael Conner

James Cooley

Tarek Saadawi

Supervisory Committee

The City University of New York

Abstract

NEW MULTIPLICATIVE 2-DIMENTIONAL FFT ALGORITHMS

Hongjiang Tian

Advisor: Professor Richard Tolimieri

The purpose of this dissertation is to present the modern technique on the design, analysis and implementation of the discrete Fourier transform and convolution algorithms which is of central importance to the field of digital signal processing and many application fields. This dissertation develops a new family of 2-dimentional Fourier transform algorithms based on the multiplicative property of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$. The field structure is determined by the irreducible polynomials over $\mathcal{Z}/P$. While the degree of the polynomial determines the dimensionality of the Fourier transform, irreducibility determines the prime numbers P. Thus the theory presented here can be used to generate a set of multiplicative multidimensional Fourier transform algorithms which can be implemented using tensor product technique.

The difference in data flow of our algorithm comes from treating $\mathcal{Z}/P \times \mathcal{Z}/P$ as a simple object. As a result, there is no intermediate data transfer stage. By using the block-diagonalization method, the calculation of the Fourier transform is expressed as a set of small convolutions located diagonally. All blocks are skew-circulant matrices or circulant matrices which are independent of each other. The number and size of blocks are controlled again

by the multiplicative group structure. These flexibility and independency are more valuable in the vectorization and parallelization with different machine architectures.

In addition to finding its advantage in the supercomputer architectures, this new multiplicative algorithm with field and ring structures has naturally defined an algebraic structure for both prime and composite numbers on X-ray data of a crystal with crystallographic symmetry groups. By incorporating the crystallographic symmetries to our algorithm, we gain computational advantage by removing all redundant data and arithmetic. The efficiency of our algorithm is apparent since the data is compressed and the computation operation is divided by the degree of symmetry against the original non-symmetry case.

CONTENTS

$N = P_1P_2$ FFT Algorithm

Chapter 6
Multiplicative 2-Dimensional
Prime Point FFT Algorithm
for $p3$ Symmetry

Chapter 7
Multiplicative 2-Dimensional
Prime Point FFT Algorithm
for $p4$ Symmetry

Chapter 1

Introduction

1.1 Introduction

The purpose of this dissertation is to present the modern technique of the discrete Fourier transform(DFT) and convolution which is of central importance to the field of digital signal processing and many other application fields. We are concerned about the efficient fast Fourier transform(FFT) algorithms of multidimension, and the applications of these new FFT algorithms to the data with crystallographic symmetry group. Advanced topics in computer architecture such as pipelining and parallelism must also be studied before one can determine all aspects of complexity.

The multidimension FFT algorithms which we will develop are concerned with digital signal processing, and the applications of the FFT algorithms are as broad as the applications of digital signal processing itself, such as X-ray crystallography, X-ray medical imaging, radar systems, sonar systems, seismic processing, ultrasonic systems, nuclear magnetic resonance, satellite photographs and so on.

1.2 Multidimensional FFT

The discrete Fourier transform(DFT) [1,2] of a N-point sequence X is defined as

$$FX(k) = \sum_{j \in \mathcal{Z}/N} X(j)\omega^{jk}, \quad k \in \mathcal{Z}/N, \tag{1}$$

where $FX(k)$ represents the kth-term of the DFT sequence FX and $\omega = e^{-2\pi i/N}$.

We also can write (1) in the matrix form

$$FX(k) = \left[\omega^{jk} \right] X(j) \quad j \in \mathcal{Z}/N, \quad k \in \mathcal{Z}/N\,. \tag{2}$$

and we denote the above matrix $[\omega^{jk}]$ by $F(N)$ which will be called the DFT matrix of order N

$$FX = F(N)X\,. \tag{3}$$

Similarly we can define the two-dimensional Fourier transform as follows:

$$FX(b_1, b_2) = \sum_{(a_1,a_2) \in \mathcal{Z}/N \times \mathcal{Z}/N} X(a_1, a_2) e^{\frac{-2\pi i}{N}(a_1 b_1 + a_2 b_2)} \tag{4}$$

where $FX(b_1, b_2)$ is the $N \times N$ array which is Fourier Transform of function $X(a_1, a_2)$.

Now we may view the function X and its Fourier transform FX as column vectors, and represent (4) in terms of tensor product

$$FX = (F(N) \otimes F(N))X\,. \tag{5}$$

Data flow and computational complexity in (1) and (4) are controlled by algebraic nature of the indexing set. Indeed, many algorithms have been developed exploiting the nature of $\mathcal{Z}/N$ to arrive at efficient ways of computing (1) and (4).

The history of the fast Fourier transform algorithms begins with the publication in 1965 of the fast Fourier transform(FFT) algorithm of J. W. Cooley and J. W. Tukey [3] in 1965, although the history itself starts much earlier. The additive Cooley-Tukey fast Fourier transform algorithm significantly reduced a number of arithmetic operations from the order of N^2 to the order of $NlogN$, but the blocklength N should be a power of two. In fact there was an earlier FFT algorithm, due to Good(1960) [4] and Thomas(1963) [5]. But this Prime Factor Good-Thomas algorithm had failed to attract until the later 1970s [6,7].

Another kind of multiplicative algorithmic development starts with the paper of Rader [8] in 1968 and was independently discovered and extensively generalized by Winograd(1976,1978) [9,10]. In these papers the multiplicative structure of the indexing set is used to compute finite Fourier transforms as convolutions. The Winograd multiplicative algorithm minimize multiplicative complexity to decrease the cost of multiplications [11].

After this, several methods have been developed to build large or medium size FFT algorithms from the small size FFT algorithms of Rader and Winograd. The Good-Thomas Prime Factor algorithm decomposes the computation of the FFT into small size FFT's. The Winograd large-size FFT algorithm uses tensor product rules to incorporate these small FFT algorithms so as to minimize multiplications. Each of these methods has advantages and disadvantages for specific implementation depending on the relative cost of additions and multiplications on a specific computer architecture. Winograd provided efficient multiplicative algorithms for prime transform size, but the minimum multiplication feature does not fit the modern computer architecture. Auslander, Feig and Winograd(1983) [12] exhibited new algorithms for

DFT on a multidimensional data set with P points along each array, where P is a prime.

Fast convolution algorithms of small blocklength were first constructed by Agarwal and Cooley(1977) [13] using clever insights but without a general technique. Winograd(1978) [10] gave a general method of construction of fast convolution algorithms. Agarwal and Cooley(1977) [13] also found a method to break long convolutions into short convolutions using the Chinese remainder theorem. Their method works well when combined with the Winograd algorithm for short convolutions.

1.3 Crystallography

The determination of the internal microscopic structure of a crystal by X-ray crystallography requires massive repetition of Fourier transform computations. X-ray data of a crystal respects the crystallographic symmetry giving rise to data redundancy. This redundancy is controlled by the crystallographic symmetry groups [14,15]. By incorporating the crystallographic symmetries to efficient Fourier transforms, one can gain computational advantage. The use of space group symmetry to reduce the cost of DFT calculation was formally introduced by Lyn Ten Eyck(1973) [16], but only special symmetries were included in the programming package. Bricogne and Tolimieri [17] also presented an algorithm which compute the two-dimensional FFT on the data admitting 90° rotational symmetry. An, Cooley and Tolimieri(1990) [18] developed the orbit exchange method which is based on the group theoretic properties of the crystallographic groups and the ring structure of the sampling lattices– is a procedure for designing symmetrized FFT algorithms for general N which reduce to symmetrized FFT algorithms on the relatively

prime factors of N.

1.4 Description of Dissertation

In this dissertation we first develop the two-dimensional multiplicative prime point FFT algorithm for $P \equiv 2(3)$ or $P \equiv 3(4)$ in chapter 3 and chapter 4. This new algorithm is based on the field structure determined by the irreducible polynomial $x^2 + x + 1$ or $x^2 + 1$ over $\mathcal{Z}/P$ separately. While the degree of the polynomial determines the dimensionality of the Fourier transform, irreducibility determines the prime numbers P. Thus the theory presented here can be used to generate a family of multiplicative multidimensional Fourier transform algorithms which can be implemented using tensor product techniques. Our choice of the case has been primarily motivated by its application to X-ray crystallography.

The algorithm is based on the multiplicative property of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$. From the initial stage of ordering $\mathcal{Z}/P \times \mathcal{Z}/P$, the multiplicative structure is used to control the data flow in the Fourier transform computation. Then the calculation of the DFT is expressed as a set of convolutions which are more efficient for vectorization and parallel processing. This is an alternative to the more usual way of using the additive property. In addition to finding its advantage in certain architectures, the algorithm adapts naturally to processing a data with plane geometric conditions: most notable of such data is the X-ray diffraction map of a crystal.

The difference in data flow in our algorithm comes from treating $\mathcal{Z}/P \times \mathcal{Z}/P$ for a prime number P as a simple object(as opposed to viewing it as the cartesian product of two copies of $\mathcal{Z}/P$.) As a result, there is no intermediate data transfer stage. There are two variants of this multiplicative algorithm.

One of these is skew-circulant structure and another is the circulant structure. By using the block-diagonalization method with these structures, the computation of the Fourier transform is expressed as a set of small convolutions located diagonally. All blocks are skew-circulant matrices or circulant matrices of varing sizes. These variants offer options as to match different computer architectures.

The block-diagonalization method derived from the multiplicative structure reduces the computational complexity and has more flexibility to implement on varying machine architectures. The advantages are as follows:

1. Using the group theoretic structures, one arrives at skew-circulant blocks or circulant blocks of varing sizes. This flexibility is valuable in matching the vector register sizes in vector facility.

2. Since the skew-circulant or circulant blocks are located diagonally, they are independent of each other. This independency lends easily to parallel processing with multi-processors.(The number and size of skew-circulant or circulant blocks are controlled again by the multiplicative group structure.)

In chapter 5 we will extend this algorithm to the $N = PQ$ case, where P and Q are relative prime.

From chapter 6 to chapter 8 we will apply this new multiplicative FFT algorithm with field and ring structures to crystallography. As applications we will develop new FFT algorithms suited for $p3$ and $p4$ symmetry cases of crystallography. X-ray data of a crystal respects crystallographic symmetry giving rise to data redundancy is controlled by the crystallographic symmetry groups. Our method uses the crystal group to define an algebraic structure on the data set which can then be used to order data producing a highly structured algorithm. By incorporating the crystallographic symmetries to

efficient Fourier transform algorithms, one can gain computational advantage by removing all redundant data and arithmetic. In section 6.5 we will derive the new efficient multiplicative FFT algorithm for $p3$ symmetry from prime number to composite number $N = 3L$ where 3 and L are relative prime by using the CRT and tensor product technique. The efficiency of our algorithm is apparent since the data is compressed and the computation operation is divided by the degree of symmetry against the original non-symmetry case.

The last chapter 9 we will describe the performance of implementation in typical computer architectures depending on different variants of the algorithms in this thesis. Beside implementation on the sequential machine, we are interesting in implementation of our algorithms on supercomputer architectures to get efficient performance.

Chapter 2

Preliminary Mathematics

2.1 Introduction

In this chapter we will focus on some mathematical objects [19,20] which will be used repeatedly in this thesis.

- The ring $\mathcal{Z}/N$ of integers mod N
- Unit Groups
- Quotient Polynomial Rings $F[x]/f(x)$
- Tensor Products and Permutations

2.2 The Ring $\mathcal{Z}/N$

Take an integer $N > 1$. For any integer $x \in N$, let $x \; mod \; N$, then the set $\mathcal{Z}/N$ will be the remainder of the division of x by N,

$$\mathcal{Z}/N = \{0, 1, 2, \cdots, N-1\} \tag{1}$$

We define addition in $\mathcal{Z}/N$ by

$$(x+y) \ mod\, N\,, \quad x,y \in \mathcal{Z}/N\,, \tag{2}$$

and multiplication in $\mathcal{Z}/N$ by

$$(x \cdot y) \ mod\, N\,, \quad x,y \in \mathcal{Z}/N\,. \tag{3}$$

Under these operations $\mathcal{Z}/N$ becomes a commutative ring with identity 1.

Consider the following mapping

$$\phi:\ Z \rightarrow \mathcal{Z}/N\,, \tag{4}$$

which is defined by

$$\phi(x) \ = \ x \ mod\, N\,. \tag{5}$$

The mapping ϕ is a ring-homomorphism in the sense that

$$\phi(x+y) \ = \ (\phi(x)+\phi(y)) \ mod\, N\,, \tag{6}$$

$$\phi(x \cdot y) \ = \ (\phi(x) \cdot \phi(y)) \ mod\, N\,. \tag{7}$$

Two integers x and y are said to be congruent $mod\ N$ if $\phi(x) = \phi(y)$. That is

$$N/(x-y)\,. \tag{8}$$

It also means that

$$x \equiv y \ mod\, N\,. \tag{9}$$

2.3 Unit Groups

The unit group of $\mathcal{Z}/N$, denoted by $U(N)$, consists of all elements $x \in \mathcal{Z}/N$ which have multiplicative inverses $y \in \mathcal{Z}/N$:

$$1 = (xy) \ mod \, N \, . \tag{1}$$

It is easy to show that $U(N)$ is a group under the ring-multiplication in $\mathcal{Z}/N$.

Theorem 1 *The set of the unit group $U(N)$ satisfies that*

$$U(N) = \{x \in \mathcal{Z}/N \mid (x, N) = 1\} \, . \tag{2}$$

Example 1

Let N=12. Then we can find unit group

$$U(12) = \{1, 5, 7, 11\} \, .$$

From theorem 1, if P is a prime, since every nonzero element in $\mathcal{Z}/P$ has a multiplicative inverse, then

$$U(P) = \{1, 2, \cdots, P-1\} \, , \tag{3}$$

Now because $\mathcal{Z}/P$ is a commutative ring with identity, it follows that $\mathcal{Z}/P$ is a finite field.

Theorem 2 *Z/P is a finite field if and only if P is a prime.*

Prove

We have shown that if P is a prime then $\mathcal{Z}/P$ is a field. Suppose P is not a prime, and write $P = N_1 N_2$ where

$$1 < N_1, N_2 < N \, .$$

By theorem 1, since $(N, N_1) = N_1 \neq 1$, N_1 does not have a multiplicative inverse in $\mathcal{Z}/N$ and $\mathcal{Z}/N$ is not a field, completing the proof of theorem 2.

Theorem 3 *For an odd prime P, and integer $x \geq 1$, the unit group*

$$U(P^x) \tag{4}$$

is a cyclic group.

This important result is proved in many number theory books, for instance [19].

Lemma 1 *An element $\gamma \in U(P^x)$, called a generator, can be found such that*

$$U(P^x) = \{\gamma^k \mid 0 \leq k < o(U(P^x))\} . \tag{5}$$

where $o(U(P^x))$ is the number of elements in the set $U(P^x)$.

2.4 Polynomial Rings

Assume F is a field. $F[x]$ is defined as the ring of polynomials in the indeterminate x which has coefficients in F. An element in $F[x]$ can be expressed as

$$f(x) = \sum_{n=0}^{k} f_n x^n , \quad f_n \in F . \tag{1}$$

If $f_k \neq 0$ in (1), the value k is called the degree of $f(x)$, i.e.

$$deg\ f(x) = k . \tag{2}$$

The zero polynomial, denoted by 0, has by convention degree $-\infty$. Then we have the important result

$$deg\ (f(x)g(x)) = deg\ f(x) + deg\ g(x) . \tag{3}$$

2.4.1 Divisibility Condition

If $f(x)$ and $g(x) \neq 0$ are polynomials in $F[x]$, then there is a unique pair of polynomials $q(x)$ and $r(x)$ in $F[x]$ satisfying

$$f(x) = q(x)g(x) + r(x) \quad deg\ r(x) < deg\ g(x)\ . \tag{4}$$

The polynomial $q(x)$ is called the *quotient* of the division of $g(x)$ into $f(x)$. The polynomial $r(x)$ is called the *remainder* of the division of $f(x)$ by $g(x)$.

Theorem 1 *If $f(x)$ and $g(x)$ are polynomials over F, there exists a unique polynomial $d(x)$ called the greatest common divisor of $f(x)$ and $g(x)$ over F*

$$d(x) = (f(x), g(x))\ . \tag{5}$$

By the divisibility condition above, if $f(x)$ and $g(x)$ are relatively prime over F, then

$$1 = c_1(x)f(x) + c_2(x)g(x)\ , \tag{6}$$

for some polynomials $c_1(x)$ and $c_2(x)$ over F.

2.4.2 The Ring F[x]/f(x)

Take a polynomial $f(x)$ of degree N over $F[x]$. We define

$$F[x]/f(x)\ , \tag{7}$$

equal to the set of all polynomials $g(x)$ over F satisfying

$$deg\ g(x) < N\ . \tag{8}$$

the polynomial $g(x)$ in $F[x]/f(x)$ can be written as

$$g(x) = \sum_{n=0}^{k-1} g_n a^n , \quad g_n \in F , \tag{9}$$

and we can regard $F[x]/f(x)$ as an N-dimensional vector space over F having basis

$$1, x, \cdots, x^{N-1} . \tag{10}$$

For any $g(x) \in F[x]$, denote the remainder of the division of $g(x)$ by $f(x)$ as follows,

$$g(x) \ mod \ f(x) \in F[x]/f(x) . \tag{11}$$

We define a ring multiplication on $F[x]/f(x)$ by

$$(g(x)h(x)) \ mod \ f(x) , \quad g(x), h(x) \in F[x]/f(x) . \tag{12}$$

By direct computation, it shows that the vector space $F[x]/f(x)$ becomes an algebra over F with the multiplication (12).

Two polynomials $g(x)$ and $h(x)$ over F are to be congruent $mod \ f(x)$ if $g(x) \ mod \ f(x) = h(x) \ mod \ f(x)$. It means

$$f(x)/(g(x) - h(x)) . \tag{13}$$

We also can write

$$g(x) \equiv h(x) \ mod \ f(x) , \tag{14}$$

Define the mapping

$$\phi : F[x] \rightarrow F[x]/f(x) , \tag{15}$$

by the formula

$$\phi(g(x)) = g(x) \ mod \ f(x) . \tag{16}$$

It is easy to show that ϕ is a ring-homomorphism of $F[x]$ onto $F[x]/f(x)$ whose kernel

$$\{g(x) \in F[x] \mid \phi(g(x)) = 0\} \tag{17}$$

is the ideal $(f(x))$.

We have known by theorem 2 of section 2.3 that $\mathcal{Z}/P$ is a field if and only if P is a prime. From now we will construct fields using the rings $F[x]/f(x)$.

Theorem 2 *The ring $F[x]/f(x)$ is a field if and only if $f(x)$ is irreducible over F.*

Prove

Suppose $f(x)$ is irreducible. Take any nonzero polynomial $g(x)$ in $F[x]/f(x)$. By (6) we have

$$1 = c_1(x)g(x) + c_2(x)f(x) ,$$

where $c_1(x)$ and $c_2(x)$ are polynomials over F. Then

$$1 \equiv c_1(x)g(x) \ \ mod\ f(x) ,$$

so $c_1(x)$ mod $f(x)$ is the multiplicative inverse of $g(x)$ in $F[x]/f(x)$. Since $g(x)$ is an arbitrary nonzero polynomial in $F[x]/f(x)$, the commutative ring $F[x]/f(x)$ is a field.

Conversely, suppose $f(x)$ is not irreducible. Then

$$f(x) = f_1(x)f_2(x) ,$$

where

$$0 < deg\ f_n(x) < deg\ f(x) , \quad n = 1, 2 .$$

It means that $f_1(x)$ and $f_2(x)$ are in $F[x]/f(x)$ and

$$0 = (f_1(x)f_2(x)) \ mod\ f(x)\ .$$

If $f_1(x)$ has a multiplicative inverse, then

$$0 \equiv f_2(x) \ mod\ f(x)\ .$$

It is contradictory, completing the proof of the converse of the theorem.

2.5 Tensor Product and Stride Permutation

2.5.1 Introduction

The tensor or Kronecker product offers a natural language for modeling and designing digital signal processing(DSP) algorithms. Closely associated with tensor products are a class of permutations, the stride permutation. These permutations govern the addressing between the stages of the tensor product decompositions of DSP algorithms, shch as digital Fourier transform(DFT) algorithm.

The linguistic power of the tensor product is shown on the papers of R. Tolimieri [20] and J. Johnson [21] where it is used to model fast Fourier transform algorithms. It show that tensor product formulation of DSP algorithms also offers the convenience of modifying the algorithms to adapt to specific computer architectures. The formalism of tensor product notation can be used to keep track of the complicated index calculation needed in implementing Fourier transform algorithms. By establishing relationships between certain tensor product constructs and computer architectures, FFT algorihtms can be easily optimized for either hardware or software implementation.

An algorithm can be described in many cases a matrix factorization

$$Y = \prod_{n=0}^{k-1} Y_k \,, \tag{1}$$

where the action of the matrix Y is to be computed. The tensor product also makes it very easy to modify an algorithm by exploiting the underlying algebraic structure of its matrix representation.

2.5.2 Tensor Product

Tensor product algebra is an important tool for presenting mathematical formulations of DSP algorithms so that these algorithms may be studied and analyzed in a unified format. In this section, we will present some of the basic properties of tensor products which are encountered in the algorithms that we will develop in future sections of this thesis. These properties will be very useful in manipulating the factorizations of discrete DFT matrices.

The tensor product of an $K \times L$ matrix A with an $M \times N$ matrix B is defined by the $KM \times LN$ matrix, $X \otimes Y$,

$$X \otimes Y = \begin{bmatrix} x_{00}Y & x_{01}Y & \cdots & x_{0,L-1}Y \\ \vdots & & & \\ x_{K-1,0}Y & x_{K-1,1}Y & \cdots & x_{K-1,L-1}Y \end{bmatrix} . \tag{2}$$

It is natural to view the tensor product $X \otimes Y$ as being formed from blocks of scalar multiples of Y.

By straightforward computation we have the associative property

Theorem 1

$$A \otimes (B \otimes C) = (A \otimes B) \otimes C \tag{3}$$

Theorem 2 *If X is an $K \times L$ matrix and Y is an $M \times N$ matrix, then*

$$(X \otimes Y)(\underline{x} \otimes \underline{y}) = X\underline{x} \otimes Y\underline{y}\,, \tag{4}$$

for any vectors $\underline{x}$ and $\underline{y}$ of sizes L and N, respectively.

Theorem 3 *If A and C are $K \times K$ matrices and B and D are $N \times N$ matrices, then*

$$(A \otimes B)(C \otimes D) = (AC \otimes BD)\,. \tag{5}$$

Prove

Take vector $\underline{x}$ and $\underline{y}$ of sizes K and N respectively. By (4)

$$(A \otimes B)(C \otimes D)(\underline{x} \otimes \underline{y}) = (A \otimes B)(C\underline{x} \otimes D\underline{y})$$

$$= AC\underline{x} \otimes BD\underline{y}\,,$$

proving (5), in light of the proceeding discussion.

An important special case of formula (5) is the following decomposition.

Lemma 1 *Denote by I_M the $M \times M$ identity matrix. Then*

$$A \otimes B = (I_M \otimes B)(A \otimes I_L) = (A \otimes I_L)(I_M \otimes B)\,, \tag{6}$$

where A is an $M \times M$ matrix and B is an $L \times L$ matrix, and $I_M \otimes B$ is the direct sum of M copies of B,

$$I_M \otimes B = \bigoplus_{i=1}^{M} B\,. \tag{7}$$

Factorization (6) decomposes the computation of $(A \otimes B)\underline{x}$ into the parallel operation $(I_M \otimes B)$ followed by the vector operation $(A \otimes I_L)$.

We will also have the inverse and transpose properties as follows:

Theorem 4

$$(A \otimes B)^{-1} = (A^{-1} \otimes B^{-1}) . \tag{8}$$

Theorem 5

$$(A \otimes B)^{t} = (A^{t} \otimes B^{t}) . \tag{9}$$

2.5.3 Stride Permutations

In this section, we will describe the stride permutations that gives the data flow required for vectorization or parallelization of a tensor product computation which plays a crucial role in the implementation of DFT computations.

The M-point stride N permutation matrix $P(M, N)$ is defined by

$$P(M, N)(\underline{x} \otimes \underline{y}) = \underline{y} \otimes \underline{x} , \tag{10}$$

where $M = LN$ and $\underline{x}$ and $\underline{y}$ are arbitrary vectors of orders L and N, respectively. The action of $P(M, N)$ on an arbitrary vector $\underline{z}$ of order M can be described as follows, take

$$Z = \begin{bmatrix} z_0 & z_N & \cdots & z_{(L-1)N} \\ z_1 & & & \\ \vdots & & & \\ z_{N-1} & z_{2N-1} & \cdots & z_{M-1} \end{bmatrix} . \tag{11}$$

Then $P(M,N)\underline{z}$ will corresponds to the transpose of Z. Take

$$Z^t = [W_0\, W_1 \cdots W_{N-1}]\,, \tag{12}$$

where

$$W_i = \begin{bmatrix} z_i \\ z_{i+N} \\ \vdots \\ z_{i+(L-1)N} \end{bmatrix}, \quad 0 \le i < N\,, \tag{13}$$

Then we can compute $P(M,N)\underline{z}$ by striding through $\underline{z}$ with stride N. The result is obtained by

$$P(M,N)\underline{z} = \begin{bmatrix} W_0 \\ W_1 \\ \cdot \\ W_{N-1} \end{bmatrix}. \tag{14}$$

By using the formula (10) frequently, the algebra of stride permutation will play an important role on the design of tensor product algorithms.

Example 1

Let $\underline{y}$ is a 6-dimensional vector of stride 1. Take M=6 and N=2, the permutation matrix is

$$P(6,2) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

we can find $P(6,2)^3 = I_6$, and

$$P(6,2)\underline{y} = \begin{bmatrix} y_0 \\ y_2 \\ y_4 \\ y_1 \\ y_3 \\ y_5 \end{bmatrix} .$$

Theorem 6 *If* $M = KLN$ *then*

$$P(M,LN) = P(M,L)P(M,N) . \tag{15}$$

Prove

Take vectors $\underline{x}$, $\underline{y}$ and $\underline{z}$ of sizes L, N and K respectively. Then

$$P(M,LN)(\underline{x} \otimes \underline{y} \otimes \underline{z}) = \underline{y} \otimes \underline{z} \otimes \underline{x} ,$$

and

$$P(M,L)P(M,N)(\underline{x} \otimes \underline{y} \otimes \underline{z}) = P(M,L)(\underline{z} \otimes \underline{x} \otimes \underline{y}) = \underline{y} \otimes \underline{z} \otimes \underline{x} ,$$

completing the proof of the theorem.

In particular, from theorem 6 we have

Lemma 2

$$P(MN,N)^{-1} = P(MN,M) . \tag{16}$$

In (6) of section 2.5.2, we have described that the tensor product $(A \otimes I_L)$ can be vectorized and $(I_M \otimes B)$ can be parallelized, or both can be combined

together. If we want to interchange the operations above, an important tool is the commutation theorem. For a given factorization the commutation theorem gives a procedure to construct different variants which can match special computer architecture.

Theorem 7 *If A is an $L \times L$ matrix and B is an $N \times N$ matrix, then*

$$P(A \otimes B)P^{-1} = B \otimes A\,, \tag{17}$$

where $P = P(M,N)$ and $M = LN$.

Prove

Set $\underline{z} = \underline{x} \otimes \underline{y}$ where $\underline{x}$ and $\underline{y}$ are vectors of orders L and N separately. Then we have

$$(A \otimes B)(\underline{x} \otimes \underline{y}) = A\underline{x} \otimes B\underline{y}\,,$$

$$P(A \otimes B)\underline{z} = P(A\underline{x} \otimes B\underline{y}) = B\underline{y} \otimes A\underline{x}\,.$$

and

$$(B \otimes A)P\underline{z} = (B \otimes A)(\underline{y} \otimes \underline{x}) = B\underline{y} \otimes A\underline{x}\,,$$

completing the proof of the theorem.

Corollary 1

$$P(I_L \otimes B)P^{-1} = B \otimes I_L\,, \tag{18}$$

where $P = P(M,N)$ and $M = LN$.

as an important application of the commutation theorem, we observe that

$$A \otimes B = (A \otimes I_N)P(M,L)(B \otimes I_L)P(M,L)^{-1} , \tag{19}$$

$$A \otimes B = P(M,L)(I_N \otimes A)P(N,L)^{-1}(I_L \otimes B) . \tag{20}$$

The formula of factorization (19) decomposes $(A \otimes B)$ into a sequence of vector operations. The first operates on vectors of size L while the second operates on vectors of size N. The intervening stride permutations provide a mathematical language for describing the readdressing between stages of the computation. In the same way, we interpret (20) as a sequence of parallel operations.

2.6 Summary

In recent years, the modern number theory plays an major role in designing the digital signal processing(DSP) algorithms, e.g. the digital Fourier transform algorithm. The topics mentioned above are only the basic knowledge which will be used frequently in the thesis.

The Chinese Remainder Theorem(CRT) is also an significant topic in modern number theory. The construction of the Chinese Remainder ring-isomorphism using idempotents is important to the rest of this thesis. This construction results in our algorithms having the nice structure and simple data flow especially for vectorization and parallelization. We will begin to discuss the details of CRT at Chapter 5.

Chapter 3

Multiplicative 2-Dimensional Prime Point ($P \equiv 2(3)$) FFT Algorithm

3.1 Introduction

In this chapter, we develop a multiplicative two-dimensional $P \times P$ Fourier transform algorithm for a prime $P \equiv 2 \ mod \ 3$. The algorithm is based on the field structure determined by the irreducible polynomial $x^2 + x + 1$ over $\mathcal{Z}/P$. While the degree of the polynomial determines the dimensionality of the Fourier transform, irreducibility determines the prime numbers P. Thus the theory presented here can be used to generate a family of multiplicative multidimensional Fourier transform algorithms. Our choice of the case has been motivated by its application to X-ray crystallography.

The algorithm uses the multiplicative property of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$. From the initial stage of ordering $\mathcal{Z}/P \times \mathcal{Z}/P$, the multiplicative structure is used to control the data flow in the Fourier transform computation.

Then the calculation of the DFT is expressed as a set of convolutions which are more efficient for vectorization and parallel processing. This is an alternative to the more usual way of using the additive property. In addition to finding its advantage in certain architectures, the algorithm adapts naturally to processing a data with plane geometric conditions: most notable of such data is the X-ray diffraction map of a crystal.

For a natural number N, denote by $\mathcal{Z}/N \times \mathcal{Z}/N$, the cartesian product of two copies of the ring $\mathcal{Z}/N$. An element of $\mathcal{Z}/N \times \mathcal{Z}/N$ is denoted by (a_1, a_2), $a_1, a_2 \in \mathcal{Z}/N$. For a function X defined on $\mathcal{Z}/N \times \mathcal{Z}/N$, the Fourier transform of X is defined by

(1)
$$FX(b_1, b_2) = \sum_{(a_1,a_2)\in \mathcal{Z}/N\times\mathcal{Z}/N} X(a_1, a_2) e^{\frac{-2\pi i}{N}(a_1 b_1 + a_2 b_2)}$$

Data flow and computational complexity in (1) are controlled by the algebraic nature of the indexing set. Indeed, many algorithms have been developed exploiting the nature of $\mathcal{Z}/N$ to arrive at efficient ways of computing (1).

Efficiency of an algorithm should be measured upon implementation. Since data flow and computational complexity should be changed dependenting on machine architectures, it becomes important to have several ways of controlling them so as to suit them on varying architectures. The difference in data flow in our algorithm comes from treating $\mathcal{Z}/P \times \mathcal{Z}/P$ for a prime number P, as a simple object (as opposed to viewing it as the cartesian product of two copies of $\mathcal{Z}/P$.) As a result, there is no intermediate data transfer stage. There are two variants of this multiplicative algorithm, called skew-circulant and circulant structure. By using the block-diagonalization method, the calculation of the Fourier transform is expressed as a set of small convolutions located diagonally. All blocks are skew-circulant matrices or circulant ma-

trices which are independent of each other. The number and size of blocks are controlled again by the multiplicative group structure. These flexibility and independency are more valuable in the vectorization and parallelization with different machine architectures. We will start our discussions to simpler cases. However, with little more analysis on the multiplicative structure of the indexing ring, the approach can be extended to cover other cases.

3.2 Circulant and Skew-Circulant Matrix

Since the calculation of our new algorithms are made by expressing the FFT as a set of cyclic convolutions, it is necessary to introduce some concepts about circulant and skew-circulant matrix before presenting the algorithms.

Considing two vectors $\underline{X}$ and $\underline{Y}$ of order N, the cyclic convolution $C(k)$ of $\underline{X}$ and $\underline{Y}$ is the vector of order N defined by

$$(1) \qquad C(k) = \underline{X} * \underline{Y} = \sum_{n \in \mathcal{Z}/N} X(k-n)Y(n) , \quad k \in \mathcal{Z}/N .$$

Cyclic convolution also can be expressed by the form of matrix multiplication. First we define the $N \times N$ cyclic shift matrix S by

$$(2) \qquad S = \begin{bmatrix} & & & & 1 \\ 1 & & & & \\ & 1 & & & \\ & & \cdot & & \\ & & & 1 & \end{bmatrix} ,$$

that is

$$S\underline{X} = S\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} X_{N-1} \\ X_0 \\ \vdots \\ X_{N-2} \end{bmatrix}. \tag{3}$$

Based on the definition of cyclic shift matrix S, we can define another circulant matrix C by

$$C = \begin{bmatrix} X_0 & X_{N-1} & \cdots & X_1 \\ X_1 & X_0 & \cdots & X_2 \\ \vdots & & & \\ X_{N-1} & X_{N-2} & \cdots & X_0 \end{bmatrix}, \tag{4}$$

Where C is related to the vector $\underline{X}$.

The relation between the cyclic shift matrix S and the circulant matrix C is

$$C = X_0 I_N + X_1 S + \cdots + X_{N-1} S^{N-1} = \sum_{n \in Z/N} X_n S^n, \tag{5}$$

where $S^N = I_N$.

Now we consider the cyclic convolution $\underline{C} = \underline{X} * \underline{Y}$. By direct computation we can find that the $N \times N$ cyclic convolution can be expressed as a circulant matrix C multiplying the vector $\underline{Y}$, that is

$$\underline{C} = \underline{X} * \underline{Y} = C\underline{Y}. \tag{6}$$

A skew-circulant matrix $\tilde{C}$ compared with the circulant matrix C is defined as

$$\tilde{C} = \begin{bmatrix} X_0 & X_1 & \cdots & X_{N-1} \\ X_1 & X_2 & \cdots & X_0 \\ \vdots & & & \\ X_{N-1} & X_0 & \cdots & X_{N-2} \end{bmatrix} . \tag{7}$$

The skew-circulant matrix $\tilde{C}$ can be expressed by circulant matrix C by

$$\tilde{C} = C\tilde{S} , \tag{8}$$

where $\tilde{S}$ is the $N \times N$ skew-cyclic shift matrix defined by

$$\tilde{S} = \begin{bmatrix} 1 & & & \\ & & & 1 \\ & & 1 & \\ & \cdot & & \\ & 1 & & \end{bmatrix} , \tag{9}$$

that is

$$\tilde{S}\underline{X} = \tilde{S}\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} X_0 \\ X_{N-1} \\ \vdots \\ X_1 \end{bmatrix} \tag{10}$$

By (9) it means that a skew-circulant matrix $\tilde{C}$ can be changed into the multiplication of a circulant matrix C and a skew-cyclic shift matrix $\tilde{S}$.

For computing the multiplication of $\tilde{C}\underline{Y}$, we can calculate the permutation $\tilde{S}\underline{Y}$ first, then we compute the cyclic convolution $C(\tilde{S}\underline{Y})$. That is

$$\tilde{C}\underline{Y} = C(\tilde{S}\underline{Y}) . \tag{11}$$

Since the permutation $\tilde{S}\underline{Y}$ is expensive sometimes, we will develop another method to block-diagonize the skew-circulant matrix $\tilde{C}$ to get more efficient

results for different machine architectures. This will be discussed in section 3.6.

3.3 Field of P^2 Elements for $P \equiv 2(3)$ and Fourier Transform Matrix

The polynomial $x^2 + x + 1$ is irreducible over $\mathcal{Z}/P$ for $P \equiv 2(3)$ [19]. Set $\rho = exp(-2\pi i/3)$. Then $\rho^2 + \rho + 1 = 0$ and $\mathcal{Z}/P[\rho]$ is a field with P^2 elements; i.e., the nonzero elements of $\mathcal{Z}/P[\rho]$ form a cyclic group under multiplication. Let γ be a cyclic generator. Then

$$1 = \gamma^0 \, , \, \gamma \, , \, \gamma^2 \, , \, \cdots \, \gamma^{P^2-2} \tag{1}$$

are the distinct nonzero elements of $\mathcal{Z}/P[\rho]$

A generator γ of $U(P)$ can be found satisfying

$$\rho \, = \, \gamma^n \, , \quad n \, = \, (P^2 - 1)/3 \, . \tag{2}$$

where $U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P$.

An element $a \in \mathcal{Z}/P[\rho]$ can be written uniquely as $a_1 + \rho a_2$, $a_1, a_2 \in \mathcal{Z}/P$. Arithmetic in $\mathcal{Z}/P[\rho]$ is defined by

$$(a_1 + \rho a_2) + (b_1 + \rho b_2) \, = \, (a_1 + b_1) + \rho(a_2 + b_2) \, . \tag{3}$$

$$(a_1 + \rho a_2)(b_1 + \rho b_2) \, = \, a_1 b_1 - a_2 b_2 + \rho(a_2 b_1 + a_1 b_2 - a_2 b_2) \, .$$

Define a mapping $\phi : \mathcal{Z}/P[\rho] \rightarrow \mathcal{Z}/P$ by

$$\phi(a_1 + \rho a_2) \, = \, a_1 \, . \tag{4}$$

Observe that

$$\phi(a+b) = \phi(a) + \phi(b)\,, \tag{5}$$

$$\phi(ab) = a_1 b_1 - a_2 b_2\,.$$

Lemma 1

$$\phi(ab) = \phi(ba)\,. \tag{6}$$

This follows from the commutativity in $\mathcal{Z}/P[\rho]$.

Lemma 2

$$\gamma^{P^2-1} = 1 \tag{7}$$

Proof of this can be found in any text book on algebra.

For a function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$, the Fourier transform of X is

$$FX(b_1, -b_2) = \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1, a_2) e^{\frac{-2\pi i}{P}(a_1 b_1 - a_2 b_2)} \tag{8}$$

$$= \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1, a_2) e^{\frac{-2\pi i}{P}\phi(ab)}\,.$$

The function X and its Fourier Transform FX can be viewed as column vectors once $\mathcal{Z}/P \times \mathcal{Z}/P$ is ordered. To this end, note that $(a_1, a_2) \in \mathcal{Z}/P \times \mathcal{Z}/P$ corresponds uniquely to $a_1 + \rho a_2 \in \mathcal{Z}/P[\rho]$. Hence an ordering of $\mathcal{Z}/P[\rho]$ will yield an ordering for $\mathcal{Z}/P \times \mathcal{Z}/P$.

Take the ordering by $U(P)$. Now we may view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent (8) in terms of matrices as

$$(9) \qquad \begin{bmatrix} Y(0) \\ \underline{Y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix} \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix},$$

where

$$(10) \qquad W(P) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \le j,k < P^2-1},$$

and the element $0 \in \mathcal{Z}/P[\rho]$ has been placed in front of $U(P)$. For the rest of our discussion, we ignore $0 \in \mathcal{Z}/P[\rho]$. (We can place this at the last stage of computations.)

3.4 New Algorithm with Skew-Circulant Structure

We will now to derive $W(P)$ depending on different ordering on $\mathcal{Z}/P \times \mathcal{Z}/P$.

Lemma 1 *Take P is prime number. For $P \equiv 2(3)$, $P^2 - 1$ is divisible by 12.*

Prove

For $P \equiv 2(3)$, i.e. $P = 3k + 2, k \in Z$, we can find $P^2 - 1 = (P-1)(P+1)$ is divisible by 4. For $P = 3k + 2$, then $P^2 - 1 = 3(3k^2 + 4k + 1)$. It means that $P^2 - 1$ is divisible by 3. So for $P \equiv 2(3)$, $P^2 - 1$ is divisible by 12.

Set $m = \frac{P^2-1}{6}$ and $n = 2m$. Also set $\omega = exp(-\frac{2\pi i}{P})$. We will need the following properties of generator γ and the mapping function ϕ, all of which are easy to show.

(1) $$\gamma^{3m} = \gamma^{\frac{P^2-1}{2}} = -1\,,$$

(2) $$\phi(\gamma^{j+3m}) = -\phi(\gamma^j)\,,$$

(3) $$\omega^{\phi(\gamma^{j+3m})} = (\omega^{\phi(\gamma^j)})^*\,.$$

where $*$ denotes the complex conjugation.

Variant 1

By lemma 1, for $P \equiv 2(3)$, P^2-1 is divisible by 12. Take $m = \frac{P^2-1}{6}$, and define the set

(4) $$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{3m-1}\}\,.$$

Thus we have that $1, \gamma^{3m}$ is the quotient of $U(P)/S$ and

(5) $$U(P) = S \cup \gamma^{3m} S$$

View S as ordered by (4). This ordering can be used to order $\gamma^{3m}S$. Thus the ordering of $U(P)$ is obtained by putting $S; \gamma^{3m}S$. We will denote by $R(P)$ the group $U(P)$ with this ordering. Now examine the matrix $W(P)$. set $\omega = e^{\frac{-2\pi i}{P}}$.

Theorem 1 *$W(P)$ is of the form*

(6) $$\begin{bmatrix} A & B \\ B & A \end{bmatrix},$$

where A and B are $3m \times 3m$ matrices with $A = B^*$. The conjugate of B is denoted by B^*.

Proof

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 \\ W_3 & W_4 \end{bmatrix},$$

where W_i, $1 \leq i \leq 4$ is an $3m \times 3m$ matrix. For $0 \leq k, l < 3m$, denote the k-th row l-th column entry of an $3m \times 3m$ matrix W_i by $W_i(k,l)$. Then using the properties of generator γ and the mapping function ϕ shown in (1)-(3),

$$W_4(k,l) = \omega^{\phi(\gamma^{3m}\gamma^k\gamma^{3m}\gamma^l)} = \omega^{\phi(\gamma^{6m}\gamma^{k+l})} = \omega^{\phi(\gamma^{k+l})} = W_1(k,l)\,.$$

$$W_3(k,l) = \omega^{\phi(\gamma^{3m}\gamma^k\gamma^l)} = \omega^{\phi(\gamma^{3m}\gamma^{k+l})},$$

By Lemma 1 of section 3.3, $W_3(k,l)$ can be changed to

$$W_3(k,l) = \omega^{\phi(\gamma^{k+l}\gamma^{3m})} = W_2(k,l)\,.$$

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks A and B.

By the properties of γ and ϕ showed in (1) to (3), we know that $\gamma^{(3m+j)} = -\gamma^j$. So matrix A equals to the conjugate of B, that is $A = B^*$. $W(P)$ will be 2×2 block circulant matrix with $A = B^*$

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix},$$

where $*$ we mean complex conjugation.

Corollary 1 *In matrix $W(P)$, each submatrix A and A^* is conjugate skew-circulant matrix.*

Prove

Denote the k-th row l-th column entry of the $3m \times 3m$ matrix A by $A(k,l)$. Then using the properties (1)-(3),

(1) For $0 \leq k, l < 3m-1$,

$$A(k,l+1) = \omega^{\phi(\gamma^{k+l+1})} = \omega^{\phi(\gamma^{(k+1)+l})} = A(k+1,l)\,,$$

(2) For $1 \leq k < 3m$,

$$A(k,3m-1) = \omega^{\phi(\gamma^{k+3m-1})} = \omega^{\phi(\gamma^{k-1}\gamma 3m)} = \omega^{\phi(-\gamma^{k-1})} = A(k-1,0)^*\,,$$

hence A is conjugate skew-circulant matrix. Since A^* is the conjugate of A, A^* is also a conjugate skew-circulant matrix.

Corollary 2 *If matrix $W(P)$ is 2×2 conjugate block-circulancy*

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix}, \tag{7}$$

where A is a conjugate skew-circulant matrix, then $W(P)$ is a skew-circulant matrix.

Prove

We know that $W(P)$ is $6m \times 6m$ matrix,

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix}.$$

We can rewrite

$$W(P) = \left[\begin{array}{ccc|ccc} a_{0,0} & \cdots & a_{0,3m-1} & a_{0,3m} & \cdots & a_{0,6m-1} \\ a_{1,0} & \cdots & a_{1,3m-1} & a_{1,3m} & \cdots & a_{1,6m-1} \\ \vdots & & & \vdots & & \\ a_{3m-1,0} & \cdots & a_{3m-1,3m-1} & a_{3m-1,3m} & \cdots & a_{3m-1,6m-1} \\ \hline a_{3m,0} & \cdots & a_{3m,3m-1} & a_{3m,3m} & \cdots & a_{3m,6m-1} \\ a_{3m+1,0} & \cdots & a_{3m+1,3m-1} & a_{3m+1,1} & \cdots & a_{3m+1,6m-1} \\ \vdots & & & \vdots & & \\ a_{6m-1,0} & \cdots & a_{6m-1,3m-1} & a_{6m-1,3m} & \cdots & a_{6m-1,6m-1} \end{array}\right],$$

where $m = \frac{P^2-1}{6}$.

Now we look at the matrix $W(P)$. Since $a_{0,0} = a^*_{0,3m}$ and $a_{0,3m} = a^*_{1,6m-1}$, hence $a_{0,0} = a_{1,6m-1}$. In general

$$a_{i,0} = a^*_{i,3m}\,, \qquad a_{i,3m} = a^*_{i+1,6m-1}\,,$$

and

$$a_{i,0} = a_{i+1,6m-1}\,, \qquad 0 \le i < 6m-1\,,$$

and

$$a_{i,j} = a_{i-1,j+1} \qquad 1 \le i < 6m\,, 0 \le j < 6m\,.$$

Hence $W(P)$ is skew-circulant matrix.

Variant 2

For $P \equiv 2(3)$, $P^2 - 1$ is divisible by 12. Set $m = \frac{P^2-1}{6}$ and $n = 2m$. The element γ^3 is of order n and generates the subgroup

$$S = \{\gamma^0, \gamma^3, \gamma^6, \cdots, \gamma^{3(n-1)}\} \tag{8}$$

For the simplicity of presentation, we will assume that n is not divisible by 3 again.(i.e., that $P^2 - 1$ is not divisible by 9. Otherwise the method we present here can be modified to yield the desired results.) Thus we have that $\{1, \gamma^n, \gamma^{2n}\}$ is the quotient of $U(P)/S$, where $U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$, and

$$U(P) = S \cup \gamma^n S \cup \gamma^{2n} S \,. \tag{9}$$

View S as ordered by (8). This ordering can be used to order $\gamma^n S$ and $\gamma^{2n} S$. Now, the ordering of $U(P)$ is obtained by putting $S; \gamma^n S; \gamma^{2n} S$. We will also denote by $R(P)$ the group $U(P)$ with this ordering.

Theorem 2 *$W(P)$ will be of the form*

$$\begin{bmatrix} A & B & C \\ B & C & A \\ C & A & B \end{bmatrix} \tag{10}$$

where A, B and C are $n \times n$ matrices.

Prove

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 & W_3 \\ W_4 & W_5 & W_6 \\ W_7 & W_8 & W_9 \end{bmatrix},$$

where W_i, $1 \leq i \leq 9$ is an $n \times n$ matrix. We will show that $W_1 = W_6 = W_8$. The other cases are proved in exactly the same way. For $0 \leq k, l < n$, denote the k-th row l-th column entry of an $n \times n$ matrix M by $M(k, l)$,

$$W_6(k,l) = \omega^{\phi(\gamma^n \gamma^{3k} \gamma^{2n} \gamma^{3l})} = \omega^{\phi(\gamma^{3k+3l} \gamma^{3n})} = \omega^{\phi(\gamma^{3k+3l})} = W_1(k,l) \,.$$

In view of Lemma 1 of section 3.3, this is also $W_8(k,l)$, completing the theorem 2.

This structure of $W(P)$ is known as block-skew-circulancy.

Let us now examine each of the blocks A, B and C. We will use the properties of γ and ϕ of (1)-(3).

Now observe that S can be decomposed into

$$S_1 = \{\gamma^0, \gamma^3, \cdots \gamma^{3(m-1)}\}\,, \tag{11}$$

$$S_2 = \{\gamma^{3m}, \gamma^{3(m+1)}, \cdots, \gamma^{3(n-1)}\}\,. \tag{12}$$

$$S_2 = \gamma^{3m} S_1 = -S_1\,. \tag{13}$$

The $n \times n$ matrix A can be decomposed as

$$A = \left[\begin{array}{c|c} A_{1,1} & A_{1,2} \\ \hline A_{2,1} & A_{2,2} \end{array}\right]\,. \tag{14}$$

where for $i,j = 1,2$, $A_{i,j}$ is the submatrix of A corresponding to the row indexing by S_i and the column indexing by S_j.

Observations

1. $A_{1,1} = A_{2,2}$
2. $A_{1,2} = A_{2,1} = A_{1,1}^*$.

We now have that

$$A = \begin{bmatrix} C_1 & C_1^* \\ C_1* & C_1 \end{bmatrix} \tag{15}$$

with $m \times m$ matrix C_1 corresponding to the indexing set S_1.

In exactly the same way, we can show that

$$B = \begin{bmatrix} C_2 & C_2^* \\ C_2^* & C_2 \end{bmatrix}, \quad C = \begin{bmatrix} C_3 & C_3^* \\ C_3^* & C_3 \end{bmatrix}. \tag{16}$$

We will refer to the above structure as conjugate skew-circulancy.

Thus, $W(P)$ is of the following structure.

$$W(P) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_2 & C_2^* & C_3 & C_3^* \\ C_1^* & C_1 & C_2^* & C_2 & C_3^* & C_3 \\ \hline C_2 & C_2^* & C_3 & C_3^* & C_1 & C_1^* \\ C_2^* & C_2 & C_3^* & C_3 & C_1^* & C_1 \\ \hline C_3 & C_3^* & C_1 & C_1^* & C_2 & C_2^* \\ C_3^* & C_3 & C_1^* & C_1 & C_2^* & C_2 \end{array}\right]. \tag{17}$$

Corollary 3 *In matrix $W(P)$, each submatrix C_1, C_2, C_3 and C_1^*, C_2^*, C_3^* is conjugate skew-circulant matrix.*

Prove

Denote the k-th row l-th column entry of an $m \times m$ matrix C_1 by $C_1(k,l)$, Then using the properties (1)-(3),

(1) For $0 \leq k, l < m-1$,

$$C_1(k,l+1) = \omega^{\phi(\gamma^{3k+3(l+1)})} = \omega^{\phi(\gamma^{3(k+1)+3l})} = C_1(k+1,l)\,,$$

(2) For $1 \leq k < m$

$$C_1(k,m-1) = \omega^{\phi(\gamma^{3k+3(m-1)})} = \omega^{\phi(\gamma^{3k-3}\gamma^{3m})} = \omega^{-\phi(\gamma^{3(k-1)})} = C_1(k-1,0)^*\,,$$

hence C_1 is conjugate skew-circulant matrix, the same as C_2 and C_3. Since each matrix C_1^*, C_2^* and C_3^* is conjugate of C_1, C_2 and C_3 separately, it is also conjugate skew-circulant matrix.

Corollary 4 *The submatrices A, B and C of the matrix $W(P)$ are skew-circulant matrices.*

Prove

We know that

$$A = \begin{bmatrix} C_1 & C_1^* \\ C_1^* & C_1 \end{bmatrix}.$$

where C_1 is conjugate skew-circulant matrix by corollary 3. Since A is 2×2 conjugate block-circulancy and by the corallory 2 it is a skew-circulant matrix. Similarly the matrices B and C are also skew-circulant matrices.

3.5 Computation with Skew-Circulant Structure

Based on the skew-circulant structure derived from the above, we will apply the matrix tensor product formulas to develop the block-diagonalization method.

Theorem 1 *For $k \times k$ block skew-circulant matrix*

$$W = \begin{bmatrix} D_0 & D_1 & \cdots & D_{k-1} \\ D_1 & D_2 & \cdots & D_0 \\ \vdots & & & \\ D_{k-1} & D_0 & \cdots & D_{k-2} \end{bmatrix}, \tag{1}$$

where W is $jk \times jk$ matrix and D_l is $j \times j$ matrix, $0 \le l < k$. then

$$W = \frac{1}{k}(F^*(k) \otimes I_j)\left(\bigoplus_{l=0}^{k-1} H_{l+1}\right)(F^*(k) \otimes I_j)\,, \tag{2}$$

where

$$H_{l+1} = D_0 + \mu^l D_1 + \mu^{2l} D_2 + \cdots + \mu^{(k-1)l} D_{k-1} = \sum_{l=0}^{k-1} \mu^{ml} D_l\,.$$

where $0 \le m < k$ and $\mu = e^{-2\pi i/k}$. $F(k)$ is one-dimensional k-point Fourier transform and $*$ is the complex conjugate. I_j is the $j \times j$ identity matrix.

Prove

First we define the $k \times k$ cyclic shift matrix S by the rule as before,

$$S\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{k-1} \end{bmatrix} = \begin{bmatrix} X_{k-1} \\ X_0 \\ \vdots \\ X_{k-2} \end{bmatrix},$$

and the skew-cyclic shift matrix $\tilde{S}$ by the rule

$$\tilde{S}\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{k-1} \end{bmatrix} = \begin{bmatrix} X_0 \\ X_{k-1} \\ \vdots \\ X_1 \end{bmatrix}.$$

Then W can be expressed by

$$\begin{aligned} W &= (I_k\tilde{S}) \otimes D_0 + (S\tilde{S}) \otimes D_1 + S^2\tilde{S} \otimes D_2 + \cdots + S^l\tilde{S} \otimes D_l \\ &= \sum_{l=0}^{k-1} (S^l\tilde{S}) \otimes D_l\,. \end{aligned} \tag{3}$$

By direct computation we can find that

$$(F(k)\otimes I_j)((I_k\tilde{S})\otimes D_0)(F(k)\otimes I_j) \;=\; (F(k)\tilde{S}F(k))\otimes D_0$$
$$= \; k\bigoplus_{m=0}^{k-1}\mu^{m\cdot 0}D_0\;,$$

and

$$(F(k)\otimes I_j)((S\tilde{S})\otimes D_1)(F(k)\otimes I_j) \;=\; (F(k)(S\tilde{S})F(k))\otimes D_1$$
$$= \; k\bigoplus_{m=0}^{k-1}\mu^{m\cdot 1}D_1\;,$$

following this way we can find

$$(4)\qquad (F(k)\otimes I_j)((S^l\tilde{S})\otimes D_l)(F(k)\otimes I_j) \;=\; (F(k)(S^l\tilde{S})F(k))\otimes D_l$$
$$= \; k\bigoplus_{m=0}^{k-1}\mu^{ml}D_l \quad 0\le l<k\;.$$

Now We compute the summation of (4) for l from 0 to $k-1$ and compare with (3) we have

$$(F(k)\otimes I_j)\,W\,(F(k)\otimes I_j) \;=\; k\bigoplus_{m=0}^{k-1}\sum_{l=0}^{k-1}\mu^{ml}D_l\;.$$

Finally we get

$$W \;=\; \frac{1}{k}(F^*(k)\otimes I_j)(\bigoplus_{m=0}^{k-1}\sum_{l=0}^{k-1}\mu^{ml}D_l)(F^*(k)\otimes I_j)\;,$$

completing the prove.

From theorem 1 we have the corollaries for $W(P)$ of variant 1 and variant 2.

Corollary 1 *For 2×2 block circulant matrix $W(P)$, where A and B are $n\times n$ matrices,*

$$(5)\qquad W(P) \;=\; \begin{bmatrix} A & B \\ B & A \end{bmatrix}\;,$$

where $W(P)$ is $2n \times 2n$ matrix, then

$$W(P) = \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} A+B & \\ & A-B \end{bmatrix} (F(2) \otimes I_n) \tag{6}$$

$$= \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_n) .$$

where $F(2) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ *is the one-dimensional 2-point Fourier transform matrix and I_n is the $n \times n$ identity matrix.*

Since A and B are skew-circulant matrices and $B = A^*$, so $H_1 = A+B = A+A^*$ will be pure real skew-circulant matrix. However $H_2 = A-B = A-A^*$ will be pure imaginary matrix and has the structure like

$$H_2 = \begin{bmatrix} h_0 & h_1 & \cdots & h_{n-1} \\ h_1 & h_2 & \cdots & -h_0 \\ \vdots & & & \\ h_{n-1} & -h_0 & \cdots & -h_{n-2} \end{bmatrix} . \tag{7}$$

We call the matrix H_2 as a negative-skew-circulant matrix. It can also be changed into a skew-circulant matrix again. We will describe the efficient methods of computation of the skew-circulant matrix in section 3.6.

Denote the 3-point Fourier transform matrix

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & \mu & \mu^2 \\ 1 & \mu^2 & \mu \end{bmatrix} \tag{8}$$

by $F(3)$, where $\mu = e^{\frac{-2\pi i}{3}}$.

Corollary 2 *For $2m \times 2m$ matrices A,B and C, let $\mu = e^{\frac{-2\pi i}{3}}$ denote a primitive cube root of 1 and n=2m*

$$W(P) = \begin{bmatrix} A & B & C \\ B & C & A \\ C & A & B \end{bmatrix} \tag{9}$$

then

$$W(P) = \frac{1}{3}(F^*(3) \otimes I_n) \begin{bmatrix} A+B+C & & \\ & A+\mu B+\mu^2 & \\ & & A+\mu^2 B+\mu C \end{bmatrix} (F^*(3) \otimes I_n)$$

$$= \frac{1}{3}(F^*(3) \otimes I_n) \begin{bmatrix} H_1 & & \\ & H_2 & \\ & & H_3 \end{bmatrix} (F^*(3) \otimes I_n)\,. \tag{10}$$

where $F(3)$ is the 1-dimensional 3-point Fourier transform matrix and I_n is the $n \times n$ identity matrix.

Observations

According to the corollary 4 of section 3.4, the matrices A, B and C are skew-circulant matrices. Since the matrices H_1, H_2 and H_3 are linear combinations of skew-circulant matrices A, B and C, Then the matrices

$$H_1 = A+B+C\,, \tag{11}$$

$$H_2 = A+\mu B+\mu^2 C\,, \tag{12}$$

$$H_3 = A+\mu^2 B+\mu C\,, \tag{13}$$

are skew-circulant matrices.

Conjugate-skew-circulancy of H_1, H_2 and H_3 also comes from the fact that they are linear combinations of conjugate-skew-circulancy of matrices A, B and C. Thus we may repeat our procedure with H_1, H_2 and H_3 to further decompose the matrices. For example

$$(14) \qquad H_1 = A+B+C = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

$$= \begin{bmatrix} C_1+C_2+C_3 & C_1^*+C_2^*+C_3^* \\ C_1^*+C_2^*+C_3^* & C_1+C_2+C_3 \end{bmatrix}$$

$$= \frac{1}{2}(F(2)\otimes I_m)\begin{bmatrix} H_{11}+H_{11}^* & \\ & H_{11}-H_{11}^* \end{bmatrix}(F(2)\otimes I_m)\,,$$

where $H_{11}+H_{11}^*$ is real skew-circulant matrix of cosine function and $H_{11}-H_{11}^*$ is negative-skew-circulant matrix of pure imaginary of sine function which can be changed to skew-circulant matrix again. Similarly H_2 and H_3 also can be diagonalized further like H_1.

Here, we have introduced the method of computation for skew-circulant structure, namely, block-diagonalization. The method of block-diagonalization derived from the multiplicative structure has the following advantages.

1. Using the group theoretic structures, one arrives at skew-circulant blocks of varing sizes. This flexibility is valuable in matching the vector register sizes in vector facility.

2. Since the skew-circulant blocks are located diagonally, they are independent of each other. This independency lends easily to parallel processing with multi-processors. (The number and size of skew-circulant blocks are controlled again by the multiplicative group structure.)

In following sections we will describe the methods for computing the skew-circulant matrix and the twiddle factor $(F^*(3) \otimes I_n)$ or $(F(2) \otimes I_n)$ in different machine architectures.

3.6 Computation of Skew-Circulant Matrix

In this section we will describe the method how to compute the multiplication of a $n \times n$ skew-circulant matrix $\tilde{C}$ with a vector of order n which has been appeared in this chapter before.

First we can use well-known convolution theorem [20]. By (9) of section 3.2 we change the skew-circulant matrix $\tilde{C}$ to the circulant matrix C by

$$\tilde{C} = C\tilde{S} \tag{1}$$

where $\tilde{S}$ is the $n \times n$ skew-cyclic shift matrix shown in (10) of section 3.2.

Then we can use the convolution theorem since the multiplication of the $n \times n$ circulant matrix C with a vector of order n is the cyclic convolution. It means that the $n \times n$ skew-circulant matrix can be computed by using n-point Fourier transform. By convolution theorem we have

$$C = F(n)^{-1} D F(n) , \tag{2}$$

where $F(n)$ is the n-point Fourier transform matrix and D is a diagonal matrix

$$D = diag(F(n)\underline{C}) . \tag{3}$$

where $\underline{C}$ is a vector of the first column of C.

Observe that

$$F(n)\tilde{S} = F(n)^* = nF(n)^{-1} , \tag{4}$$

hence we can compute the skew-circulant matrix $\tilde{C}$ directly by

$$\tilde{C} = C\tilde{S} = F(n)^{-1}DF(n)\tilde{S} = nF(n)^{-1}DF(n)^{-1} \ . \tag{5}$$

the above formula (5) is expressed as inverse Fourier transform matrix $F(n)^{-1}$. It also can be expressed as forward Fourier transform matrix $F(n)$ by

$$\tilde{C} \ = \ F(n)D'F(n) \ , \tag{6}$$

where D' is the diagonal matrix determined by

$$D' \ = \ \frac{1}{n} diag\,(F^{*}(n)\underline{C}) \ . \tag{7}$$

Secondly, we will present the block-diagonalized method which block-diagonalize the original skew-circulant matrix to a variation of the convolution theorem via the matrix tensor product formulas. Let $\tilde{C}$ is $n \times n$ matrix and $n = 2^{\alpha}q$. By the block-diagonalized method of theorem 1 of section 3.5, we can decompose the matrix $\tilde{C}$ by α steps. Each step will produce 2×2 block-diagonal matrix, one is skew-circulant matrix again and another is negative-skew-circulant matrix.

The negative-skew-circulant matrix can be changed to the skew-circulant matrix again by the following ways:

(i) If G is a $n \times n$ negative-skew-circulant matrix,

$$G \ = \ \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-1} \\ g_1 & g_2 & \cdots & -g_0 \\ \vdots & & & \\ g_{n-1} & -g_0 & \cdots & -g_{n-2} \end{bmatrix} , \tag{8}$$

then it can be diagonized as follows,

$$G = \begin{bmatrix} 1 & & & \\ & \beta & & \\ & & \ddots & \\ & & & \beta^{n-1} \end{bmatrix} \begin{bmatrix} g_0 & -\beta^{n-1}g_1 & \cdots & -\beta g_{n-1} \\ -\beta^{n-1}g_1 & -\beta^{n-2}g_2 & \cdots & g_0 \\ \vdots & & & \\ -\beta g_{n-1} & g_0 & \cdots & -\beta^2 g_{n-2} \end{bmatrix} \begin{bmatrix} 1 & & & \\ & \beta & & \\ & & \ddots & \\ & & & \beta^{n-1} \end{bmatrix} \tag{9}$$

where $\beta^n = -1$. After diagonalized, the middle matrix is a skew-circulant matrix again.

When n is odd number, it will be very easy since $\beta = -1$ now. Thus the formula (9) has been changed to

$$G = \begin{bmatrix} 1 & & & \\ & -1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \begin{bmatrix} g_0 & -g_1 & \cdots & g_{n-1} \\ -g_1 & g_2 & \cdots & g_0 \\ \vdots & & & \\ g_{n-1} & g_0 & \cdots & -g_{n-2} \end{bmatrix} \begin{bmatrix} 1 & & & \\ & -1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}.$$

(ii) When $n = pq = 2^\alpha q$, G can be rewrited as a block negative-skew-circulant matrix,

$$G = \begin{bmatrix} D_0 & D_1 & \cdots & D_{q-1} \\ D_1 & D_2 & \cdots & -D_0 \\ \vdots & & & \\ D_{q-1} & -D_0 & \cdots & -D_{q-2} \end{bmatrix}, \tag{10}$$

where D_i is $P \times P$ matrix. Then G can be changed to block skew-circulant matrix again,

$$G = \begin{bmatrix} I_p & & & \\ & -I_p & & \\ & & \ddots & \\ & & & I_p \end{bmatrix} \begin{bmatrix} D_0 & -D_1 & \cdots & D_{q-1} \\ -D_1 & D_2 & \cdots & D_0 \\ \vdots & & & \\ D_{q-1} & D_0 & \cdots & -D_{q-2} \end{bmatrix} \begin{bmatrix} I_p & & & \\ & -I_p & & \\ & & \ddots & \\ & & & I_p \end{bmatrix} \tag{11}$$

(iii) The matrix G can be changed to the difference of one skew-circulant matrix G' and another special matrix, named the triangle matrix T,

$$(12) \qquad G = G' - T = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-1} \\ g_1 & g_2 & \cdots & g_0 \\ \vdots & & & \\ g_{n-1} & g_0 & \cdots & g_{n-2} \end{bmatrix} - 2 \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & g_0 \\ \vdots & & & \\ 0 & g_0 & \cdots & g_{n-2} \end{bmatrix}.$$

The skew-circulant matrix $\tilde{C}$ which has appeared in our algorithm is either pure real number or pure imaginary number. The trade-off of method (i), (ii) and (iii) is dependent on different kinds of skew-circulant matrices and the different machine architectures.

There are several algorithms for computing with skew-circulant matrices. By incorporating the different methods of computing skew-circulant matrices to the matrix $W(P)$, we can get our efficient multiplicative algorithm for different machine architectures.

3.7 Computation of $(F(2) \otimes I_n)$ or $(F^*(3) \otimes I_n)$

Applying the tensor products and permutations which has introduced in section 2.5, we can get the method of vector or parallel operations for the computations of $(F(2) \otimes I_n)$ or $(F^*(3) \otimes I_n)$.

We know that $F(2)$ is the one-dimensional 2-point Fourier transform matrix and $F^*(3)$ is the conjugate of 3-point Fourier transform matrix

$$(1) \qquad F(2) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad F^*(3) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \mu & \mu^2 \\ 1 & \mu^2 & \mu \end{bmatrix},$$

where $\mu = e^{\frac{-2\pi i}{3}}$ and $*$ is the complex conjugation.

Let I_n is the $n \times n$ identity matrix, then $(F(2) \otimes I_n)$ will be $2n \times 2n$ matrix

$$F(2) \otimes I_n = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes I_n = \begin{bmatrix} I_n & I_n \\ I_n & -I_n \end{bmatrix}$$

$$= \left[\begin{array}{ccc|ccc} 1 & & & 1 & & \\ & 1 & & & 1 & \\ & & \ddots & & & \ddots \\ & & 1 & & & 1 \\ \hline 1 & & & -1 & & \\ & 1 & & & -1 & \\ & & \ddots & & & \ddots \\ & & 1 & & & -1 \end{array}\right] \tag{2}$$

By commutative law of the tensor products mentioned in theorem 7 of section 2.5, the tensor product $(F(2) \otimes I_n)$ or $(F^*(3) \otimes I_n)$ can be interchanged into another forms $(I_n \otimes F(2))$ or $(I_n \otimes F^*(3))$ which are better for parallelization. That is

$$F(2) \otimes I_n = P(I_n \otimes F(2))P^{-1}\,, \tag{3}$$

where P and P^{-1} are permutation matrices, $P = P(2n, 2)$ and $P^{-1} = P(2n, n)$, P^{-1} is the inverse matrix of matrix P. $(I_n \otimes F(2))$ is the direct sum of n copies of $F(2)$, i.e. it is a block diagonalized matrix of dimension $2n$ with each block is $F(2)$. Its structure is given by

$$I_n \otimes F(2) = \begin{bmatrix} F(2) & & & \\ & F(2) & & \\ & & \ddots & \\ & & & F(2) \end{bmatrix}, \tag{4}$$

Similarly the structure of $I_n \otimes F^*(3)$ is given by

$$(5) \qquad I_n \otimes F^*(3) = \begin{bmatrix} F^*(3) & & & \\ & F^*(3) & & \\ & & \ddots & \\ & & & F^*(3) \end{bmatrix}.$$

3.8 An Example

We will go through our algorithms using computation of the 2-dimensional 5-point Fourier transform as an example. The indexing set $\mathcal{Z}/5 \times \mathcal{Z}/5$ will be identified with the field $\mathcal{Z}/5[\rho]$. $\gamma = 1 + 4\rho$ is a generator of the multiplicative cyclic group of nonzero elements of $\mathcal{Z}/5[\rho]$.

$$(1) \qquad \begin{array}{ll} \gamma = 1 + 4\rho & \gamma^{13} = 4 + \rho \\ \gamma^2 = 2\rho & \gamma^{14} = 3\rho \\ \gamma^3 = 2 + 4\rho & \gamma^{15} = 3 + \rho \\ \gamma^4 = 1 + \rho & \gamma^{16} = 4 + 4\rho \\ \gamma^5 = 2 + \rho & \gamma^{17} = 3 + 4\rho \\ \gamma^6 = 3 & \gamma^{18} = 2 \\ \gamma^7 = 3 + 2\rho & \gamma^{19} = 2 + 3\rho \\ \gamma^8 = \rho & \gamma^{20} = 4\rho \\ \gamma^9 = 1 + 2\rho & \gamma^{21} = 4 + 3\rho \\ \gamma^{10} = 3 + 3\rho & \gamma^{22} = 2 + 2\rho \\ \gamma^{11} = 1 + 3\rho & \gamma^{23} = 4 + 2\rho \\ \gamma^{12} = 4 & \gamma^{24} = \gamma^0 = 1 \end{array}$$

The input and output data will be ordered by the above.

Now the two-dimensional finite Fourier transform of $\underline{X}$ is

$$\underline{Y} = W(5)\,\underline{X} + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}, \tag{2}$$

W(5) with respect to the multiplicative ordering is

$$W(5) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \le j,k < 24} \tag{3}$$

where $\omega = e^{-2\pi i/5}$.

Variant 1

The indexed set $R(5)$ is obtained from $\{\gamma^j\}$, with j taken from the following ordered set,

$$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11;\ 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23;\}\,.$$

Then we have

$$W(5) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix}, \tag{4}$$

where $W(5)$ is 2×2 block skew-circulant matrix and A and A^* are conjugate skew-circulant matrices,

$$
(5) \qquad A = \begin{bmatrix}
\omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega \\
\omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 \\
1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 \\
\omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 \\
\omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 \\
\omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 \\
\omega^3 & \omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 \\
\omega^3 & 1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 \\
1 & \omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 \\
\omega & \omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 \\
\omega^3 & \omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 \\
\omega & \omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2
\end{bmatrix},
$$

and A^* is conjugate of A.

Applying corollary 1 of 3.5 to $W(5)$, we have

$$
(6) \qquad W(5) = \frac{1}{2}(F(2) \otimes I_{12}) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_{12}),
$$

where $H_1 = A + A^*$ is skew-circulant matrix of pure real number. $H_2 = A - A^*$ is negative skew-circulant matrix of pure imaginary. It can be diagonalized again.

Variant 2

Now we choose the another ordering of indexing set $R(P)$ from $\{\gamma^j\}$, with j taken from the following ordered set,

$$
\{0,3,6,9,12,15,18,21;\ 8,11,14,17,20,23,26,29;\ 16,19,22,25,28,31,34,37;\}.
$$

Then we have

$$(1) \qquad W(5) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_2 & C_2^* & C_3 & C_3^* \\ C_1^* & C_1 & C_2^* & C_2 & C_3^* & C_3 \\ \hline C_2 & C_2^* & C_3 & C_3^* & C_1 & C_1^* \\ C_2^* & C_2 & C_3^* & C_3 & C_1^* & C_1 \\ \hline C_3 & C_3^* & C_1 & C_1^* & C_2 & C_2^* \\ C_3^* & C_3 & C_1^* & C_1 & C_2^* & C_2 \end{array}\right],$$

$W(5)$ is block skew-circulant matrix where C_1, C_2 and C_3 are conjugate skew-circulant matrices,

$$(2) \quad C_1 = \begin{bmatrix} \omega & \omega^2 & \omega^3 & \omega \\ \omega^2 & \omega^3 & \omega & \omega^4 \\ \omega^3 & \omega & \omega^4 & \omega^3 \\ \omega & \omega^4 & \omega^3 & \omega^2 \end{bmatrix}, \; C_2 = \begin{bmatrix} 1 & \omega & 1 & \omega^3 \\ \omega & 1 & \omega^3 & 1 \\ 1 & \omega^3 & 1 & \omega^4 \\ \omega^3 & 1 & \omega^4 & 1 \end{bmatrix}, \; C_3 = \begin{bmatrix} \omega^4 & \omega^2 & \omega^2 & \omega \\ \omega^2 & \omega^2 & \omega & \omega \\ \omega^2 & \omega & \omega & \omega^3 \\ \omega & \omega & \omega^3 & \omega^3 \end{bmatrix}.$$

Applying the corollary 2 of 3.5 we get

$$(3) \qquad W(5) = \frac{1}{3}[F^*(3) \otimes I_8] \begin{bmatrix} H_1 & & \\ & H_2 & \\ & & H_3 \end{bmatrix} [F^*(3) \otimes I_8],$$

where H_1, H_2 and H_3 are skew-circulant matrices.

$$H_1 = A + B + C = \begin{bmatrix} C_1 + C_2 + C_3 & C_1^* + C_2^* + C_3^* \\ C_1^* + C_2^* + C_3^* & C_1 + C_2 + C_3 \end{bmatrix},$$

$$H_2 = A + \mu B + \mu^2 C = \begin{bmatrix} C_1 + \mu C_2 + \mu_2 C_3 & C_1^* + \mu C_2^* + \mu^2 C_3^* \\ C_1^* + \mu C_2^* + \mu^2 C_3^* & C_1 + \mu C_2 + \mu^2 C_3 \end{bmatrix},$$

and

$$H_3 = A + \mu^2 B + \mu C = \begin{bmatrix} C_1 + \mu^2 C_2 + \mu C_3 & C_1^* + \mu^2 C_2^* + \mu C_3^* \\ C_1^* + \mu^2 C_2^* + \mu C_3^* & C_1 + \mu^2 C_2 + \mu C_3 \end{bmatrix}.$$

Since H_1, H_2 and H_3 are conjugate-skew-circulancy, it can be block-diagonalized further to get more efficient results.

3.9 New Algorithm with Circulant Structure

In the above sections of this chapter we have developed a multiplicative $P \times P$ two-dimensional Fourier transform algorithm for a prime $P \equiv 2 \; mod \; 3$ which has a block-diagonal structure with each block of skew-circulant matrix. The algorithm is based on the field structure determined by the irreducible polynomial x^2+x+1 over $\mathcal{Z}/P$ for $P \equiv 2 \; mod \; 3$. Efficient algorithm should be measured upon implementation. Data flow and complexity of Fourier transform are controlled by the algebraic nature of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$. It is reasonable to have several ways of controlling them so as to suit them efficiently on varying architectures.

From this section we will use the same multiplicative property of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$, which is used to control the data flow in the Fourier transform computation. But the difference is that we will use another kind of initial stage of ordering $\mathcal{Z}/P \times \mathcal{Z}/P$. After that we will get a circulant structure which has block-diagonal structure with each block of circulant matrix. This is an alternative to the skew-circulant structure described in the above sections. We know that the multiplication of a $P \times P$ circulant matrix and a vector of size P is the matrix form of cyclic convolution. There are many efficient algorithms we can use to calculate the cyclic convolution on different machine architectures [1,13,23]. This is one of the reasons why we want to find circulant structure comparing with skew-circulant matrix. After that we can use the existing efficient algorithms of cyclic convolution to compute a set of circulant matrices which located diagonally.

Now we will take quick review on the field of P^2 elements for $P \equiv 2(3)$ and Fourier transform matrix as mentioned in section 3.3. For $P \equiv 2(3)$ the polynomial $x^2 + x + 1$ is irreducible over $\mathcal{Z}/P$. Set $\rho = exp(-2\pi i/3)$. Then $\rho^2 + \rho + 1 = 0$ and $\mathcal{Z}/P[\rho]$ is a field with P^2 elements; i.e., the nonzero elements of $\mathcal{Z}/P[\rho]$ form a cyclic group under multiplication. Let γ be a cyclic generator. Then

$$1 = \gamma^0, \gamma, \gamma^2, \cdots, \gamma^{P^2-2} \tag{1}$$

are the distinct nonzero elements of $\mathcal{Z}/P[\rho]$.

An element $a \in \mathcal{Z}/P[\rho]$ can be written uniquely as $a_1 + \rho a_2$, $a_1, a_2 \in \mathcal{Z}/P$. Arithmetic in $\mathcal{Z}/P[\rho]$ is defined by

$$(a_1 + \rho a_2) + (b_1 + \rho b_2) \;=\; (a_1 + b_1) + \rho(a_2 + b_2)\,. \tag{2}$$

$$(a_1 + \rho a_2)(b_1 + \rho b_2) \;=\; a_1 b_1 - a_2 b_2 + \rho(a_2 b_1 + a_1 b_2 - a_2 b_2)\,.$$

Define a mapping $\phi : \mathcal{Z}/P[\rho] \rightarrow \mathcal{Z}/P$ by

$$\phi(a_1 + \rho a_2) \;=\; a_1\,. \tag{3}$$

Observe that

$$\phi(a + b) \;=\; \phi(a) + \phi(b)\,, \tag{4}$$

$$\phi(ab) \;=\; a_1 b_1 - a_2 b_2\,.$$

Then we have the same properties of generator γ and the mapping function ϕ are existed as (5),(6) of section 3.3 and (1)-(3) of section 3.4.

For a function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$, the Fourier transform of X now is

$$(5) \qquad FX(b_1, -b_2) = \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1, a_2) e^{\frac{-2\pi i}{P}(a_1 b_1 - a_2 b_2)}$$

$$= \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1, a_2) e^{\frac{-2\pi i}{P}\phi(ab)} .$$

Note that $(a_1, a_2) \in \mathcal{Z}/P \times \mathcal{Z}/P$ corresponds uniquely to $a_1 + \rho a_2 \in \mathcal{Z}/P[\rho]$. Hence an ordering of $\mathcal{Z}/P[\rho]$ will yield an ordering for $\mathcal{Z}/P \times \mathcal{Z}/P$. Take the ordering by $U(P)$, where $U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$.

Now we may view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent (5) in terms of matrices as

$$(6) \qquad \begin{bmatrix} Y(0) \\ \underline{Y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix} \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix},$$

where

$$(7) \qquad W(P) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0\le j,k<P^2-1},$$

and the element $0 \in \mathcal{Z}/P[\rho]$ has been placed in front of $U(P)$. For the rest of our discussion, we ignore $0 \in \mathcal{Z}/P[\rho]$. (We can place this at the last stage of computations.)

Variant 1

We assume already that P is a prime number and $P \equiv 2(3)$. By the lemma 1 of section 3.4, $P^2 - 1$ is divisible by 12. Set $m = \frac{P^2-1}{6}$, and define the set

$$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{3m-1}\} . \tag{8}$$

Thus we have that $1, \gamma^{3m}$ is the quotient of $U(P)/S$ and

$$U(P) = S \cup \gamma^{3m} S . \tag{9}$$

We can rewrite the set S as

$$S' = \{\gamma^{3m-1}, \gamma^{3m-2}, \cdots, \gamma, \gamma^0\} . \tag{10}$$

Then $U(P)$ can be rewritten as

$$U(P) = \gamma^{3m} S' \cup S' . \tag{11}$$

View S and S' as ordered by (8) and (10). These orderings can also be used to order $\gamma^{3m}S$ and $\gamma^{3m}S'$. We will denote by $R_1(P)$ with $S; \gamma^{3m}S$ and $R_2(P)$ with $\gamma^{3m}S'; S'$ instead of the original $U(P)$. We order the column indexing by $R_1(P)$ and order the row indexing by $R_2(P)$. Now we examine the matrix $W(P)$.

Theorem 1 *$W(P)$ is of the form*

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} , \tag{12}$$

where A and B are $3m \times 3m$ conjugate circulant matrices with $A = B^*$. The conjugate of B is denoted by B^*.

Proof

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 \\ W_3 & W_4 \end{bmatrix},$$

where W_i, $1 \leq i \leq 4$ is an $3m \times 3m$ matrix. For $0 \leq k, l < 3m$, denote the k-th row l-th column entry of an $3m \times 3m$ matrix W_i by $W_i(k,l)$. Then using the properties of generator γ and the mapping function ϕ shown in (1)-(3) of section 3.4,

$$W_4(k,l) = \omega^{\phi(\gamma^{6m-1-k}\gamma^{3m}\gamma^l)} = \omega^{\phi(\gamma^{3m}\gamma^{6m-1-k}\gamma^l)} = W_1(k,l) .$$

$$W_3(k,l) = \omega^{\phi(\gamma^{6m-1-k}\gamma^l)} = \omega^{\phi(\gamma^{3m}\gamma^{6m-1-k}\gamma^{3m}\gamma^l)} = W_2(k,l) .$$

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks A and B.

By the properties of γ and ϕ, we know that $\gamma^{(3m+j)} = -\gamma^j$. Thus matrix A equals to the conjugate of B, that is $A = B^*$. $W(P)$ will be 2×2 block circulant matrix with $A = B^*$, that is

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix},$$

where $*$ we mean complex conjugation.

Now we look each submatrix A and A^*. Denote the k-th row l-th column entry of the $3m \times 3m$ matrix A by $A(k,l)$,

(1) For $0 \leq k, l < 3m - 1$,

$$A(k+1, l+1) = \omega^{\phi(\gamma^{3m}\gamma^{3m-1-(k+1)}\gamma^{l+1})} = \omega^{\phi(\gamma^{3m}\gamma^{3m-1-k})\gamma^l)} = A(k,l) ,$$

(2) For $0 \leq l < 3m - 1$,

$$A(3m-1, l) = \omega^{\phi(\gamma^{6m-(3m-1)+l})} = \omega^{\phi(\gamma^{l+1}\gamma^{3m})} = \omega^{-\phi(\gamma^{l+1})} = A(0, l+1)^* ,$$

hence A is conjugate circulant matrix. Since A^* is the conjugate of A, A^* is also a conjugate circulant matrix.

Corollary 1 *If matrix $W(P)$ is 2×2 block-circulancy, that is*

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix}, \tag{13}$$

where A and A^* are conjugate circulant matrices, then $W(P)$ is a circulant matrix.

Prove

We know that $W(P)$ is $6m \times 6m$ matrix,

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix}.$$

We can rewrite

$$W(P) = \left[\begin{array}{cccc|ccc} a_{0,0} & a_{0,1} & \cdots & a_{0,3m-1} & a_{0,3m} & \cdots & a_{0,6m-1} \\ \vdots & & & & \vdots & & \\ a_{3m-1,0} & a_{3m-1,1} & \cdots & a_{3m-1,3m-1} & a_{3m-1,3m} & \cdots & a_{3m-1,6m-1} \\ \hline a_{3m,0} & a_{3m,1} & \cdots & a_{3m,3m-1} & a_{3m,3m} & \cdots & a_{3m,6m-1} \\ \vdots & & & & \vdots & & \\ a_{6m-1,0} & a_{6m-1,1} & \cdots & a_{6m-1,3m-1} & a_{6m-1,3m} & \cdots & a_{6m-1,6m-1} \end{array}\right],$$

where $m = \frac{P^2-1}{6}$.

Now we look at the matrix $W(P)$. Since $a_{0,1} = a^*_{3m,1}$ and $a_{3m,1} = a^*_{6m-1,0}$, thus $a_{0,1} = a_{6m-1,0}$. In general

$$a_{0,j} = a^*_{3m,j}, \qquad a_{3m,j} = a^*_{6m-1,j-1},$$

we have

$$a_{0,j} = a_{6m-1,j-1}\ , 1 \leq j < 6m\ ,$$

and

$$a_{i,j} = a_{i-1,j-1} \qquad 1 \leq i < 6m\ , 1 \leq j < 6m\ .$$

Hence $W(P)$ is circulant matrix.

Variant 2

For $P \equiv 2(3)$, $P^2 - 1$ is divisible by 12. Set $m = \frac{P^2-1}{6}$ and $n = 2m$. The element γ^3 is of order n and generates the subgroup

$$S = \{\gamma^0, \gamma^3, \gamma^6, \cdots, \gamma^{3(n-1)}\} \tag{14}$$

For the simplicity of presentation, we will assume that n is not divisible by 3 again.(i.e., that $P^2 - 1$ is not divisible by 9. Otherwise the method we present here can be modified to yield the desired results.) Thus we have that $\{1, \gamma^n, \gamma^{2n}\}$ is the quotient of $U(P)/S$, where $U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$, and

$$U(P) = S \cup \gamma^n S \cup \gamma^{2n} S\ . \tag{15}$$

Now we rewrite the subgroup S as

$$S' = \{\gamma^{3(n-1)}, \gamma^{3(n-2)}, \cdots, \gamma^3, \gamma^0\}\ . \tag{16}$$

Then $U(P)$ can be rewritten as

$$U(P) = \gamma^{2n} S' \cup \gamma^n S' \cup S'\ . \tag{17}$$

View S and S' as ordered by (14) and (16). These orderings can be used to order $\gamma^n S, \gamma^{2n} S$ and $\gamma^{2n} S', \gamma^n S'$. We will denote by $R_1(P)$ with $S; \gamma^n S; \gamma^{2n} S$

and $R_2(P)$ with $\gamma^{2n}S'; \gamma^n S'; S'$ instead of the original $U(P)$. We will order the column indexing by $R_1(P)$ and order the row indexing by $R_2(P)$. Now we examine the matrix $W(P)$.

Theorem 2 *$W(P)$ will be of the form*

$$\begin{bmatrix} A & C & B \\ B & A & C \\ C & B & A \end{bmatrix} \tag{18}$$

where A, B and C are $n \times n$ matrices.

Prove

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 & W_3 \\ W_4 & W_5 & W_6 \\ W_7 & W_8 & W_9 \end{bmatrix},$$

where W_i, $1 \leq i \leq 9$ is an $n \times n$ matrix. We will show that $W_2 = W_6 = W_7$. The other cases are proved in exactly the same way. For $0 \leq k, l < n$, denote the k-th row l-th column entry of an $n \times n$ matrix M by $M(k, l)$,

$$W_2(k,l) = \omega^{\phi(\gamma^{2n}\gamma^{3(n-1)-3k}\gamma^n\gamma^{3l})} = \omega^{\phi(\gamma^{3(n-1)-3k+3l})} = W_7(k,l)\,.$$

In view of Lemma 1 of section 3.3, this is also $W_6(k, l)$, completing the theorem 2.

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks A, B and C. We will use the properties of γ and ϕ of (1)-(3) of section 3.4 again.

Now observe that S can be decomposed into

(19) $$S_1 = \{\gamma^0, \gamma^3, \cdots \gamma^{3(m-1)}\},$$

(20) $$S_2 = \{\gamma^{3m}, \gamma^{3(m+1)}, \cdots, \gamma^{3(n-1)}\}.$$

(21) $$S_2 = \gamma^{3m} S_1 = -S_1.$$

and S' can be decomposed into

(22) $$S'_1 = \{\gamma^{3(n-1)}, \gamma^{3(n-2)}, \cdots, \gamma^{3m}\},$$

(23) $$S'_2 = \{\gamma^{3(m-1)}, \gamma^{3(m-2)}, \cdots, \gamma^{0}\},$$

(24) $$S'_2 = \gamma^{3m} S'_1 = -S'_1.$$

The $n \times n$ matrix A can be decomposed as

(25) $$A = \left[\begin{array}{c|c} A_{1,1} & A_{1,2} \\ \hline A_{2,1} & A_{2,2} \end{array}\right].$$

where for $i, j = 1, 2$, $A_{i,j}$ is the submatrix of A corresponding to the row indexing by S_i and the column indexing by S'_j.

Observations

1. $A_{1,1} = A_{2,2}$
2. $A_{1,2} = A_{2,1} = A^*_{1,1}$.

We now have that

(26) $$A = \begin{bmatrix} C_1 & C_1^* \\ C_1* & C_1 \end{bmatrix}$$

with $m \times m$ matrix C_1 corresponding to the indexing set S_1 in row and $\gamma^{2n}S_1'$ in column.

In exactly the same way, we can show that

$$B = \begin{bmatrix} C_2 & C_2^* \\ C_2^* & C_2 \end{bmatrix}, \quad C = \begin{bmatrix} C_3 & C_3^* \\ C_3^* & C_3 \end{bmatrix}. \tag{27}$$

We will refer to the above structure as conjugate circulancy.

Thus, $W(P)$ is of the following structure.

$$W(P) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_3 & C_3^* & C_2 & C_2^* \\ C_1^* & C_1 & C_3^* & C_3 & C_2^* & C_2 \\ \hline C_2 & C_2^* & C_1 & C_1^* & C_3 & C_3^* \\ C_2^* & C_2 & C_1^* & C_1 & C_3^* & C_3 \\ \hline C_3 & C_3^* & C_2 & C_2^* & C_1 & C_1^* \\ C_3^* & C_3 & C_2^* & C_2 & C_1^* & C_1 \end{array}\right]. \tag{28}$$

Corollary 2 *In matrix $W(P)$, each submatrix C_1, C_2, C_3 and C_1^*, C_2^*, C_3^* is conjugate circulant matrix.*

Prove

Denote the k-th row l-th column entry of an $m \times m$ matrix C_1 by $C_1(k,l)$,

(1) For $0 \leq k, l < m-1$,

$$C_1(k+1, l+1) = \omega^{\phi(\gamma^{2n}\gamma^{3(n-1)-3(k+1)}\gamma^{3(l+1)})} = \omega^{\phi(\gamma^{2n}\gamma^{3(n-1)-3k}\gamma^{3l})} = C_1(k,l)\,,$$

(2) For $0 \leq l < m-1$,

$$C_1(m-1, l) = \omega^{\phi(\gamma^{2n}\gamma^{3(n-1)-3(m-1)}\gamma^{3l})} = \omega^{-\phi(\gamma^{2n}\gamma^{3(n-1)}\gamma^{3(l+1)})} = C_1(0, l+1)^*\,.$$

hence C_1 is conjugate circulant matrix, the same as C_2 and C_3. Since each matrix C_1^*, C_2^* and C_3^* is conjugate of C_1, C_2 and C_3 separately, it is also the conjugate circulant matrix.

Corollary 3 *The submatrices A, B and C of the matrix $W(P)$ are circulant matrices.*

Prove

We know that

$$A = \begin{bmatrix} C_1 & C_1^* \\ C_1^* & C_1 \end{bmatrix}.$$

Since A is 2×2 conjugate block-circulancy, by the corallory 1 it is a circulant matrix. Similarly the matrices B and C are also circulant matrices.

3.10 Computation with Circulant Structure

Based on the circulant structure derived from the above, we will apply the matrix tensor product formulas to develop the block-diagonalization method.

Theorem 1 *For $k \times k$ block circulant matrix*

$$W = \begin{bmatrix} D_0 & D_{k-1} & \cdots & D_1 \\ D_1 & D_0 & \cdots & D_2 \\ \vdots & & & \\ D_{k-1} & D_{k-2} & \cdots & D_0 \end{bmatrix}, \tag{1}$$

where W is $jk \times jk$ matrix and D_l is $j \times j$ matrix, $0 \leq l < k$. then

$$W = \frac{1}{k}(F(k) \otimes I_j)(\bigoplus_{l=0}^{k-1} H_{l+1})(F^*(k) \otimes I_j), \tag{2}$$

where

$$H_{l+1} = D_0 + \mu^l D_1 + \mu^{2l} D_2 + \cdots + \mu^{(k-1)l} D_{k-1} = \sum_{l=0}^{k-1} \mu^{ml} D_l \,.$$

where $0 \le m < k$ *and* $\mu = e^{-2\pi i/k}$. $F(k)$ *is one-dimensional* k*-point Fourier transform and* $*$ *is the complex conjugate.* I_j *is the* $j \times j$ *identity matrix.*

Prove

similarly like the proving of the theorem 1 of section 3.5, we define the $k \times k$ cyclic shift matrix S by the rule as before,

$$S \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{k-1} \end{bmatrix} = \begin{bmatrix} X_{k-1} \\ X_0 \\ \vdots \\ X_{k-2} \end{bmatrix},$$

Then W can be expressed by

$$W = I_k \otimes D_0 + S \otimes D_1 + S^2 \otimes D_2 + \cdots + S^l \otimes D_l \tag{3}$$
$$= \sum_{l=0}^{k-1} S^l \otimes D_l \,.$$

By direct computation we can find that

$$(F^*(k) \otimes I_j)((S^l \otimes D_l)(F(k) \otimes I_j) = (F^*(k) S^l F(k)) \otimes D_l \tag{4}$$
$$= k \bigoplus_{m=0}^{k-1} \mu^{ml} D_l \quad 0 \le l < k \,.$$

Now We compute the summation of (4) for l from 0 to $k-1$ and compare with (3) we have

$$(F^*(k) \otimes I_j)\, W\, (F(k) \otimes I_j) = k \bigoplus_{m=0}^{k-1} \sum_{l=0}^{k-1} \mu^{ml} D_l \,.$$

Finally we get

$$W = \frac{1}{k}(F(k) \otimes I_j)(\bigoplus_{m=0}^{k-1} \sum_{l=0}^{k-1} \mu^{ml} D_l)(F^*(k) \otimes I_j) ,$$

completing the prove.

From theorem 1 we have the corollaries for $W(P)$ of variant 1 and variant 2.

Corollary 1 *For 2×2 block circulant matrix $W(P)$, where A and B are $n \times n$ matrices,*

$$W(P) = \begin{bmatrix} A & B \\ B & A \end{bmatrix} , \tag{5}$$

where $W(P)$ is $2n \times 2n$ matrix, then

$$\begin{aligned} W(P) &= \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} A+B & \\ & A-B \end{bmatrix} (F(2) \otimes I_n) \\ &= \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_n) . \end{aligned} \tag{6}$$

where $F(2) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ is the one-dimensional 2-point Fourier transform matrix and I_n is the $n \times n$ identity matrix.

Since A and B are circulant matrices and $B = A^*$, so $H_1 = A+B = A+A^*$ will be pure real circulant matrix. However $H_2 = A - B = A - A^*$ will be pure imaginary circulant matrix and has the structure like

$$H_2 = \begin{bmatrix} h_0 & -h_{n-1} & \cdots & -h_1 \\ h_1 & h_0 & \cdots & -h_2 \\ \vdots & & & \\ h_{n-1} & h_{n-2} & \cdots & h_0 \end{bmatrix} . \tag{7}$$

We call the matrix H_2 as a negative-circulant matrix. It can also be changed into a circulant matrix again. By section 3.2 we know that the multiplication of a circulant matrix with a vector is cyclic convolution. We will describe the efficient algorithms of cyclic convolution in section 3.6.

Denote the 3-point Fourier transform matrix

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & \mu & \mu^2 \\ 1 & \mu^2 & \mu \end{bmatrix} \tag{8}$$

by $F(3)$, where $\mu = e^{\frac{-2\pi i}{3}}$.

Corollary 2 *For $2m \times 2m$ matrices A,B and C, let $\mu = e^{\frac{-2\pi i}{3}}$ denote a primitive cube root of 1 and n=2m*

$$W(P) = \begin{bmatrix} A & C & B \\ B & A & C \\ C & B & A \end{bmatrix} \tag{9}$$

then

$$W(P) = \frac{1}{3}(F(3) \otimes I_n) \begin{bmatrix} A+B+C & & \\ & A+\mu B+\mu^2 C & \\ & & A+\mu^2 B+\mu C \end{bmatrix} (F^*(3) \otimes I_n)$$

$$= \frac{1}{3}(F(3) \otimes I_n) \begin{bmatrix} H_1 & & \\ & H_2 & \\ & & H_3 \end{bmatrix} (F^*(3) \otimes I_n). \tag{10}$$

where $F(3)$ is the 1-dimensional 3-point Fourier transform matrix and I_n is the $n \times n$ identity matrix.

Observations

According to the corollary 4 of section 3.4, the matrices A, B and C are circulant matrices. Since the matrices H_1, H_2 and H_3 are linear combinations of circulant matrices A, B and C, Then the matrices

$$H_1 = A + B + C \,, \tag{11}$$

$$H_2 = A + \mu B + \mu^2 C \,, \tag{12}$$

$$H_3 = A + \mu^2 B + \mu C \,, \tag{13}$$

are circulant matrices.

Conjugate-circulancy of H_1, H_2 and H_3 also comes from the fact that they are linear combinations of conjugate-circulancy of matrices A, B and C. Thus we may repeat our procedure with H_1, H_2 and H_3 to further decompose the matrices. For example

$$H_1 = A + B + C = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \tag{14}$$

$$= \begin{bmatrix} C_1 + C_2 + C_3 & C_1^* + C_2^* + C_3^* \\ C_1^* + C_2^* + C_3^* & C_1 + C_2 + C_3 \end{bmatrix}$$

$$= \frac{1}{2}(F(2) \otimes I_m) \begin{bmatrix} H_{11} + H_{11}^* & \\ & H_{11} - H_{11}^* \end{bmatrix} (F(2) \otimes I_m) \,,$$

where $H_{11} + H_{11}^*$ is real circulant matrix and $H_{11} - H_{11}^*$ is negative-circulant matrix of pure imaginary which can be changed to circulant matrix again. Similarly H_2 and H_3 also can be diagonalized further like H_1.

Here, we have introduced the method of computation for circulant structure, namely, block-diagonalization. The method of block-diagonalization derived from the multiplicative structure has the following advantages.

1. Using the group theoretic structures, one arrives at circulant blocks of varing sizes. This flexibility is valuable in matching the vector register sizes in vector facility.

2. Since the circulant blocks are located diagonally, they are independent of each other. This independency lends easily to parallel processing with multi-processors. (The number and size of circulant blocks are controlled again by the multiplicative group structure.)

The computation of the twiddle factor $(F(2) \otimes I_n)$ or $(F(3) \otimes I_n)$ is the same as described in section 3.7. In next we will describe the methods for computing the cyclic convolution.

3.11 Computation of Circulant Matrix

From section 3.2 we knew already that the computation of a $n \times n$ circulant matrix with a vector of order n is the matrix form of cyclic convolution. There are many algorithms of cyclic convolution. The best-known method for calculating a cyclic convolution is to use the convolution theorem and a fast Fourier transform algorithm. Take C as $n \times n$ circulant matrix and $\underline{C}$ as a vector of the first column of C, then

$$C = F(n)^{-1} D F(n) , \tag{1}$$

where D is a diagonal matrix by

$$D = diag\,(F(n)\underline{C}) . \tag{2}$$

It means that the cyclic convolution can be computed by the efficient n-point Fourier transform.

When the blocklength is small, the best convolution algorithm as measured by the number of multiplications and additions are the Winograd small convolution algorithms [10] which is efficient for sequential machine. For cyclic convolution of large blocklength we can use a procedure that is known as the Agarwal-Cooley convolution algorithm [13]. The Agarwal-Cooley convolution algorithm construct a fast algorithm for a one-dimensional cyclic convolution by temporarily mapping it into a multidimensional cyclic convolution. It gives a way to build algorithm for large cyclic convolution by combining the small efficient convolution algorithm like the Winograd small convolution algorithm. By using Chinese remainder theorem to order the indexing of data, this indexing changes a one-dimensional cyclic convolution into a multidimensional cyclic convolution. Then we can compute a multidimensional cyclic convolution by nesting a fast algorithm for one-dimensional cyclic convolution inside another fast algorithm for one-dimensional cyclic convolution. Agarwal and Burrus [22] showed how a mapping of the indices of a one-dimensional convolution into multidimension can reduce computation.

Besides the implementation on sequential machine, Agarwal and Cooley [13,23] also shown how the rectangular transform convolution algorithm can be vectorized. It was pointed out that algorithm formulation and implementation not only achieves full vector utilization but successfully copes with the problems of hierarchical storage.

Based on the Agarwal-Cooley convolution algorithm, we also can use the block-diagonalized method to block-diagonalize the circulant matrix further according to the theorem 1 of section 3.10.

Now we will present the block-diagonalized method which block-diagonalize the original circulant matrix C via the matrix tensor product formulas simi-

larly as section 3.6.

Let C is $n \times n$ matrix and $n = 2^{\alpha}q$. By the block-diagonalized method of theorem 1 of section 3.10, we can decompose the matrix C by α steps. Each step will produce 2×2 block-diagonal matrix, one is circulant matrix again and another is negative-circulant matrix.

The negative-circulant matrix can be changed to the circulant matrix by the following ways:

(i) If G is a $n \times n$ negative-circulant matrix,

$$G = \begin{bmatrix} g_0 & -g_{n-1} & \cdots & -g_1 \\ g_1 & g_0 & \cdots & -g_2 \\ \vdots & & & \\ g_{n-1} & g_{n-2} & \cdots & g_0 \end{bmatrix}, \tag{3}$$

then it can be diagonized as follows,

$$G = \begin{bmatrix} \beta^{n-1} & & & \\ & \beta^{n-2} & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \begin{bmatrix} -\beta g_0 & g_{n-1} & \cdots & -\beta^2 g_1 \\ -\beta^2 g_1 & -\beta g_0 & \cdots & -\beta^3 g_2 \\ \vdots & & & \\ g_{n-1} & -\beta^{n-1} g_{n-2} & \cdots & -\beta g_0 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & \beta & & \\ & & \ddots & \\ & & & \beta^{n-1} \end{bmatrix} \tag{4}$$

where $\beta^n = -1$. After diagonalized, the middle matrix is a circulant matrix again.

When n is odd number, it will be very easy since $\beta = -1$ now. Thus the formula (18) has been changed to

$$G = \begin{bmatrix} 1 & & & \\ & -1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \begin{bmatrix} g_0 & g_{n-1} & \cdots & -g_1 \\ -g_1 & g_0 & \cdots & g_2 \\ \vdots & & & \\ g_{n-1} & -g_{n-2} & \cdots & g_0 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & -1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}.$$

(ii) When $n = pq = 2^\alpha$, G can be rewrited as a block negative-circulant matrix,

$$(5) \qquad G = \begin{bmatrix} D_0 & -D_{q-1} & \cdots & -D_1 \\ D_1 & D_0 & \cdots & -D_2 \\ \vdots & & & \\ D_{q-1} & D_{q-2} & \cdots & D_0 \end{bmatrix},$$

where D_i is $P \times P$ matrix. Then G can be changed to block circulant matrix again,

$$(6) G = \begin{bmatrix} I_p & & & \\ & -I_p & & \\ & & \ddots & \\ & & & I_p \end{bmatrix} \begin{bmatrix} D_0 & D_{q-1} & \cdots & -D_1 \\ -D_1 & D_0 & \cdots & D_2 \\ \vdots & & & \\ D_{q-1} & -D_{q-2} & \cdots & D_0 \end{bmatrix} \begin{bmatrix} I_p & & & \\ & -I_p & & \\ & & \ddots & \\ & & & I_p \end{bmatrix}$$

(iii) The matrix G can be changed to the difference of one circulant matrix G' and another special matrix, named the triangle matrix T,

$$(7) \qquad G = G' - T = \begin{bmatrix} g_0 & g_{n-1} & \cdots & g_1 \\ g_1 & g_0 & \cdots & g_2 \\ \vdots & & & \\ g_{n-1} & g_{n-2} & \cdots & g_0 \end{bmatrix} - 2 \begin{bmatrix} 0 & g_{n-1} & \cdots & g_1 \\ 0 & 0 & \cdots & g_2 \\ \vdots & & & \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

3.12 An Example

We will go through our algorithms using computation of the 2-dimensional 5-point Fourier transform as an example. The indexing set $Z/5 \times Z/5$ will be identified with the field $Z/5[\rho]$. $\gamma = 1 + 4\rho$ is a generator of the multiplicative cyclic group of nonzero elements of $Z/5[\rho]$.

$$
\begin{aligned}
&\gamma = 1+4\rho && \gamma^{13} = 4+\rho \\
&\gamma^2 = 2\rho && \gamma^{14} = 3\rho \\
&\gamma^3 = 2+4\rho && \gamma^{15} = 3+\rho \\
&\gamma^4 = 1+\rho && \gamma^{16} = 4+4\rho \\
&\gamma^5 = 2+\rho && \gamma^{17} = 3+4\rho \\
&\gamma^6 = 3 && \gamma^{18} = 2 \\
&\gamma^7 = 3+2\rho && \gamma^{19} = 2+3\rho \\
&\gamma^8 = \rho && \gamma^{20} = 4\rho \\
&\gamma^9 = 1+2\rho && \gamma^{21} = 4+3\rho \\
&\gamma^{10} = 3+3\rho && \gamma^{22} = 2+2\rho \\
&\gamma^{11} = 1+3\rho && \gamma^{23} = 4+2\rho \\
&\gamma^{12} = 4 && \gamma^{24} = \gamma^0 = 1
\end{aligned} \tag{1}
$$

The input and output data will be ordered by the above.

Now the two-dimensional finite Fourier transform of $\underline{X}$ is

$$
\underline{Y} = W(5)\,\underline{X} + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}, \tag{2}
$$

W(5) with respect to (20) is

$$
W(5) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \le j,k < 24} \tag{3}
$$

where $e^{-2\pi i/5}$.

Variant 1

We choose the ordering of (22) as

$$
j = 0,1,2,3,4,5,6,7,8,9,10,11;12,13,14,15,16,17,18,19,20,21,22,23;
$$

$$k = 23,22,21,20,19,18,17,16,15,14,13,12;11,10,9,8,7,6,5,4,3,2,1,0;$$

Then we have

$$W(5) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix},$$

where

$$A = \begin{bmatrix}
\omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega & \omega^3 \\
\omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 & \omega \\
\omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 & 1 \\
1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 & \omega^3 \\
\omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 & \omega^3 \\
\omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega & \omega^2 \\
\omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 & \omega \\
\omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 & \omega^2 \\
\omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega & 1 \\
1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega & \omega \\
\omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4 & \omega \\
\omega^4 & \omega^4 & 1 & \omega^3 & \omega^4 & \omega^3 & \omega^2 & \omega^2 & 1 & \omega^4 & \omega^2 & \omega^4
\end{bmatrix},$$

Applying theorem 1 to $W(5)$ we have

$$W(5) = \frac{1}{2}(F(2) \otimes I_8) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_8),$$

where $H_1 = A + A^*$ is the circulant matrix of real number. And $H_2 = A - A^*$ is the negative circulant matrix which can change to circulant again.

Variant 2

Now we choose the ordering of (22) as follows:

$$j = 0,3,6,9,12,15,18,21;\ 8,11,14,17,20,23,2,5;\ 16,19,22,1,4,7,10,13;$$

$$k = 13,10,7,4,1,22,19,16;\ 5,2,23,20,17,14,11,8;\ 21,18,15,12,9,6,3,0;$$

$$(4) \qquad W(5) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_3 & C_3^* & C_2 & C_2^* \\ C_1^* & C_1 & C_3^* & C_3 & C_2^* & C_2 \\ \hline C_2 & C_2^* & C_1 & C_1^* & C_3 & C_3^* \\ C_2^* & C_2 & C_1^* & C_1 & C_3^* & C_3 \\ \hline C_3 & C_3^* & C_2 & C_2^* & C_1 & C_1^* \\ C_3^* & C_3 & C_2^* & C_2 & C_1^* & C_1 \end{array}\right],$$

$W(5)$ is block circulant matrix and C_1, C_2 and C_3 are conjugate circulant matrices,

$$(5) C_1 = \begin{bmatrix} \omega^4 & \omega^4 & \omega^2 & \omega^2 \\ \omega^3 & \omega^4 & \omega^4 & \omega^2 \\ \omega^3 & \omega^3 & \omega^4 & \omega^4 \\ \omega & \omega^3 & \omega^3 & \omega^4 \end{bmatrix}, C_2 = \begin{bmatrix} \omega^4 & \omega & \omega^2 & \omega^3 \\ \omega^2 & \omega^4 & \omega & \omega 2 \\ \omega^3 & \omega^2 & \omega^4 & \omega \\ \omega^4 & \omega^3 & \omega^2 & \omega^4 \end{bmatrix}, C_3 = \begin{bmatrix} \omega^2 & 1 & \omega & 1 \\ 1 & \omega^2 & 1 & \omega \\ \omega^4 & 1 & \omega^2 & 1 \\ 1 & \omega^4 & 1 & \omega^2 \end{bmatrix}.$$

and

$$(6) \qquad W(5) = \frac{1}{3}[F^*(3) \otimes I_8] \begin{bmatrix} H_1 & & \\ & H_2 & \\ & & H_3 \end{bmatrix} [F(3) \otimes I_8],$$

where H_1, H_2 and H_3 are circulant matrices,

$$H_1 = A + B + C = \begin{bmatrix} C_1 + C_2 + C_3 & C_1^* + C_2^* + C_3^* \\ C_1^* + C_2^* + C_3^* & C_1 + C_2 + C_3 \end{bmatrix},$$

$$H_2 = A + \mu B + \mu^2 C = \begin{bmatrix} C_1 + \mu C_2 + \mu_2 C_3 & C_1^* + \mu C_2^* + \mu^2 C_3^* \\ C_1^* + \mu C_2^* + \mu^2 C_3^* & C_1 + \mu C_2 + \mu^2 C_3 \end{bmatrix},$$

and

$$H_3 = A + \mu^2 B + \mu C = \begin{bmatrix} C_1 + \mu^2 C_2 + \mu C_3 & C_1^* + \mu^2 C_2^* + \mu C_3^* \\ C_1^* + \mu^2 C_2^* + \mu C_3^* & C_1 + \mu^2 C_2 + \mu C_3 \end{bmatrix}.$$

Since H_1, H_2 and H_3 are conjugate-circulancy, it can be block-diagonalized further to get more efficient results.

3.13 Summary

For a prime point $P \equiv 2(3)$, we have developed a multiplicative two-dimensional $P \times P$ Fourier transform algorithm in this chapter. The algorithm is based on the field structure determined by the irreducible polynomial $x^2 + x + 1$ over $\mathcal{Z}/P$. By the multiplicative property of the indexing set $\mathcal{Z}/P \times \mathcal{Z}/P$, the multiplicative structure is used to control the data flow in the Fourier transform computation. Then the calculation of the Fourier transform is expressed as a set of skew-circulant matrices or circulant matrices located diagonally which are more efficient for different machine architectures. For the circulant structure, it means that the computation of the Fourier transform has been changed to a set of cyclic convolutions which also located diagonally. This structure is more powerful to get efficient results for some machine architectures, especially there exists library of more efficient cyclic convolution subroutines in SIMD and MIMD machines. The theory described in this chapter can be used to develop a family of multiplicative multidimensional Fourier transform algorithms with tensor product techniques. In next chapter we will present the another case for $P \equiv 3(4)$.

Chapter 4

Multiplicative 2-Dimensional Prime Point ($P \equiv 3(4)$) FFT Algorithm

4.1 Introduction

In chapter 3 we have developed the new multiplicative 2-dimensional prime point FFT algorithm, but the prime point should satisfy $P \equiv 2(3)$. For wide applications it is necessary for this multiplicative algorithm to cover more prime points. We will extend the prime point $P \equiv 2(3)$ to $P \equiv 3(4)$. Another reason to discuss the case of $P \equiv 3(4)$ is that it can be applyed to the data with $p4$ symmetry of crystallography. We will describe this case in chapter 8.

In this chapter we will discuss the field of P^2 elements for $P \equiv 3(4)$. It is similar to the case of the field of P^2 elements for $P \equiv 2(3)$. In this case x^2+1 is irreducible over $\mathcal{Z}/P$ for $P \equiv 3(4)$ [19]. Set $\rho = exp(-2\pi i/4)$. Then $\rho^2+1 = 0$ and $\mathcal{Z}/P[\rho]$ is a field with P^2 elements; i.e., the nonzero elements of $\mathcal{Z}/P[\rho]$ form a cyclic group under multiplication. Let γ be a cyclic generator. Then

(1) $$1 = \gamma^0, \gamma, \gamma^2, \cdots, \gamma^{P^2-2}$$

are the distinct nonzero elements of $\mathcal{Z}/P[\rho]$.

A generator γ of $U(P)$ can be found satisfying

(2) $$\rho \; = \; \gamma^n \, , n \; = \; (P^2 - 1)/4 \, .$$

where $U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$.

An element $a \in \mathcal{Z}/P[\rho]$ can be written uniquely as $a_1 + \rho a_2$, $a_1, a_2 \in \mathcal{Z}/P$. Arithmetic in $\mathcal{Z}/P[\rho]$ is defined by

(3) $$a + b \; = \; (a_1 + \rho a_2) + (b_1 + \rho b_2) \; = \; (a_1 + b_1) + \rho(a_2 + b_2) \, .$$

$$ab \; = \; (a_1 + \rho a_2)(b_1 + \rho b_2) \; = \; a_1 b_1 - a_2 b_2 + \rho(a_1 b_2 + a_2 b_1) \, .$$

Define a mapping ϕ: $\mathcal{Z}/P[\rho] \rightarrow \mathcal{Z}/P$ by

(4) $$\phi(a_1 + \rho a_2) \; = \; 2a_1 \, .$$

$$\phi(ab) \; = \; 2(a_1 b_1 - a_2 b_2)$$

Lemma 1

(5) $$\phi(ab) \; = \; \phi(ba) \, .$$

This follows from the commutativity in $\mathcal{Z}/P[\rho]$.

Lemma 2

(6) $$\gamma^{P^2-1} \; = \; 1$$

This is the same as lemma 2 of section 3.3.

For a function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$, the Fourier transform of X is

(7)
$$FX(2b_1, -2b_2) = \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P)} X(a_1, a_2) e^{\frac{-2\pi i}{P}(2a_1b_1 - 2a_2b_2)}$$
$$= \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P)} X(a_1, a_2) e^{\frac{-2\pi i}{P}\phi(ab)}$$

The function X and its Fourier Transform FX can be viewed as column vectors once $\mathcal{Z}/P \times \mathcal{Z}/P$ is ordered. To this end, note that $(a_1, a_2) \in \mathcal{Z}/P \times \mathcal{Z}/P$ corresponds uniquely to $a_1 + \rho a_2 \in \mathcal{Z}/P[\rho]$. Hence an ordering of $\mathcal{Z}/P[\rho]$ will yield an ordering for $\mathcal{Z}/P \times \mathcal{Z}/P$.

Take the ordering by $U(P)$. Now we may view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent (5) in terms of matrices as

(8)
$$\begin{bmatrix} Y(0) \\ \underline{Y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix} \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix},$$

where

(9)
$$W(P) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0\le j,k<P^2-1}$$

and the element $0 \in \mathcal{Z}/P[\rho]$ has been placed in front of $U(P)$. For the rest of our discussion, we ignore $0 \in \mathcal{Z}/P[\rho]$ (We can place this at last stage of computation).

4.2 New Algorithm with Skew-Circulant Structure

We will now to derive $W(P)$ depending on different ordering on $\mathcal{Z}/P \times \mathcal{Z}/P$.

Lemma 1 *Take P is prime number. For $P \equiv 3(4)$, P^2-1 is divisible by 8.*

Prove

For $P \equiv 3(4)$, i.e. $P = 4k+3$, $k \in \mathcal{Z}$, we can find $P^2-1 = (4k+3)^2-1 = 8(2k^2+3k+1)$. It means that P^2-1 is divisible by 8.

Set $m = \frac{P^2-1}{4}$ and $n = 2m$. Also set $\omega = exp(-\frac{2\pi i}{P})$. We will need the following properties of generator γ and the mapping function ϕ, all of which are easy to show.

$$\gamma^n = \gamma^{\frac{P^2-1}{2}} = -1\,, \tag{1}$$

$$\phi(\gamma^{j+n}) = -\phi(\gamma^j)\,, \tag{2}$$

$$\omega^{\phi(\gamma^{j+n})} = (\omega^{\phi(\gamma^j)})^*\,. \tag{3}$$

where $*$ denotes the complex conjugation.

By lemma 1, for $P \equiv 3(4)$, P^2-1 is divisible by 8. Take $n = \frac{P^2-1}{2}$, and define the set

$$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{n-1}\}\,. \tag{4}$$

Thus we have that $1, \gamma^n$ is the quotient of $U(P)/S$ and

$$U(P) = S \cup \gamma^n S \tag{5}$$

View S as ordered by (4). This ordering can be used to order $\gamma^n S$. Thus the ordering of $U(P)$ is obtained by putting $S; \gamma^n S$. We will denote by $R(P)$ the group $U(P)$ with this ordering. Now examine the matrix $W(P)$. set $\omega = e^{\frac{-2\pi i}{P}}$.

Theorem 1 *$W(P)$ is of the form*

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix}, \tag{6}$$

where A and B are $n \times n$ conjugate skew-circulant matrices with $A = B^*$. The conjugate of B is denoted by B^*.

Proof

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 \\ W_3 & W_4 \end{bmatrix},$$

where W_i, $1 \leq i \leq 4$ is an $n \times n$ matrix. For $0 \leq k, l < n$, denote the k-th row l-th column entry of an $n \times n$ matrix W_i by $W_i(k, l)$. Then using the properties of generator γ and the mapping function ϕ shown in (1)-(3),

$$W_4(k,l) = \omega^{\phi(\gamma^n \gamma^k \gamma^n \gamma^l)} = \omega^{\phi(\gamma^{2n} \gamma^{k+l})} = \omega^{\phi(\gamma^{k+l})} = W_1(k,l)\,.$$

$$W_3(k,l) = \omega^{\phi(\gamma^n \gamma^k \gamma^l)} = \omega^{\phi(\gamma^n \gamma^{k+l})}\,, = \omega^{\phi(\gamma^k \gamma^n \gamma^l)} = W_2(k,l)\,.$$

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks A and B.

By the properties of γ and ϕ showed in (1) to (3), we know that $\gamma^{(n+j)} = -\gamma^j$. So matrix A equals to the conjugate of B, that is $A = B^*$. $W(P)$ will be 2×2 block circulant matrix with $A = B^*$

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix},$$

where $*$ we mean complex conjugation.

Now we will find out each submatrix A and A^* is conjugate skew-circulant matrix in matrix $W(P)$.

Denote the k-th row l-th column entry of the $n \times n$ matrix A by $A(k,l)$. Then using the properties (1)-(3),

(1) For $0 \leq k, l < n-1$,

$$A(k,l+1) = \omega^{\phi(\gamma^{k+l+1})} = \omega^{\phi(\gamma^{(k+1)+l})} = A(k+1,l) \ ,$$

(2) For $1 \leq k < n$,

$$A(k,n-1) = \omega^{\phi(\gamma^{k+n-1})} = \omega^{\phi(\gamma^{k-1}\gamma n)} = \omega^{\phi(-\gamma^{k-1})} = A(k-1,0)^* \ ,$$

hence A is conjugate skew-circulant matrix. Since A^* is the conjugate of A, A^* is also a conjugate skew-circulant matrix.

By the corollary 2 of section 3.4, we also know that $W(P)$ is a skew-circulant matrix.

By the theorem 1 of section 3.5, $W(P)$ can be diagonalized further as

$$(7) \qquad W(P) = \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} A+A^* & \\ & A-A^* \end{bmatrix} (F(2) \otimes I_n)$$

$$= \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_n) \ .$$

where $F(2)$ is the one-dimensional 2-point Fourier transform matrix and I_n is identity matrix.

$H_1 = A + A^*$ is skew-circulant matrix of pure real. However $H_2 = A - A^*$ will be pure imaginary and a negative-skew-circulant matrix which is the same structure as (7) of section 3.5. It can be changed into a skew-circulant matrix again.

The method described here is most the same as variant 1 of section 3.4 both of which are skew-circulant structure, except for each method is suitable for different size of prime point.

4.3 New Algorithm with Circulant Structure

We assume already that P is a prime number and $P \equiv 3(4)$. By Set $n = \frac{P^2-1}{2}$, and define the set

$$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{n-1}\} . \tag{1}$$

Thus we have that $1, \gamma^n$ is the quotient of $U(P)/S$ and

$$U(P) = S \cup \gamma^n S . \tag{2}$$

We can rewrite the set S as

$$S' = \{\gamma^{n-1}, \gamma^{n-2}, \cdots, \gamma, \gamma^0\} . \tag{3}$$

Then $U(P)$ can be rewritten as

$$U(P) = \gamma^n S' \cup S' . \tag{4}$$

View S and S' as ordered by (1) and (3). These orderings can also be used to order $\gamma^n S$ and $\gamma^n S'$. We will denote by $R_1(P)$ with $S; \gamma^n S$ and $R_2(P)$ with $\gamma^n S'; S'$ instead of the original $U(P)$. We order the column indexing by $R_1(P)$ and order the row indexing by $R_2(P)$. Now we examine the matrix $W(P)$.

Theorem 1 *$W(P)$ is of the form*

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} , \tag{5}$$

where A and B are $n \times n$ conjugate circulant matrices with $A = B^*$. The conjugate of B is denoted by B^*.

Proof

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 \\ W_3 & W_4 \end{bmatrix},$$

where W_i, $1 \leq i \leq 4$ is an $n \times n$ matrix. For $0 \leq k, l < n$, denote the k-th row l-th column entry of an $n \times n$ matrix W_i by $W_i(k,l)$. Then using the properties of generator γ and the mapping function ϕ,

$$W_4(k,l) = \omega^{\phi(\gamma^{2n-1-k}\gamma^n\gamma^l)} = \omega^{\phi(\gamma^n\gamma^{2n-1-k}\gamma^l)} = W_1(k,l)\,.$$

$$W_3(k,l) = \omega^{\phi(\gamma^{2n-1-k}\gamma^l)} = \omega^{\phi(\gamma^n\gamma^{2n-1-k}\gamma^n\gamma^l)} = W_2(k,l)\,.$$

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks A and B. By the properties of γ and ϕ, we know that $\gamma^{(n+j)} = -\gamma^j$. Thus matrix A equals to the conjugate of B, that is $A = B^*$. $W(P)$ will be 2×2 block circulant matrix with $A = B^*$, that is

$$W(P) = \begin{bmatrix} A & A^* \\ A^* & A \end{bmatrix},$$

where $*$ we mean complex conjugation.

Now we look each submatrix A and A^*. Denote the k-th row l-th column entry of the $n \times n$ matrix A by $A(k,l)$,

(1) For $0 \leq k, l < n-1$,

$$A(k+1,l+1) = \omega^{\phi(\gamma^n\gamma^{n-1-(k+1)}\gamma^{l+1})} = \omega^{\phi(\gamma^n\gamma^{n-1-k}\gamma^l)} = A(k,l)\,,$$

(2) For $0 \leq l < n-1$,

$$A(n-1,l) = \omega^{\phi(\gamma^{2n-(n-1)+l})} = \omega^{\phi(\gamma^{l+1}\gamma^n)} = \omega^{-\phi(\gamma^{l+1})} = A(0,l+1)^*\,,$$

hence A is conjugate circulant matrix. Since A^* is the conjugate of A, A^* is also a conjugate circulant matrix.

By the corollary 1 of section 3.9, $W(P)$ is also a circulant matrix.

By the theorem 1 of section 3.10, $W(P)$ can be diagonalized further as

$$(6) \qquad W(P) = \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} A + A^* & \\ & A - A^* \end{bmatrix} (F(2) \otimes I_n)$$

$$= \frac{1}{2}(F(2) \otimes I_n) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_n) ,$$

where $F(2)$ is the one-dimensional 2-point Fourier transform matrix and I_n identity matrix.

$H_1 = A + A^*$ is a circulant matrix of pure real. However $H_2 = A - A^*$ will be pure imaginary and be a negative-circulant matrix which is the same structure as (7) of section 3.10. It can be changed into a circulant matrix again.

The method described here be similar the same as variant 1 of section 3.9 both of which are circulant structure, except for each method is suitable for different size.

The main advantage of this method described in section 4.2 and 4.3 is the extension of the original size $P \equiv 2(3)$ to the size of $P \equiv 3(4)$. Thus more prime points can be covered with this efficient algorithm.

4.4 An Example

We will go through this algorithm by using computation of the 2-dimensional 3-point Fourier transform as an example. The indexing set $\mathcal{Z}/3 \times \mathcal{Z}/3$ will be

identified with the field $\mathcal{Z}/3[\rho]$. $\gamma = 1+2\rho$ is a generator of the multiplicative cyclic group of nonzero elements of $\mathcal{Z}/3[\rho]$.

$$
\begin{aligned}
&\gamma = 1 + 2\rho \\
&\gamma^2 = \rho \\
&\gamma^3 = 1 + \rho \\
&\gamma^4 = 2 \\
&\gamma^5 = 2 + \rho \\
&\gamma^6 = 2\rho \\
&\gamma^7 = 2 + 2\rho \\
&\gamma^8 = \gamma^0 = 1
\end{aligned} \tag{1}
$$

Now the 2-dimensional finite Fourier transform of $\underline{X}$ is

$$
\underline{Y} \;=\; W(3)\underline{X} \;+\; \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}, \tag{2}
$$

where $W(3)$ with respect to (7) of section 4.1 is

$$
W(3) \;=\; \left[\; \omega^{\phi(\gamma^{j+k})} \;\right]_{0 \le j,k < 8}, \tag{3}
$$

and $\omega = e^{\frac{-2\pi i}{3}}$.

Variant 1

Let the ordering of j and k are

$$j = 0, 1, 2, 3, 4, 5, 6, 7;$$

$$k = 0, 1, 2, 3, 4, 5, 6, 7;$$

Then

$$W(3) = \begin{bmatrix} A & B \\ B & A \end{bmatrix} = \frac{1}{2}(F(2) \otimes I_4) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_4),$$

where

$$A = \begin{bmatrix} \omega^2 & \omega^2 & 1 & \omega^2 \\ \omega^2 & 1 & \omega^2 & \omega \\ 1 & \omega^2 & \omega & \omega \\ \omega^2 & \omega & \omega & 1 \end{bmatrix}$$

$H_1 = A + A^*$ is pure real and skew-circulant matrix. Set $\psi = -\frac{2\pi}{3}$,

$$H_1 = 2 \begin{bmatrix} cos2\psi & cos2\psi & 1 & cos2\psi \\ cos2\psi & 1 & cos2\psi & cos\psi \\ 1 & cos2\psi & cos\psi & cos\psi \\ cos2\psi & cos\psi & cos\psi & 1 \end{bmatrix}.$$

$H_2 = A - A^*$ is pure imaginary and negative skew-circulant matrix which can change to skew-circulant matrix again,

$$H_2 = -2i \begin{bmatrix} sin2\psi & sin2\psi & 1 & sin2\psi \\ sin2\psi & 1 & sin2\psi & -sin\psi \\ 1 & sin2\psi & -sin\psi & -sin\psi \\ sin2\psi & -sin\psi & -sin\psi & -1 \end{bmatrix}.$$

Variant 2

To find circulant structure we order j and k of (9) as follows:

$$j = 0,1,2,3,4,5,6,7;$$

$$k = 7,6,5,4,3,2,1,0;$$

Then

$$W(5) = \frac{1}{2}(F(2) \otimes I_4) \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} (F(2) \otimes I_4)\,, \tag{4}$$

where $H_1 = A + A^*$ and $H_2 = A - A^*$.

$$A = \begin{bmatrix} \omega & \omega^2 & \omega^2 & 1 \\ 1 & \omega & \omega^2 & \omega^2 \\ \omega & 1 & \omega & \omega^2 \\ \omega & \omega & 1 & \omega \end{bmatrix}.$$

H_1 is real circulant matrix and H_2 is pure imaginary negative circulant matrix which can be changed to circulant matrix again.

4.5 Summary

This chapter provides the multiplicative two-dimensional Fourier transform algorithm extended from the size $P \equiv 2(3)$ to the size $P \equiv 3(4)$. The main advantage of this method is that more prime points can use this efficient algorithm. Thus more prime points can be covered with this efficient algorithm. When $P < 100$, the coverage is $\frac{20}{25} = 80\%$. There are also two kinds of skew-circulant and circulant structures which have block-diagonal structure with each block of skew-circulant matrix or circulant matrix located diagonally. Then we can use the same methods described in chapter 3 to get more efficient performance of implementation on different machine architectures.

Chapter 5

Multiplicative 2-Dimensional $N = P_1P_2$ FFT Algorithm

5.1 Introduction

We have developed the new multiplicative $P \times P$ two-dimensional Fourier transform algorithms when P is a prime number $P \equiv 2(3)$ and $P \equiv 3(4)$ in chapter 3 and chapter 4. In this chapter we will use the Chinese Remainder Theorem(CRT) to develop the two-dimensional $N = P_1P_2$ FFT algorithm which nests the multiplicative two-dimensional prime point algorithms in the Good-Thomas prime factor algorithm, where P_1 and P_2 are relative prime. That means the large number $N \times N$ two-dimensional FFT will be replaced by small number $P_1 \times P_1$ and $P_2 \times P_2$ two-dimensional FFTs which we has developed already from chapter 3 to chapter 5. Since the Chinese remainder theorem plays a major role in generalizing the algorithm which is going to be described, we will first introduce CRT theorem.

5.2 Chinese Remainder Theorem(CRT)

Let $N = P_1P_2$. Two positive integers P_1 and P_2 are called relatively prime if the greatest common divisor(GCD) of P_1 and P_2 is 1, denoted by $GCD(P_1, P_2) = 1$. We can form the ring direct product by

$$\mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \,. \tag{1}$$

An element in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$ can be written uniquely as an ordered pair

$$(u_1, u_2)\,, \quad u_1 \in \mathcal{Z}/P_1,\ \ u_2 \in \mathcal{Z}/P_2\,. \tag{2}$$

Arithmetic operations in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$ are defined by componentwise addtion

$$(u_1, u_2) + (v_1, v_2) \ = \ ((u_1 + v_1) \ \ mod\, P_1,\ \ (u_2 + v_2) \ \ mod\, P_2)\,,$$

and componentwise multiplication

$$(u_1, u_2)(v_1, v_2) \ = \ ((u_1v_1) \ \ mod\, P_1\,, (u_2v_2) \ \ mod\, P_2)\,.$$

Theorem 1 *Chinese remainder theorem(CRT)*

Let $N = P_1P_2$, with $(P_1, P_2) = 1$, then there exists a ring-isomorphism

$$\mathcal{Z}/N \cong \mathcal{Z}/P_1 \times \mathcal{Z}/P_2\,. \tag{4}$$

We will construct the ring-isomorphism using idempotents. Since $(P_1, P_2) = 1$, there exists integer C_1 and C_2 satisfy

$$C_1P_1 \ + \ C_2P_2 \ = \ 1 \tag{5}$$

Now we define

$$e_1 \equiv C_2P_2 \quad mod\, N\,, \tag{6}$$

$$e_2 \equiv C_1P_1 \quad mod\, N\,. \tag{7}$$

Rewrite (6) as

(8) $$e_1 = C_2P_2 + NM\,, \quad M \in \mathcal{Z}\,.$$

We can find out from (5) and (8) that

(9) $$e_1 \equiv 1 \ mod\, P_1\,, \quad e_1 \equiv 0 \ mod\, P_2\,.$$

Similarly we can find out continuely that

(10) $$e_2 \equiv 1 \ mod\, P_2\,, \quad e_2 \equiv 0 \ mod\, P_1\,.$$

The idempotents e_1 and e_2 are uniquely determined by conditions (9) and (10). The set

(11) $$\{e_1, e_2\}$$

is called the system of idempotents corresponding to the factorization $N = P_1P_2$, where $(P_1, P_2) = 1$. Examples of systems of idempotents are given in table 1.

N	P_1	P_2	e_1	e_2	f_1	f_2
6	2	3	3	4	2	1
10	2	5	5	6	3	1
12	3	4	4	9	3	1
15	3	5	10	6	2	2
21	3	7	7	15	5	1
28	4	7	21	8	2	3
30	2	15	15	16	8	1

Table 1 Examples of Idempotents

Besides (9) and (10), e_1 and e_2 have the following more properties:

(12) $$e_1 e_2 \equiv 0 \ mod\, N\ .$$

(13) $$e_1 + e_2 \equiv 1 \ mod\, N\ .$$

(14) $$e_1^2 \equiv e_1 \ mod\, N\ , \quad e_2^2 \equiv e_2 \ mod\, N\ .$$

These properties above uniquely determine e_1 and e_2 in $\mathcal{Z}/N$. Hence we can find f_1 and f_2 in $\mathcal{Z}/N$ with

(15) where $(f_2, P_1) = 1$ and $(f_1, P_2) = 1$.

where $(f_2, P_1) = 1$ and $(f_1, P_2) = 1$.

We now show that the existence of system of idempotents gives rise to a way of identifying $\mathcal{Z}/N$ and $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$. Define the following mapping

(16) $$\phi : \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \to \mathcal{Z}/N\ ,$$

by the formula

(17) $$\phi(u_1, u_2) \ = \ (u_1 e_1 \ + \ u_2 e_2) \ mod\, N\ ,$$

where $u_1 \in \mathcal{Z}/P_1$, $u_2 \in \mathcal{Z}/P_2$, the set $\{e_1, e_2\}$ is the system of idempotents corresponding to the factorization $N = P_1 P_2$.

Observation

The mapping ϕ is a ring-isomorphism.

For $u, v \in \mathcal{Z}/N$, by (17) we can see that

$$\phi(u+v) = \phi(u_1 + v_1, u_2 + v_2) = (u_1 + v_1)e_1 + (u_2 + v_2)e_2$$

$$= \ (u_1 e_1 + u_2 e_2) + (v_1 e_1 + v_2 e_2) \ = \ \phi(u) + \phi(v) \ mod\, N\ .$$

also by the properties of $\{e_1, e_2\}$ shown in (12) and (14) we can find out that

$$\phi(u)\phi(v) = (u_1e_1 + u_2e_2)(v_1e_1 + v_2e_2) = u_1v_1e_1 + u_2v_2e_2$$

$$= \phi(uv)$$

Hence the mapping ϕ preserves the arithmetic structure of rings and ϕ is a ring homomorphism.

This means that every ordered pair (u_1, u_2), $u_1 \in \mathcal{Z}/P_1, u_2 \in \mathcal{Z}/P_2$ can be written uniquely as

(18) $$u_1e_1 + u_2e_2 \equiv u \mod N .$$

From the above description we see that the inverse mapping ϕ^{-1} is given by the following way,

(19) $$\phi^{-1} : \mathcal{Z}/N \to \mathcal{Z}/P_1 \times \mathcal{Z}/P_2$$

and

(20) $$\phi^{-1}(u) = (u \bmod P_1, u \bmod P_2), \quad u \in \mathcal{Z}/N .$$

which identifies that every element $u \in \mathcal{Z}/N$ can be identifyed uniquely with the ordered pair $(u \bmod P_1, u \bmod P_2)$ in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$.

Example 1

Let $N = 12 = 3 \times 4$. By Table 1 we know that $e_1 = 4$ and $e_2 = 9$. The mapping ϕ is given by the table 2 below.

$\mathcal{Z}/3 \times \mathcal{Z}/4$	$\mathcal{Z}/12$
(0,0)	0
(0,1)	9
(0,2)	6
(0,3)	3
(1,0)	4
(1,1)	1
(1,2)	10
(1,3)	7
(2,0)	8
(2,1)	5
(2,2)	2
(2,3)	11

Table 2

5.3 $N = P_1P_2$ FFT Algorithm

5.3.1 Row-Column Method

For a natural number N, denote by $\mathcal{Z}/N \times \mathcal{Z}/N$, the cartesian product of two copies of the ring $\mathcal{Z}/N$. An element of $\mathcal{Z}/N \times \mathcal{Z}/N$ is denoted by (u_1, u_2), $u_1, u_2 \in \mathcal{Z}/N$. For a function X defined on $\mathcal{Z}/N \times \mathcal{Z}/N$, the two-dimensional Fourier transform of X is defined by

(1)
$$FX(v_1, v_2) = \sum_{(u_1,u_2)\in \mathcal{Z}/N\times\mathcal{Z}/N} X(u_1, u_2)e^{\frac{-2\pi i}{N}(u_1v_1+u_2v_2)}.$$

Formula (1) can be rewritten as

$$(2) \qquad FX(v_1, v_2) = \sum_{u_2 \in Z/N} \Big(\sum_{u_1 \in Z/N} X(u_1, u_2) e^{\frac{-2\pi i}{N} u_1 v_1} \Big) e^{\frac{-2\pi i}{N} u_2 v_2} .$$

Then (2) can be computed in the following successive procedures:

1) Compute $F(N)$ along u_1 for $u_2 \in Z/N$.

$$(3) \qquad FX_1(v_1, u_2) = \sum_{u_1 \in Z/N} X(u_1, u_2) e^{\frac{-2\pi i}{N} u_1 v_1} .$$

2) Compute $F(N)$ along u_2 for $u_1 \in Z/N$.

$$(4) \qquad FX_2(v_1, v_2) = \sum_{u_2 \in Z/N} X(v_1, u_2) e^{\frac{-2\pi i}{N} u_2 v_2} .$$

This Row-Column method decomposes the two-dimensional Fourier transform computation into a series of one-dimensional Fourier transform computations, but each length of one-dimensional Fourier transform is fixed by N.

We also can use tensor product technique to describe the Row-Column method. The function X and its Fourier transform FX can be viewed as column vector once $Z/N \times Z/N$ is ordered in (1). So we may view the function X defined on $Z/N \times Z/N$ and its Fourier transform FX as column vectors, and represent (1) in terms of matrices as

$$(5) \qquad FX = (F(N) \otimes F(N))X$$

where $F(N)$ is one-dimensional Fourier transform matrix.

By the properties of tensor product of lemma 1 in section 2.5.2, (5) can be changed to

$$(6) \qquad FX = (F(N) \otimes I_N)(I_N \otimes F(N))X$$

then by the theorem 7 of section 2.5.3,

$$(7) \qquad FX = P^{-1}(I_N \otimes F(N))P(I_N \otimes F(N))X$$

where P^{-1} and P are permutation matrices

$$P = P^{-1} = P(N^2, N). \tag{8}$$

and $(I_N \otimes F(N))$ means the computation of one-dimensional Fourier transform N times.

5.3.2 Prime Factor Method

From now we will apply Chinese remainder theorem to Fourier transform computation. We will provide a way using Chinese remainder theorem to replace a large size one-dimensional Fourier transform computation with a small size two-dimensional computation.

By the Chinese remainder theorem from setion 5.2, for $N = P_1P_2$ and $(P_1P_2) = 1$, we have defined the following mappings

$$\phi : \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \rightarrow \mathcal{Z}/N\,, \tag{9}$$

by the formula

$$\phi(u_1, u_2) = (u_1e_1 + u_2e_2) \ mod\, N\,, \tag{10}$$

where $u_1 \in \mathcal{Z}/P_1$ and $u_2 \in \mathcal{Z}/P_2$, the $\{e_1, e_2\}$ is the system of idempotents corresponding to the factorization $N = P_1P_2$. And the inverse mapping ϕ^{-1} is given by

$$\phi^{-1} : \mathcal{Z}/N \rightarrow \mathcal{Z}/P_1 \times \mathcal{Z}/P_2\,, \tag{11}$$

that is

$$\phi^{-1}(u) = (u \ mod\, P_1,\ u \ mod\, P_2)\,, \quad u \in \mathcal{Z}/N\,. \tag{12}$$

which identifies that every element $u \in \mathcal{Z}/N$ can be identified uniquely with the ordered pair $(u \ mod\, P_1, u \ mod\, P_2)$ in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$.

Denote $\mathcal{Z}/N \times \mathcal{Z}/N$ by $(\mathcal{Z}/N)^2$. The mapping ϕ and ϕ^{-1} can be extended to

$$\psi : (\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2 \to (\mathcal{Z}/N)^2 , \tag{13}$$

$$\psi((u_1, v_1), (u_2, v_2)) = (\phi(u_1, u_2), \phi(v_1, v_2)) .$$

And

$$\psi^{-1} : (\mathcal{Z}/N)^2 \to (\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2 , \tag{14}$$

$$\psi^{-1}(u, v) = (\phi^{-1}(u), \phi^{-1}(v)) ,$$

Now let $X(u)$ be a function on $\mathcal{Z}/N$. The one-dimensional Fourier transform is defined as

$$FX(v) = \sum_{u \in \mathcal{Z}/N} X(u) e^{\frac{-2\pi i}{N} uv} . \tag{15}$$

For $u, v \in \mathcal{Z}/N$, we can use the system of idempotents $\{e_1, e_2\}$ to change u and v by

$$u = u_1 e_1 + u_2 e_2 \ mod\, N , \tag{16}$$

and

$$v = v_1 e_1 + v_2 e_2 \ mod\, N . \tag{17}$$

Then

$$FX(v_1 e_1 + v_2 e_2) = \sum_{u_1 \in \mathcal{Z}/N, u_2 \in \mathcal{Z}/N} e^{\frac{-2\pi i}{N}(u_1 e_1 + u_2 e_2)(v_1 e_1 + v_2 e_2)} , \tag{18}$$

By the properties of the system of idempotents $\{e_1, e_2\}$,

$$e^{\frac{-2\pi i}{N}(u_1 e_1 + u_2 e_2)(v_1 e_1 + v_2 e_2)} = e^{\frac{-2\pi i}{N} u_1 v_1 e_1} e^{\frac{-2\pi i}{N} u_2 v_2 e_2} .$$

by (15) of section 8.2, $e_1 = f_2 P_2$ and $e_2 = f_1 P_1$, the above equality can be changed to

$$e^{\frac{-2\pi i}{P_1} u_1 v_1 f_2} e^{\frac{-2\pi i}{P_2} u_2 v_2 f_1}$$

(18) can be rewritten as

$$(19) \quad FX(v_1e_1 + v_2e_2) = \sum_{u_1 \in \mathcal{Z}/P_1 u_2 \in \mathcal{Z}/P_2} X(u_1e_1 + u_2e_2) e^{\frac{-2\pi i}{P_1} u_1 v_1 f_2} e^{\frac{-2\pi i}{P_2} u_2 v_2 f_1} .$$

This method is known as Good-Thomas [4,5] Prime Factor algorithm which also can be presented by using the tensor product algebra.

Viewing the function X and its Fourier transform FX as column vector on $\mathcal{Z}/N$ then (15) can be expressed as

$$(20) \quad FX = Q'(F(P_1) \otimes F(P_2))QX .$$

where $F(P_1)$ and $F(P_2)$ are one-dimensional Fourier transform matrices. Q and Q' are permutation matrices.

The Good-Thomas Prime Factor algorithm also can be applied to the computation of multidimensional Fourier transform as well. Now we consider about the two-dimensional Fourier transform in the spirit of the Good-Thomas Prime Factor algorithm. The function X and its Fourier transform are defined on $\mathcal{Z}/N \times \mathcal{Z}/N$ viewing as column vectors. Let $N = P_1P_2$ and $(P_1, P_2) = 1$. Define

$$(21) \quad F(P_2) \otimes F(P_2) = F_2(P_2) ,$$

and

$$(22) \quad F(P_1) \otimes F(P_1) = F_2(P_1) .$$

(21) and (22) are known as two-dimensional Fourier transform matrices. Then the $N \times N$ two-dimensional Fourier transform can be presented as

$$\begin{aligned} FX &= Q_1 \left(F_2(P_2) \otimes F_2(P_1)\right) Q_2X \\ &= Q_1 \left(F_2(P_2) \otimes I_{P_1^2}\right) \left(I_{P_2^2} \otimes F_2(P_1)\right) Q_2X , \\ &= Q_3 \left(I_{P_1^2} \otimes F_2(P_2)\right) Q_4 \left(I_{P_2^2} \otimes F_2(P_1)\right) Q_2X ; \end{aligned} \quad (23)$$

where $N = P_1P_2$, $(P_1, P_2) = 1$ and Q_i, $1 \leq i \leq 4$ are permutation matrices. The permutation matrix Q_2 can be found out by using Chinese remainder theorem to get the mapping from the indexings of $F_2(P_2)$ and $F_2(P_1)$.

(23) means that if $N = P_1P_2$ and $(P_1, P_2) = 1$, the large number $N \times N$ two-dimensional Fourier transform can be computed by a series of small number routines of $P_1^2 \times P_1^2$ and $P_2^2 \times P_2^2$ two-dimensional Fourier transform matrices $F(P_1) \otimes F(P_1)$ and $F(P_2) \otimes F(P_2)$. But these routines $F_2(P_1)$ and $F_2(P_2)$ are not efficient. The improvement is to replace it by other more efficient routines which we present in last three chapters.

We have developed the efficient two-dimensional Fourier transform for $P \equiv 2(3)$ and $P \equiv 3(4)$ already. We view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent the two-dimensional Fourier transform in (1) of section 5.3 in terms of matrices by (9) and (10) of section 3.3 as

$$\begin{bmatrix} Y(0) \\ \underline{Y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix} \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix},$$

$$= M(P) \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix}, \tag{24}$$

where

$$W(P) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \leq j,k < P^2-1},$$

and $M(P)$ is the $P^2 \times P^2$ matrix of the multiplicative two-dimensional prime

point FFT algorithm,

$$M(P) = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix}. \tag{25}$$

Then each term of two-dimensional Fourier transform matrices $F_2(P_2)$ or $F_2(P_1)$ in (23) can be replaced by $M(P_1)$ or $M(P_2)$ in (25). The two-dimensional Fourier transform now is

$$\begin{aligned} FX &= Q_5 \left(M(P_2) \otimes I_{P_1^2}\right) \left(I_{P_2^2} \otimes M(P_1)\right) Q_6 X \\ &= Q_7 \left(I_{P_1^2} \otimes M(P_2)\right) Q_8 \left(I_{P_2^2} \otimes M(P_1)\right) Q_6 X \end{aligned} \tag{26}$$

where Q_5, Q_6, Q_7 and Q_8 are permutation matrices. The permutation matrix Q_6 can be found out by using Chinese remainder theorem to get the mapping from the indexing orderings of $M(P_2)$ and $M(P_1)$.

Comparing with the method in formula (23), the prime factor method with nesting the multiplicative two-dimensional prime point FFT algorithm in (26) provide a more efficient way to solve the large number $N = P_1P_2$ two-dimensional Fourier transform depending on the small efficient two-dimensional prime point FFT algorithm of $M(P_1)$ and $M(P_2)$. By (26) we can see that this structure can be implemented in parallel architecture. Furthermore, the sizes of P_1 and P_2 are more flexible to match different machine architectures.

5.4 Summary

The multiplicative two-dimensional $N = P_1P_2$ Fourier transform algorithm presented in this chapter is the extension of the Good-Thomas prime factor al-

gorithm of one-dimensional FFT. Based on the efficient multiplicative $P_1 \times P_1$ and $P_2 \times P_2$ two-dimensional FFT algorithms developed in preceding chapters, we can use the Chinese Remainder Theorem to build large $N \times N$ two-dimensional FFT algorithm where $N = P_1 P_2$ and $(P_1, P_2) = 1$ to gain computational advantage. Furthermore, by continuing using Chinese Remaninder Theorem, this method can also be extended to the case of $N = P_1 P_2 P_3$ where P_1, P_2 and P_3 are relatively prime in pairs.

Chapter 6

Multiplicative 2-Dimensional Prime Point FFT Algorithm for $p3$ Symmetry

6.1 Introduction

X-ray method for determining the structure factors of a crystal requires massive repetition of Fourier transform computations. X-ray data of a crystal respects the crystallographic symmetry giving rise to data redundancy. This redundancy is controlled by the crystallographic symmetry groups. By incorporating the crystallographic symmetries to efficient Fourier transform algorithms, one can gain computational advantage. As an application of the multiplicative algorithm which we have developed in chapter 3, we will illustrate incorporating of the $p3$ symmetry to this algorithm [24] in this chapter.

6.2 p3 Symmetry

The $p3$ symmetry may be represented as the following group of 2×2 matrices,

$$S_{p3} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & 1 \\ -1 & 0 \end{bmatrix} \right\} \tag{1}$$

Denoting a generator of S_{p3} by α, say

$$\alpha = \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \tag{2}$$

then S_{p3} is cyclic group as

$$S_{p3} = \{I_2 = \alpha^3, \alpha, \alpha^2\} . \tag{3}$$

Then the redundancy in X-ray data is

$$f(x,y) = f(\alpha(x,y)) = f(-y, x-y) , \tag{4}$$

and

$$f(x,y) = f(\alpha^2(x,y)) = f(y-x, -x) \tag{5}$$

where f is the function whose value is the X-ray data at the lattice point (x,y). Thus $p3$ can be viewed as acting on a two-dimensional lattice via the matrix multiplication.

Observe now that the minimal polynomial of α in the variable λ is

$$\lambda^2 + \lambda + 1 . \tag{6}$$

Thus the action of α on $\mathcal{Z}/P \times \mathcal{Z}/P$ corresponds to the multiplication action by ρ on $\mathcal{Z}/P[\rho]$, i.e.,

$$\alpha(x,y) \rightarrow (-y, x-y) \tag{7}$$

$$\rho(x+\rho y) \to \rho x + \rho^2(y) = -y + \rho(x-y)\,.$$

Denote the Fourier transform of f by $\hat{f}$.

Theorem 1 *For a function f defined on $\mathcal{Z}/N \times \mathcal{Z}/N$, if $f(x,y) = f(\alpha(x,y))$, then*

$$\hat{f}(u,v) = \hat{f}(\alpha^{\#}(u,v))\,, \tag{8}$$

where $\alpha^{\#}$ denotes the inverse transpose of the matrix α, $\alpha^{\#} = (\alpha^{-1})^t$.

Prove

We define the bilinear form on $\mathcal{Z}/N \times \mathcal{Z}/N$,

$$< (x,y),(u,v) > = xu + yv\,.$$

Let the inverse of the generator α is

$$\alpha^{-1} = \begin{bmatrix} \alpha_1 & \alpha_2 \\ \alpha_3 & \alpha_4 \end{bmatrix},$$

and the transpose of the inverse of the generator α is

$$(\alpha^{-1})^t = \begin{bmatrix} \alpha_1 & \alpha_3 \\ \alpha_2 & \alpha_4 \end{bmatrix}.$$

Then we can find that

$$< \alpha^{-1}(x,y),(u,v) > = < (\alpha_1 x + \alpha_2 y, \alpha_3 x + \alpha_4 y),(u,v) >$$

$$= x(\alpha_1 u + \alpha_3 v) + y(\alpha_2 u + \alpha_4 v) = < (x,y),(\alpha^{-1})^t(u,v) >$$

The Fourier transform of f as $\hat{f}$ is defined by

$$\hat{f}(u,v) = \sum_{Z/N\times Z/N} f(x,y)e^{\frac{-2\pi i}{N}<(x,y),(u,v)>}$$

we replacing (x,y) by $\alpha^{-1}(x,y)$, then

$$\hat{f}(u,v) = \sum_{Z/N\times Z/N} f(\alpha^{-1}(x,y))e^{\frac{-2\pi i}{N}<\alpha^{-1}(x,y),(u,v)>}$$

$$= \sum_{Z/N\times Z/N} f(x,y)e^{\frac{-2\pi i}{N}<(x,y),(\alpha^{-1})^t(u,v)>} = \hat{f}(\alpha^{\#}(u,v)) .$$

completing the proof of theorem 1.

6.3 New Algorithm for p3 Symmetry

From chap3 we treat $Z/P\times Z/P$ for a prime number $P \equiv 2(3)$, as a simple object. The polynomial x^2+x+1 is irreducible over Z/P for $P \equiv 2(3)$. Set $\rho = exp(-2\pi i/3)$. Then $\rho^2+\rho+1=0$ and $Z/P[\rho]$ is a field with P^2 elements; i.e., the nonzero elements of $Z/P[\rho]$ form a cyclic group under multiplication. Let γ be a cyclic generator. Then

(1) $$1 = \gamma^0 , \gamma , \gamma^2 , \cdots \gamma^{P^2-2}$$

are the distinct nonzero elements of $Z/P[\rho]$

An element $a \in Z/P[\rho]$ can be written uniquely as $a_1+\rho a_2$, $a_1, a_2 \in Z/P$. Arithmetic in $Z/P[\rho]$ is defined by

(2) $$(a_1+\rho a_2)+(b_1+\rho b_2) = (a_1+b_1)+\rho(a_2+b_2) .$$

$$(a_1+\rho a_2)(b_1+\rho b_2) = a_1b_1-a_2b_2+\rho(a_2b_1+a_1b_2-a_2b_2) .$$

Define a mapping $\phi : \mathcal{Z}/P[\rho] \to \mathcal{Z}/P$ by

$$\phi(a_1 + \rho a_2) = a_1 . \tag{3}$$

Observe that

$$\phi(a+b) = \phi(a) + \phi(b) , \tag{4}$$

$$\phi(ab) = a_1 b_1 - a_2 b_2 .$$

Variant 1

We now look variant 2 in section 3.4. For $P \equiv 2(3)$, the element γ^3 is of order n and generates the subgroup

$$S = \{\gamma^0, \gamma^3, \gamma^6, \cdots, \gamma^{3(n-1)}\} \tag{5}$$

where $n = \frac{P^2-1}{3}$.

$U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$, and

$$U(P) = S \cup \gamma^n S \cup \gamma^{2n} S . \tag{6}$$

View S as ordered by (5). This ordering can be used to order $\gamma^n S$ and $\gamma^{2n} S$. Now, the ordering of $U(P)$ is obtained by putting $S; \gamma^n S; \gamma^{2n} S$. We will denote by $R(P)$ the group $U(P)$ with this ordering.

Then we have gotten the block-skew-circulant matrix in (17) of section

3.10 as

$$(7) \qquad W(P) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_2 & C_2^* & C_3 & C_3^* \\ C_1^* & C_1 & C_2^* & C_2 & C_3^* & C_3 \\ \hline C_2 & C_2^* & C_3 & C_3^* & C_1 & C_1^* \\ C_2^* & C_2 & C_3^* & C_3 & C_1^* & C_1 \\ \hline C_3 & C_3^* & C_1 & C_1^* & C_2 & C_2^* \\ C_3^* & C_3 & C_1^* & C_1 & C_2^* & C_2 \end{array}\right].$$

where C_1, C_2 and C_3 are conjugate-skew-circulant matrices.

Viewing f as indexed by (5) and (6) with $p3$ symmetry, we have that the corresponding column vector $\underline{X}$ is of the following form.

$$(8) \qquad \underline{X} = \begin{bmatrix} X_1 \\ X_1 \\ X_1 \end{bmatrix}$$

where X_1 is the length $n = (P^2 - 1)/3$ subvector of $\underline{X}$. By theorem 1, the Fourier transform $\underline{Y}$ of $\underline{X}$ has a similar structure. ($\underline{Y}$ is of the same structure upto permutation. However, the essential information is contained in the first subvector of length n.) Denote by Y_1 the first of the three subvectors of $\underline{Y}$.

Then $\underline{Y}$ can be obtained from Y_1. The computation of Y_1 in turn can be made as follows:

$$(9) \qquad Y_1 = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_2 & C_2^* & C_3 & C_3^* \\ C_1^* & C_1 & C_2^* & C_2 & C_3^* & C_3 \end{array}\right] \begin{bmatrix} X_1 \\ X_1 \\ X_1 \end{bmatrix} + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}$$

$$= H_1 X_1 + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}.$$

where H_1 can be found out as follows:

By the $p3$ symmetry we can reduce the input data to an asymmetric unit X_s including elements $X(0)$ and X_1. Thus the size of input data reduces from $3n+1$ to $n+1$, $n = (P^2-1)/3$. We set

$$U = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix}, \tag{10}$$

and

$$V = \begin{bmatrix} 1 & \\ & I_n \\ & I_n \\ & I_n \end{bmatrix}. \tag{11}$$

By matrix computation, we obtain

$$U_s = V^t U V = \begin{bmatrix} 1 & 3 \cdot 1_n^t \\ 3 \cdot 1_n & 3 \cdot H_1 \end{bmatrix}, \tag{12}$$

where

$$H_1 = \begin{bmatrix} C_1 + C_2 + C_3 & C_1^* + C_2^* + C_3^* \\ C_1^* + C_2^* + C_3^* & C_1 + C_2 + C_3 \end{bmatrix}$$

From variant 2 of section 3.5, we have proved that H_1 is a skew-circultant matrix which is conjugate-skew-circulancy.

Now the input data is

$$\underline{X} = V \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma^3) \\ \vdots \\ X(\gamma^{3n-3}) \end{bmatrix}, \tag{13}$$

and the output data is

$$\begin{bmatrix} Y(0) \\ Y(1) \\ \vdots \\ Y(n) \end{bmatrix} = K_s V^t U V \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma^3) \\ \vdots \\ X(\gamma^{3n-3}) \end{bmatrix} = K_s U_s \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma^3) \\ \vdots \\ X(\gamma^{3n-3}) \end{bmatrix}. \tag{14}$$

where

$$K_s = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{3} I_{(P^2-1)/3} \end{bmatrix}. \tag{15}$$

By applying the multiplicative algorithm, we have presented that the computation of Fourier transform with p3 symmetry has been changed to asymmetry unit with only one-third size. The original $P^2 \times P^2$ two-dimensional Fourier transform matrix has changed to the $\frac{P^2-1}{3} \times \frac{P^2-1}{3}$ skew-circulant matrix. The efficiency of this algorithm is apparent since the computation operation is divided by the degree of symmetry against the original non-symmetry case.

Furthermore, since H_1 is conjugate-skew-circulancy, it can be block-diagonalized as shown in (14) of section 3.5.

$$H_1 = \begin{bmatrix} C_1 + C_2 + C_3 & C_1^* + C_2^* + C_3^* \\ C_1^* + C_2^* + C_3^* & C_1 + C_2 + C_3 \end{bmatrix}$$

$$(16) \quad = (F(2) \otimes I_m) \begin{bmatrix} (H_{11} + H_{11}^*)/2 & \\ & (H_{11} - H_{11}^*)/2 \end{bmatrix} (F(2) \otimes I_m) ,$$

where $(H_{11} + H_{11}^*)/2$ is real skew-circulant matrix of cosine funtion and $(H_{11} - H_{11}^*)/2$ is negative-skew-circulant matrix of pure imaginary sine function which can be changed to skew-circulant matrix again. This step has simplified the computation further.

Variant 2

Similarly we also can derive the circulant structure from variant 2 of section 4.2. For $P \equiv 2(3)$, take $n = \frac{P^2-1}{3}$. The element γ^3 is of order n and generates the subgroup

$$(17) \qquad S = \{\gamma^0, \gamma^3, \gamma^6, \cdots, \gamma^{3(n-1)}\}$$

$U(P)$ is the multiplicative group of nonzero elements of $\mathcal{Z}/P[\rho]$, and

$$(18) \qquad U(P) = S \cup \gamma^n S \cup \gamma^{2n} S .$$

Now we rewrite the subgroup S as

$$(19) \qquad S' = \{\gamma^{3(n-1)}, \gamma^{3(n-2)}, \cdots, \gamma^3, \gamma^0\} .$$

Then $U(P)$ can be rewritten as

$$(20) \qquad U(P) = \gamma^{2n} S' \cup \gamma^n S' \cup S' .$$

View S and S' as ordered by (17) and (19). These orderings can be used to order $\gamma^n S, \gamma^{2n} S$ and $\gamma^{2n} S', \gamma^n S'$. We will denote by $R_1(P)$ with $S; \gamma^n S; \gamma^{2n} S$ and $R_2(P)$ with $\gamma^{2n} S'; \gamma^n S'; S'$ instead of the original $U(P)$. We will order the column indexing by $R_1(P)$ and order the row indexing by $R_2(P)$. Then the matrix $W(P)$ which has been derived in (28) of section 3.9 is

$$W(P) = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_3 & C_3^* & C_2 & C_2^* \\ C_1^* & C_1 & C_3^* & C_3 & C_2^* & C_2 \\ \hline C_2 & C_2^* & C_1 & C_1^* & C_3 & C_3^* \\ C_2^* & C_2 & C_1^* & C_1 & C_3^* & C_3 \\ \hline C_3 & C_3^* & C_2 & C_2^* & C_1 & C_1^* \\ C_3^* & C_3 & C_2^* & C_2 & C_1^* & C_1 \end{array}\right] . \tag{21}$$

where C_1, C_2 and C_3 are conjugate-circulant matrices.

Similarly like variant 1, we can find out

$$Y_1 = \left[\begin{array}{cc|cc|cc} C_1 & C_1^* & C_3 & C_3^* & C_2 & C_2^* \\ C_1^* & C_1 & C_3^* & C_3 & C_2^* & C_2 \end{array}\right] \left[\begin{array}{c} X_1 \\ X_1 \\ X_1 \end{array}\right] + \left[\begin{array}{c} X(0) \\ \vdots \\ X(0) \end{array}\right] \tag{22}$$

$$= H_1 X_1 + \left[\begin{array}{c} X(0) \\ \vdots \\ X(0) \end{array}\right] .$$

and the output data is

$$\left[\begin{array}{c} Y(0) \\ Y(1) \\ \vdots \\ Y(n) \end{array}\right] = K_s U_s \left[\begin{array}{c} X(0) \\ X(1) \\ X(\gamma^3) \\ \vdots \\ X(\gamma^{3n-3}) \end{array}\right] . \tag{23}$$

where

$$K_s = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{3}I_{(P^2-1)/3} \end{bmatrix}. \tag{24}$$

and

$$U_s = \begin{bmatrix} 1 & 3 \cdot 1_n^t \\ 3 \cdot 1_n & 3 \cdot H_1 \end{bmatrix}, \tag{25}$$

where

$$H_1 = \begin{bmatrix} C_1 + C_3 + C_2 & C_1^* + C_3^* + C_2^* \\ C_1^* + C_3^* + C_2^* & C_1 + C_3 + C_2 \end{bmatrix} \tag{26}$$

The difference comparing with variant 1 is that H_1 is circulant matrix with block-circulance. Now the original $P^2 \times P^2$ two-dimensional Fourier transform matrix has changed to the $\frac{P^2-1}{3} \times \frac{P^2-1}{3}$ circulant matrix. The efficiency of this algorithm is apparent since the computation operation is a cyclic convolution of one-third size.

Furthermore, since H_1 is conjugate-circulancy, it can be block-diagonalized as shown in (6) of section 3.10.

$$H_1 = \begin{bmatrix} C_1 + C_3 + C_2 & C_1^* + C_3^* + C_2^* \\ C_1^* + C_3^* + C_2^* & C_1 + C_3 + C_2 \end{bmatrix} \tag{27}$$

$$= (F(2) \otimes I_m) \begin{bmatrix} (H_{11} + H_{11}^*)/2 & \\ & (H_{11} - H_{11}^*)/2 \end{bmatrix} (F(2) \otimes I_m),$$

where $(H_{11} + H_{11}^*)/2$ is real circulant matrix of cosine funtion and $(H_{11} - H_{11}^*)/2$ is negative-circulant matrix of pure imaginary sine function which can be changed to skew-circulant matrix again. This step has decreased the computation operations more.

6.4 An Example

We will continue with our example of the case $P = 5$ of section 3.8 and 3.12. The rest of our discussion will be computing the two-dimensional 5-point Fourier transform of data exhibitting $p3$ symmetry.

Thus the action of α on $\mathcal{Z}/5 \times \mathcal{Z}/5$ corresponds to the multiplication action by ρ on $\mathcal{Z}/5[\rho]$, i.e.,

$$\alpha(x,y) \to (4y, x+4y) \tag{1}$$

$$\rho(x+\rho y) \to \rho x + \rho^2(y) = 4y + \rho(x+4y) .$$

Then the redundancy in X-ray data of $p3$ symmetry is

$$f(x,y) = f(4y, x+4y) . \tag{2}$$

Viewing f as indexed by (6) or (18), we have that the corresponding column vector $\underline{X}$ is of the following form.

$$\underline{X} = \begin{bmatrix} X_1 \\ X_1 \\ X_1 \end{bmatrix} \tag{3}$$

where X_1 is the length 8 subvector of $\underline{X}$. The Fourier transform $\underline{Y}$ of $\underline{X}$ has a similar structure. The essential information is contained in the first subvector of length 8. Denote by Y_1 the first of the three subvectors of $\underline{Y}$.

Then $\underline{Y}$ can be obtained from Y_1. The computation of Y_1 by (9) or (22) is as follows:

$$Y_1 = H_1 X_1 + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix} . \tag{4}$$

where H_1 is shown as (12) or (26).

By the $p3$ symmetry we can reduce the input data to an asymmetric unit X_s. Thus the size of input data reduces from 25 to 9. Thus by (14) and (23) the output data is

$$\begin{bmatrix} Y(0) \\ Y(1) \\ \vdots \\ Y(8) \end{bmatrix} = K_s U_s \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma^3) \\ \vdots \\ X(\gamma^{21}) \end{bmatrix}. \tag{5}$$

where

$$K_s = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{3} I_8 \end{bmatrix}. \tag{6}$$

In case of variant 1, H_1 is a skew-circulant matrix with block-skew-circulance. C_1, C_2 and C_3 are shown as in (8) of section 3.8. While in case of variant 2, H_1 is a circulant matrix with block-circulance, C_1, C_2 and C_3 are the same as in (24) of section 3.12.

6.5 New Algorithm for $N = 3L$ with $p3$ Symmetry

In this section we will develop the new multiplicative $p3$ symmetry two-dimensional FFT algorithm of case $N = 3L$ where 3 and L are relative prime. By this algorithm the size of two-dimensional FFT can be extended from prime number to composite number for two-dimensional FFT with $p3$ symmetry.

We first build the $p3$ symmetry two-dimensional FFT algorithm of the 3×3 and $L \times L$ cases with ring structure or field structure. Then by using the Chinese Remainder Theorem(CRT) and tensor product technique we can

derive the efficient $3L \times 3L$ $p3$ symmetry two-dimensional FFT algorithm. We will give a typical example of size 12×12 to describe this efficient two-dimensional FFT algorithm for $p3$ symmetry. It also can be developed to the general case.

From section 3.3, we know that the polynomial x^2+x+1 is irreducible over $\mathcal{Z}/P$ for a prime $P \equiv 2(3)$, and $\mathcal{Z}/P[x]/(x^2+x+1)$ is a field. Otherwise when P equals to the other number, the polynomial $x^2 + x + 1$ is not irreducible over $\mathcal{Z}/P$. It means that $\mathcal{Z}/P[x]/(x^2 + x + 1)$ will be a quotient polynomial ring instead of a field. Set $\rho = exp(-2\pi i/3)$, i.e. $\rho^2 + \rho + 1 = 0$. We will use the notation $\mathcal{Z}/P[\rho]$ instead of $\mathcal{Z}/P[x]/(x^2 + x + 1)$. An element $a \in \mathcal{Z}/P[\rho]$ can be written uniquely as a polynomial $a_1 + \rho a_2$, a_1, $a_2 \in \mathcal{Z}/P$. Arithmetic in $\mathcal{Z}/P$ is taken mod $(\rho^2 + 1)$ defined by

$$(1) \qquad (a_1 + \rho a_2) + (b_1 + \rho b_2) \;=\; (a_1 + b_1) + \rho(a_2 + b_2)\,.$$

$$(a_1 + \rho a_2)(b_1 + \rho b_2) \;=\; a_1 b_1 - a_2 b_2 + \rho(a_2 b_1 + a_1 b_2 - a_2 b_2)\,.$$

We first build 3×3 two-dimensional FFT algorithm with $p3$ symmetry using ring structure. Define a mapping

$$(2) \qquad \phi : \mathcal{Z}/P[\rho] \to \mathcal{Z}/P$$

by

$$\phi(a_1 + \rho a_2) \;=\; a_1$$

Observe that

$$(3) \qquad \phi(a + b) \;=\; \phi(a) + \phi(b)\,,$$

$$\phi(ab) \;=\; a_1b_1 - a_2b_2\ .$$

For a function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$, the Fourier transform of X is

$$(4)\qquad FX(b_1,-b_2) \;=\; \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1,a_2)e^{\frac{-2\pi i}{P}(a_1b_1-a_2b_2)}$$
$$= \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P} X(a_1,a_2)e^{\frac{-2\pi i}{P}\phi(ab)}\ .$$

The function X and its Fourier Transform FX can be viewed as column vectors once $\mathcal{Z}/P \times \mathcal{Z}/P$ is ordered. Note that $(a_1, a_2) \in \mathcal{Z}/P \times \mathcal{Z}/P$ corresponds uniquely to $a_1 + \rho a_2 \in \mathcal{Z}/P[\rho]$. Hence an ordering of $\mathcal{Z}/P[\rho]$ will yield an ordering for $\mathcal{Z}/P \times \mathcal{Z}/P$.

Now we may view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent (4) in terms of matrices as

$$(5)\qquad \underline{Y} \;=\; M(P)\,\underline{X}$$

where

$$(6)\qquad M(P) \;=\; \left[\ \omega^{\phi(ab)}\ \right]\ ,$$

For 3×3 case, the indexing set on $\mathcal{Z}/3 \times \mathcal{Z}/3$ can be identified as follows:

$$(7)\qquad \{0,\ 1-\rho,\ -1+\rho,\ 1,\ \rho,\ \rho^2,\ -1,\ -\rho,\ -\rho^2\}\ ,$$

then the matrix $M(P)$ will be as

$$(8)\qquad M(3) \;=\; \begin{bmatrix} 1 & 1_2^t & 1_6^t \\ 1_2 & I(2) & C_1 \otimes 1_3^t \\ 1_6 & C_1 \otimes 1_3 & C \end{bmatrix}\ ,$$

where

$$I_2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \quad C_1 = \begin{bmatrix} \omega & \omega^2 \\ \omega^2 & \omega \end{bmatrix},$$

and

$$C = \begin{bmatrix} C_2 & C_2^* \\ C_2^* & C_2 \end{bmatrix}, \quad C_2 = \begin{bmatrix} \omega & 1 & \omega^2 \\ 1 & \omega^2 & \omega \\ \omega^2 & \omega & 1 \end{bmatrix}.$$

where $*$ we mean complex conjugation and $\omega = e^{-2\pi i/3}$.

Now we make use of the input data with $p3$ symmetry. By section 6.2, the action of α on $\mathcal{Z}/P \times \mathcal{Z}/P$ corresponds to the multiplication action by ρ on $\mathcal{Z}/P[\rho]$, that is

(9)
$$\alpha(x,y) \to (-y, x-y)$$
$$\rho(x+\rho y) \to \rho x + \rho^2(y) = -y + \rho(x-y)$$

Viewing X as indexing by (7) with $p3$ symmetry, we have that the corresponding column vector $\underline{X}$ is of the following form,

(10)
$$\begin{bmatrix} X(0,0) \\ X(1,2) \\ X(2,1) \\ X(1,0) \\ X(1,0) \\ X(1,0) \\ X(2,0) \\ X(2,0) \\ X(2,0) \end{bmatrix},$$

where $\{X(0,0), X(1,2), X(2,1), X(1,0), X(2,0)\}$ is asymmetry unit of $p3$ symmetry group with input data on $\mathcal{Z}/3 \times \mathcal{Z}/3$.

Define the matrix

$$A_3 = \begin{bmatrix} I_3 & & \\ & 1_3 & \\ & & 1_3 \end{bmatrix}, \tag{11}$$

and A_3^t is the transposed matrix of A_3.

By direct computation we have

$$R(3) = A_3^t M(3) A_3 = \begin{bmatrix} 1 & 1_2^t & 1_2^t \otimes 3 \\ 1_2 & I(2) & 3C_1 \\ 1_6 \otimes 3 & 3C_1 & 0 \end{bmatrix} \tag{12}$$

and the output asymmetry unit is

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(2) \\ 3Y(3) \\ 3Y(4) \end{bmatrix} = R(3) \begin{bmatrix} X(0,0) \\ X(1,2) \\ X(2,1) \\ X(1,0) \\ X(2,0) \end{bmatrix}. \tag{13}$$

Comparing with (8) we can see that this ring structure method is efficient since the data has been compressed and the computation is reduced. This is only a simple example which will be used in composite number 12×12 case. We can also extend this algorithm to more general case.

By the similar method we also can solve the 4×4 case. Take $\gamma = 1 + 3\rho$. The indexing set on $\mathcal{Z}/4 \times \mathcal{Z}/4$ can be defined as

$$(14) \qquad \{0, 2, 2\rho, 2\rho^2, 1, \gamma^3, \gamma^2, \gamma^5, \gamma^4, \gamma, -1, -\gamma^3, -\gamma^2, -\gamma^5, -\gamma^4, -\gamma\} ,$$

where $\{1, \gamma^3, \gamma^2, \gamma^5, \gamma^4, \gamma\}$ is a cyclic group. The generator is γ and $\gamma^6 = 1$.

By define the mapping

$$(15) \qquad \phi : \mathcal{Z}/P[\rho] \to \mathcal{Z}/P$$

by

$$\phi(a_1 + \rho a_2) \; = \; 2a_1 - a_2$$

we have

$$(16) \qquad M(4) \; = \; \begin{bmatrix} 1 & 1_3^t & 1_6^t & 1_6^t \\ 1_3 & I(3) & C_3 \otimes 1_2^t & C_3 \otimes 1_2^t \\ 1_6 & C_3 \otimes 1_2 & C_4 & C_4^* \\ 1_6 & C_3 \otimes 1_2 & C_4^* & C_4 \end{bmatrix} ,$$

where

$$C_3 = \begin{bmatrix} 1 & -1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix} , \quad C_4 = \begin{bmatrix} -1 & 1 & i & -i & i & i \\ 1 & -1 & -i & i & i & i \\ i & -i & i & i & -1 & 1 \\ -i & i & i & i & 1 & -1 \\ i & i & -1 & 1 & i & -i \\ i & i & 1 & -1 & -i & i \end{bmatrix} .$$

Viewing X as indexing by (14) with $p3$ symmetry, we can find the asymmetry unit of input data on $\mathcal{Z}/4 \times \mathcal{Z}/4$ as

(17) $$\{X(0,0), X(2,0), X(1,0), X(1,2), X(3,0), X(3,2)\}$$

Define the matrix

(18) $$A_4 = \begin{bmatrix} 1 & & & & \\ & 1_3 & & & \\ & & 1_3 \otimes I_2 & & \\ & & & 1_3 \otimes I_2 & \end{bmatrix},$$

and set A_4^t as the transposed matrix of A_4.

Then we have that

(19) $$R(4) = A_4^t M(4) A_4$$

$$= \begin{bmatrix} 1 & 3 & 1_2^t \otimes 3 & 1_2^t \otimes 3 \\ 3 & 9 & 1_2^t \otimes -3 & 1_2^t \otimes -3 \\ 1_2 \otimes 3 & 1_2 \otimes -3 & C_5 & C_5^* \\ 1_2 \otimes 3 & 1_2 \otimes -3 & C_5^* & C_5 \end{bmatrix},$$

where

$$C_5 = \begin{bmatrix} 3(-1+2i) & 3 \\ 3 & 3(-1-2i) \end{bmatrix}.$$

thus the output asymmetry unit will be

$$
(20) \qquad \begin{bmatrix} Y(0) \\ 3Y(1) \\ 3Y(2) \\ 3Y(3) \\ 3Y(4) \\ 3Y(5) \end{bmatrix} = R(4) \begin{bmatrix} X(0,0) \\ X(2,0) \\ X(1,0) \\ X(1,2) \\ X(3,0) \\ X(3,2) \end{bmatrix} .
$$

compare with (16), we also see that it is a efficient method for 4×4 case.

By the property of $p3$ symmetry group, we have developed the 3×3 and 4×4 $p3$ symmetry algorithms with ring structure. Based on this efficient 3×3 and $L \times L$ symmetry algorithm we can develop the new multiplicative two-dimensional $p3$ symmetry FTT algorithm for $N = 3L$ composite number case, with $(L, 3) = 1$, using the Chinese Remainder Theorem(CRT) with tensor product method.

By applying the Chinese remainder theorem to Fourier transform computation, we will provide a way using Chinese remainder theorem to compute a large size two-dimensional Fourier transform computation nested with the small size two-dimensional computation.

By the Chinese remainder theorem from setion 5.2, for $N = 3L$ and $(L, 3) = 1$, we define the following mappings

$$
(21) \qquad \phi : \mathcal{Z}/L \times \mathcal{Z}/3 \rightarrow \mathcal{Z}/N ,
$$

by the formula

$$
(22) \qquad \phi(u_1, u_2) = (u_1 e_1 + u_2 e_2) \ mod N ,
$$

where $u_1 \in \mathcal{Z}/L$ and $u_2 \in \mathcal{Z}/3$, the $\{e_1, e_2\}$ is the system of idempotents

corresponding to the factorization $N = 3L$. And the inverse mapping ϕ^{-1} is given by

$$\phi^{-1} : \mathcal{Z}/N \to \mathcal{Z}/L \times \mathcal{Z}/3 , \tag{23}$$

that is

$$\phi^{-1}(u) = (u \bmod L,\ u \bmod 3) , \quad u \in \mathcal{Z}/N . \tag{24}$$

which identifies that every element $u \in \mathcal{Z}/N$ can be identified uniquely with the ordered pair $(u \bmod L,\ u \bmod 3)$ in $\mathcal{Z}/L \times \mathcal{Z}/3$.

Denote $\mathcal{Z}/N \times \mathcal{Z}/N$ by $(\mathcal{Z}/N)^2$. The mapping ϕ and ϕ^{-1} can be extended to

$$\psi : (\mathcal{Z}/L)^2 \times (\mathcal{Z}/3)^2 \to (\mathcal{Z}/N)^2 , \tag{25}$$

$$\psi((u_1, v_1), (u_2, v_2)) = (\phi(u_1, u_2), \phi(v_1, v_2)) .$$

And

$$\psi^{-1} : (\mathcal{Z}/N)^2 \to (\mathcal{Z}/L)^2 \times (\mathcal{Z}/3)^2 , \tag{26}$$

$$\psi^{-1}(u, v) = (\phi^{-1}(u), \phi^{-1}(v)) ,$$

By theorem 1 of section 8.2 and the Chinese Remainder Theorem we can find out the asymmetric unit of $p3$ symmetric group on $\mathcal{Z}/N \times \mathcal{Z}/N$. Then the Fourier transform of X can be found out by

$$Y_N(s) = R(N)\, X_N(s) = (A_N^t (M(L) \otimes M(3)) A_N)\, X_N(s) , \tag{27}$$

where $M(L)$ and $M(3)$ are non-symmetry FFT matrices as shown in (8) and (16).

For the 12×12 case, the output of Fourier transform is

$$Y_{12}(s) = R(12)\, X_{12}(s)\ , \tag{28}$$

where

$$R(12) \ = \ A_{12}^{t}(M(4) \otimes M(3))A_{12}\ . \tag{29}$$

Define the matrix

$$A_{12}^{t} \ = \ \begin{bmatrix} I_9 & & & & & & & & & & & \\ & I_9 & S & S^2 & & & & & & & & \\ & & & & I_2 \otimes I_9 & I_2 \otimes S & I_2 \otimes S^2 & & & & & \\ & & & & & & & I_2 \otimes I_9 & I_2 \otimes S & I_2 \otimes S^2 \end{bmatrix}\ , \tag{30}$$

and A_{12}^{t} is the transposed matrix of A_{12}. By using the matrix properties and tensor product theory continuely, our algorithm will be described as a matrix factorization. This tensor product factorization decomposes the action of $R(12)$ into a sequence of elementary operations which is useful in matching different computer architecture.

Define the matrix

$$\mathcal{S} = I_9 \oplus I_9 \oplus S \oplus S^2 \oplus I_9 \oplus I_9 \oplus S \oplus S \oplus S^2 \oplus S^2 \oplus I_9 \oplus I_9 \oplus S \oplus S \oplus S^2 \oplus S^2 \tag{31}$$

and $\mathcal{S}^{-1}$ as inverse matrix of $\mathcal{S}$. Then

$$R(12) \ = \ (A_4^{t} \otimes I_9)\, \mathcal{S}^{-1}\, (M(4) \otimes M(3))\, \mathcal{S}\, (A_4 \otimes I_9) \tag{32}$$

where A_4, $M(3)$ and $M(4)$ are the same as in (18),(8) and (16), and

$$(33) \qquad S = \begin{bmatrix} I_3 & & \\ & S_3 & \\ & & S_3 \end{bmatrix}, \quad S_3 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

Define another matrix

$$(34) \qquad F = I_{16} \otimes F_0 ,$$

where

$$(35) \qquad F_0 = \begin{bmatrix} I_3 & & \\ & F(3) & \\ & & F(3) \end{bmatrix},$$

and $F(3)$ is one-dimensional 3-point FFT matrix. Then

$$R(12) = (A_4^t \otimes I_9)F(F^{-1}\mathcal{S}^{-1}F)(F^{-1}(M(4) \otimes M(3)F^{-1})(F\mathcal{S}F^{-1})F(A_4 \otimes I_9)$$

$$= (I_6 \otimes F_0)(A_4^t \otimes I_9)\mathcal{T}\!\!\!/(I_{16} \otimes F_0^{-1}M(3)F_0^{-1})(M(4) \otimes I_9)\mathcal{T}\!\!\!/(A_4 \otimes I_9)(I_6 \otimes F_0)$$

where

$$(36) \quad \mathcal{T}\!\!\!/ = I_9 \oplus I_9 \oplus D \oplus D^2 \oplus I_9 \oplus I_9 \oplus D \oplus D \oplus D^2 \oplus D^2 \oplus I_9 \oplus I_9 \oplus D \oplus D \oplus D^2 \oplus D^2$$

and

$$(37) \qquad D = I_3 \oplus 1 \oplus \omega \oplus \omega^2 \oplus 1 \oplus \omega \oplus \omega^2$$

From that we have

$$R(12) = P_0(F_0 \otimes I_6)(I_9 \otimes A_4^t)\mathcal{T}\!\!\!/'(F_0^{-1}M(3)F_0^{-1} \otimes I_{16})(I_9 \otimes M(4))\mathcal{T}\!\!\!/'(I_9 \otimes A_4)(F_0 \otimes I_6)P_0^{-1}$$

where P_0 is the permutation matrix and

$$(38) \qquad \mathcal{T}\!\!\!/' = I_{16} \oplus I_{16} \oplus I_{16} \oplus I_{16} \oplus D_1 \oplus D_1^2 \oplus I_{16} \oplus D_1 \oplus D_1^2 ,$$

and

$$D_1 = 1 \oplus D_3 \oplus (D_3 \otimes I_2) \oplus (D_3 \otimes I_2)\,, \tag{39}$$

with

$$D_3 = 1 \oplus \omega \oplus \omega^2\,. \tag{40}$$

then the matrix $R(12)$ will be

$$R(12) = P_0(F_0 \otimes I_6)(F_0^{-1} M(3) F_0^{-1} \otimes I_6)\{(I_9 \otimes A_4^t)\mathcal{D}'(I_9 \otimes M(4))\mathcal{D}'(I_9 \otimes A_4)\}(F_0 \otimes I_6)P_0^{-1}$$

Define

$$F_2 = \begin{bmatrix} I_2 & \\ & F(2) \otimes I_2 \end{bmatrix}, \tag{41}$$

finally we have

$$R(12) = P_0(I_9 \otimes F_2)(F_0 \otimes I_6)(F_0^{-1} M(3) F_0^{-1} \otimes I_6)\mathcal{Z}(F_0 \otimes I_6)(I_9 \otimes F_2)P_0^{-1} \tag{42}$$

where

$$F_0^{-1} M(3) F_0^{-1} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & \omega & 0 & 0 & \omega^2 & 0 & 0 \\ 1 & 1 & 1 & \omega^2 & 0 & 0 & \omega & 0 & 0 \\ 1 & \omega & \omega^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \omega^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \omega & 0 & 0 & 0 \\ 1 & \omega^2 & \omega & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \omega^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \omega \end{bmatrix}, \tag{43}$$

and

$$\mathcal{Z} = Z_4 \oplus Z_4 \oplus Z_4 \oplus Z_4' \oplus Z_4'' \oplus Z_4 \oplus Z_4' \oplus Z_4'' \tag{44}$$

where

$$Z_4 = \begin{bmatrix} 1 & 3 & 3 & 3 & 0 & 0 \\ 1 & 3 & -1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 \\ 1 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2i & 0 \\ 0 & 0 & 0 & 0 & 0 & 2i \end{bmatrix},$$

$$Z_4' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 \\ 0 & 2 & -1 & 1 & 0 & 0 \\ 0 & 2 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -i & -\sqrt{3} \\ 0 & 0 & 0 & 0 & -\sqrt{3} & -i \end{bmatrix},$$

$$Z_4'' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 \\ 0 & 2 & -1 & 1 & 0 & 0 \\ 0 & 2 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -i & \sqrt{3} \\ 0 & 0 & 0 & 0 & \sqrt{3} & -i \end{bmatrix}.$$

From the final tensor product decompositions (42) of this new FFT algorithm, the action of $R(12)$ is decomposed into a sequence of elementary operations of vectorization and parallelization. For example $\mathcal{Z}$ has block-diagonal structure with each independent simplified block. This algorithm is more valuable with different machine architectures.

6.6 Summary

As the applications of our algorithm to X-ray data of the crystallographic symmetric group, we have described the multiplicative two-dimensional prime point Fourier transform algorithm for $p3$ symmetry using field structure in section 6.3 and 6.4. Our algorithm adapts naturally to processing a data with $p3$ symmetry when $P \equiv 2(3)$. Since the crystallographic symmetry giving rise to data redundancy, our algorithm reduce the data to one-third size, i.e. the original $P^2 \times P^2$ two-dimensional Fourier transform matrix has changed to the $\frac{P^2-1}{3} \times \frac{P^2-1}{3}$ circulant or skew-circulant matrix. The efficiency of this algorithm is apparent since the data compression and the computation operation is divided by the degree of symmetry against the original non-symmetry case.

In section 6.5 we have developed the new two-dimensional multiplicative $p3$ symmetry FFT algorithm of case $N = 3L$ with $(L, 3) = 1$ by ring structure. By using the CRT and tensor product technique we have derived the efficient algorithm from prime to composite number which is valuable with different computer architectures.

Chapter 7

Multiplicative 2-Dimensional Prime Point FFT Algorithm for $p4$ Symmetry

7.1 Introduction

In last chapter we have described the multiplicative two-dimensional prime point FFT algorithm for $p3$ symmetry. In this chapter we will present the multiplicative two-dimensional prime point FFT algorithm for $p4$ symmetry. The redundancy now is controlled by the crystallographic symmetry group $p4$. The $p4$ symmetry are also known as 90° rotation symmetry. As another application of our algorithm developed before, we will illustrate incorporating of the crystallographic $p4$ symmetry to the efficient multiplicative two-dimensional prime point FFT algorithm to gain computational advantage.

7.2 p4 Symmetry

The $p4$ symmetry may be represented as the following group of 2×2 matrices.

$$(1) \qquad S_{p4} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \right\}$$

Denoting a generator of $p4$ by α, say

$$(2) \qquad \alpha = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

then S_{p4} is cyclic group given by

$$(3) \qquad S_{p4} = \{I_2 = \alpha^4, \alpha, \alpha^2, \alpha^3\}.$$

the redundancy in X-ray data is

$$f(x,y) = f(\alpha(x,y)) = f(-y,x),$$

$$f(x,y) = f(\alpha^2(x,y)) = f(-x,-y),$$

and

$$(4) \qquad f(x,y) = f(\alpha^3(x,y)) = f(y,-x),$$

where f is the function whose value is the X-ray data at the lattice point (x,y). Thus $p4$ can be viewed as acting on a two-dimentional lattice via the matrix multiplication.

Denote the Fourier transform of f by $\hat{f}$. According to the theorem 1 of section 7.2, we have the following lemma for $p4$ case.

Lemma 1 *For a function f defined on $\mathcal{Z}/N \times \mathcal{Z}/N$, if $f(x,y) = f(\alpha(x,y))$, i.e. f has $p4$ symmetry. Then*

$$(5) \qquad \hat{f}(u,v) = \hat{f}(\alpha(u,v)).$$

Prove

According to the theorem 1 of section 6.2, we have

$$\hat{f}(u,v) \;=\; \hat{f}(\alpha^{\#}(u,v))\,,$$

where $\alpha^{\#}$ denotes the inverse transpose of the matrix α, $\alpha^{\#} = (\alpha^{-1})^t$. Since

$$\alpha^{\#} = (\alpha^{-1})^t = \left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}^{-1}\right)^t = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = \alpha\,.$$

Then

$$\hat{f}(u,v) \;=\; \hat{f}(\alpha^{\#}(u,v)) \;=\; \hat{f}(\alpha(u,v))\,.$$

Observe now that the minimal polynomial of α in the variable λ is

$$\lambda^2 + 1\,. \tag{6}$$

Thus the action of α on $\mathcal{Z}/P \times \mathcal{Z}/P$ corresponds to the multiplication action by ρ on $\mathcal{Z}/P[\rho]$, i.e.,

$$\alpha(x,y) \to (-y,x) \tag{7}$$

$$\rho(x+\rho y) \to \rho x + \rho^2(y) \;=\; -y + \rho x\,.$$

7.3 New Algorithm for p4 Symmetry

From chapter 4 we treat $\mathcal{Z}/P \times \mathcal{Z}/P$ for a prime number $P \equiv 3(4)$, as a simple object. The polynomial $x^2 + 1$ is irreducible over $\mathcal{Z}/P$ for $P \equiv 3(4)$.

Set $\rho = exp(-2\pi i/4)$. Then $\rho^2+1 = 0$ and $\mathcal{Z}/P[\rho]$ is a field with P^2 elements; i.e., the nonzero elements of $\mathcal{Z}/P[\rho]$ form a cyclic group under multiplication. Let γ be a cyclic generator. Then

$$1 = \gamma^0 , \gamma , \gamma^2 , \cdots \gamma^{P^2-2} \tag{1}$$

are the distinct nonzero elements of $\mathcal{Z}/P[\rho]$

An element $a \in \mathcal{Z}/P[\rho]$ can be written uniquely as $a_1 + \rho a_2$, $a_1, a_2 \in \mathcal{Z}/P$. Arithmetic in $\mathcal{Z}/P[\rho]$ is defined by

$$a + b \;=\; (a_1 + \rho a_2) + (b_1 + \rho b_2) \;=\; (a_1 + b_1) + \rho(a_2 + b_2) \,. \tag{2}$$

$$ab \;=\; (a_1 + \rho a_2)(b_1 + \rho b_2) \;=\; a_1 b_1 - a_2 b_2 + \rho(a_1 b_2 + a_2 b_1) \,.$$

Define a mapping ϕ: $\mathcal{Z}/P[\rho] \rightarrow \mathcal{Z}/P$ by

$$\phi(a_1 + \rho a_2) \;=\; 2a_1 \,. \tag{3}$$

$$\phi(ab) \;=\; 2(a_1 b_1 - a_2 b_2)$$

For a function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$, the Fourier transform of X is

$$\begin{aligned} FX(2b_1, -2b_2) \;&=\; \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P)} X(a_1, a_2) e^{\frac{-2\pi i}{P}(2a_1 b_1 - 2a_2 b_2)} \\ &=\; \sum_{(a_1,a_2)\in \mathcal{Z}/P\times\mathcal{Z}/P)} X(a_1, a_2) e^{\frac{-2\pi i}{P}\phi(ab)} \end{aligned} \tag{4}$$

The function X and its Fourier Transform FX can be viewed as column vectors once $\mathcal{Z}/P \times \mathcal{Z}/P$ is ordered. To this end, note that $(a_1, a_2) \in \mathcal{Z}/P \times$

$\mathcal{Z}/P$ corresponds uniquely to $a_1+\rho a_2 \in \mathcal{Z}/P[\rho]$. Hence an ordering of $\mathcal{Z}/P[\rho]$ will yield an ordering for $\mathcal{Z}/P \times \mathcal{Z}/P$.

Take the ordering by $U(P)$. Now we may view the function X defined on $\mathcal{Z}/P \times \mathcal{Z}/P$ and its Fourier transform FX as column vectors, and represent (4) in terms of matrices as

$$\begin{bmatrix} Y(0) \\ \underline{Y} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix} \begin{bmatrix} X(0) \\ \underline{X} \end{bmatrix}, \tag{5}$$

where

$$W(P) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \le j,k < P^2-1} \tag{6}$$

and the element $0 \in \mathcal{Z}/P[\rho]$ has been placed in front of $U(P)$. For the rest of our discussion, we ignore $0 \in \mathcal{Z}/P[\rho]$ (We can place this at last stage of computation).

Variant 1

By lemma 1 of section 5.2, for $P \equiv 3(4)$, P^2-1 is divisible by 8. Take $n = \frac{P^2-1}{2}$ and $m = \frac{n}{2}$, and define the set

$$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{m-1}\}. \tag{7}$$

Thus we have that $1, \gamma^m, \gamma^{2m}, \gamma^{3m}$ is the quotient of $U(P)/S$ and

$$U(P) = S \cup \gamma^m S \cup \gamma^{2m} S \cup \gamma^{3m} S \tag{8}$$

View S as ordered by (7). This ordering can be used to order $\gamma^m S, \gamma^{2m} S$ and $\gamma^{3m} S$. Thus the ordering of $U(P)$ is obtained by putting $S; \gamma^m S; \gamma^{2m} S; \gamma^{3m} S$. We will denote by $R(P)$ the group $U(P)$ with this ordering. Now examine the matrix $W(P)$. set $\omega = e^{\frac{-2\pi i}{P}}$.

Theorem 1 *W(P) will be of the form*

$$W(P) = \begin{bmatrix} A & B & A^* & B^* \\ B & A^* & B^* & A \\ A^* & B^* & A & B \\ B^* & A & B & A^* \end{bmatrix}, \tag{9}$$

where A,B are $m \times m$ matrices.

Prove

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 & W_3 & W_4 \\ W_5 & W_6 & W_7 & W_8 \\ W_9 & W_{10} & W_{11} & W_{12} \\ W_{13} & W_{14} & W_{15} & W_{16} \end{bmatrix},$$

where W_i, $1 \leq i \leq 16$ are $m \times m$ matrices.

We will show that $W_1 = W_8 = W_{11} = W_{14}$. The other cases are proved in exactly the same way. For $0 \leq k, l < m$, denote the k-th row l-th column entry of an $m \times m$ matrix W_i by $W_i(k,l)$, $1 \leq i \leq 16$.

$$W_8(k,l) = \omega^{\phi(\gamma^{3m}\gamma^k\gamma^m\gamma^l)} = \omega^{\phi(\gamma^{k+l}\gamma^{4m})} = \omega^{\phi(\gamma^{k+l})} = W_1(k,l)\ .$$

Similarly by the properties of γ and ϕ shown in (1) to (3) of section 4.2, we also can prove that

$$W_{11} = W_1 = W_{14}$$

It means that $W(P)$ can be rewrite as

$$W(P) = \begin{bmatrix} W_1 & W_2 & W_3 & W_4 \\ W_2 & W_3 & W_4 & W_1 \\ W_3 & W_4 & W_1 & W_2 \\ W_4 & W_1 & W_2 & W_3 \end{bmatrix} . \tag{10}$$

This structure of $W(P)$ is known as block-skew-circulancy.

Let us now examine each of the blocks W_i, $1 \leq i \leq 4$, as follows: We see that

$$W_3^* = \omega^{-\phi(\gamma^k \gamma^{2m} \gamma^l)} = \omega^{\phi(\gamma^{k+l})} = W_1 .$$

similarly we can prove $W_4^* = W_2$, completing the prove of theorem 1.

Lemma 1 *Following by the theorem 1, the matrix*

$$H_1 = \begin{bmatrix} A + A^* + B + B^* \end{bmatrix} . \tag{11}$$

is skew-circulant matrix of pure real number.

Prove

A, A^*, B and B^* are $m \times m$ matrix, so H_1 is also $m \times m$ matrix. Denote the k-th row l-th column entry of the $m \times m$ matrix M by $M(k,l)$. Then

(1) For $0 \leq k, l < m-1$, we knew that the matrix $W(P)$ is a skew-circulant matrix by the corollary 2 of section 3.4. Since the matrix H_1 is the linear combination of submatrices A, A^*, B and B^*, so

$$H_1(k, l+1) = H_1(k+1, l) ,$$

(2) Since $W(P)$ is the skew-circulant matrix, we have

$$A(k, m-1) = B(k-1, 0) , \quad B(k, m-1) = A^*(k-1, 0) ,$$

$$A^*(k,m-1) \;=\; B^*(k-1,0)\,, \quad B^*(k,m-1) \;=\; A(k-1,0)\,.$$

where $1 \leq k < m$. Now look at the matrix H_1, we have

$$H_1(k,m-1) = A(k,m-1) + B(k,m-1) + A^*(k,m-1) + B^*(k,m-1)$$

$$= B(k-1,0) + A^*(k-1,0) + B^*(k-1,0) + A(k-1,0) = H_1(k-1,0)\,.$$

hence H_1 is the skew-circulant matrix. Furthermore since A^* and B^* are conjugates of A and B, H_1 is real, completing the proof of lemma 1.

Viewing the input data function f as indexed by (7) and (8) with $p4$ symmetry, we have that the corresponding column vector $\underline{X}$ is of the following form

$$\underline{X} \;=\; \begin{bmatrix} X_1 \\ X_1 \\ X_1 \\ X_1 \end{bmatrix} \tag{12}$$

where X_1 is the length $m = \frac{P^2-1}{4}$ subvector of $\underline{X}$. By lamma 1 of section 7.2 the Fourier transform $\underline{Y}$ of $\underline{X}$ has a similar structure. However, the essential information is contained in the first subvector of length m. Denote by $\underline{Y_1}$ the first of the four subvectors of $\underline{Y}$.

Then $\underline{Y}$ can be obtained from Y_1. The computation of Y_1 in turn can be made as follows:

$$Y_1 \;=\; \begin{bmatrix} A & B & A^* & B^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_1 \\ X_1 \\ X_1 \end{bmatrix} + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix} \tag{13}$$

$$= H_1 X_1 + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}.$$

By the $p4$ symmetry we can reduce the input data to an asymmetric unit X_s including $X(0)$ and X_1. Thus the size of input data reduces from $4m+1$ to $m+1$. H_1 can be found out as follows. Set

$$U = \begin{bmatrix} 1 & 1 & \cdots\cdots & 1 \\ 1 & & & \\ \vdots & & W(P) & \\ 1 & & & \\ 1 & & & \end{bmatrix}, \tag{14}$$

and

$$V = \begin{bmatrix} 1 & & & \\ & I_m & & \\ & I_m & & \\ & I_m & & \\ & I_m & & \end{bmatrix}. \tag{15}$$

By matrix computation, we obtain

$$U_s = V^t U V = \begin{bmatrix} 1 & 4 \cdot 1_m^t \\ 4 \cdot 1_m & 4 \cdot H_1 \end{bmatrix}, \tag{16}$$

where

$$H_1 = \begin{bmatrix} A + A^* + B + B^* \end{bmatrix},$$

from lemma 1 of section 7.3 we know that H_1 is skew-circulant matrix of pure real number.

Now the input data is

$$\underline{X} = V \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma) \\ \vdots \\ X(\gamma^{m-1}) \end{bmatrix}, \tag{17}$$

and the output data is

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(\gamma) \\ \vdots \\ Y(\gamma^{m-1}) \end{bmatrix} = K_s V^t U V \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma) \\ \vdots \\ X(\gamma^{m-1}) \end{bmatrix} \tag{18}$$

$$= K_s U_s \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma) \\ \vdots \\ X(\gamma^{m-1}) \end{bmatrix}.$$

where

$$U_s = \begin{bmatrix} 1 & 4 \cdot 1_m^t \\ 4 \cdot 1_m & 4 \cdot H_1 \end{bmatrix}, \tag{19}$$

and H_1 is as

$$H_1 = \begin{bmatrix} A + A^* + B + B^* \end{bmatrix}, \tag{20}$$

also

$$K_s = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{4}I_{(P^2-1)/4} \end{bmatrix} . \tag{21}$$

By applying the multiplicative algorithm, we find that the computation of Fourier transformwith $p4$ symmetry has been changed to asymmetry unit with only one-fourth size. Since the original $P^2 \times P^2$ two-dimensional Fourier transform matrix of complex number has change to a $\frac{P^2-1}{4} \times \frac{P^2-1}{4}$ skew-circulant matrix of real number, we gain computational advantage.

Variant 2

We will now to derive $W(P)$ depending on different ordering on $\mathcal{Z}/P \times \mathcal{Z}/P$ comparing with variant 1.

For $P \equiv 3(4)$, take $n = \frac{P^2-1}{2}$ and $m = \frac{n}{2}$, and define the set

$$S = \{\gamma^0, \gamma, \gamma^2, \cdots, \gamma^{m-1}\} . \tag{22}$$

then

$$U(P) = S \cup \gamma^m S \cup \gamma^{2m} S \cup \gamma^{3m} S . \tag{23}$$

We can rewrite the set S as

$$S' = \{\gamma^{m-1}, \gamma^{m-2}, \cdots, \gamma, \gamma^0\} . \tag{24}$$

Then $U(P)$ can be rewrite as

$$U(P) = \gamma^{3m} S' \cup \gamma^{2m} S' \cup \gamma^m S' \cup S' . \tag{25}$$

View S and S' as ordered by (20) and (22). We will denote by $R_1(P)$ with $S; \gamma^m S; \gamma^{2m} S; \gamma^{3m} S$ and $R_2(P)$ with $\gamma^{3m} S'; \gamma^{2m} S'; \gamma^m S'; S'$. We order the column and the row indexing by $R_1(P)$ and $R_2(P)$ separately. Then we examine the matrix $W(P)$ again.

Theorem 2 *W(P) will be of the form*

$$W(P) = \begin{bmatrix} A & B^* & A^* & B \\ B & A & B^* & A^* \\ A^* & B & A & B^* \\ B^* & A^* & B & A \end{bmatrix}, \tag{26}$$

where A,B are $m \times m$ matrices.

Prove

Let

$$W(P) = \begin{bmatrix} W_1 & W_2 & W_3 & W_4 \\ W_5 & W_6 & W_7 & W_8 \\ W_9 & W_{10} & W_{11} & W_{12} \\ W_{13} & W_{14} & W_{15} & W_{16} \end{bmatrix},$$

where W_i, $1 \leq i \leq 16$ is an $m \times m$ matrix. We will show that $W_4 = W_5 = W_{10} = W_{15}$. The other cases are proved in exactly the same way. For $0 \leq k, l < m$, denote the k-th row l-th column entry of an $m \times m$ matrix M by $M(k,l)$,

$$W_5(k,l) = \omega^{\phi(\gamma^{2m}\gamma^{m-1-k}\gamma^l)} = \omega^{\phi(\gamma^{4m}\gamma^{2m}\gamma^{m-1-k}\gamma^l)}$$

$$= \omega^{\phi(\gamma^{3m}\gamma^{m-1-k}\gamma^{3m}\gamma^l)} = W_4(k,l)\,.$$

Similarly by the properties of γ and ϕ shown in (1) to (3) of section 4.2, we also can prove that

$$W_{10} = W_4 = W_{15}$$

It means that $W(P)$ can be rewrite as

$$W(P) = \begin{bmatrix} W_1 & W_4 & W_3 & W_2 \\ W_2 & W_1 & W_4 & W_3 \\ W_3 & W_2 & W_1 & W_4 \\ W_4 & W_3 & W_2 & W_1 \end{bmatrix}. \tag{27}$$

This structure of $W(P)$ is known as block-circulancy.

Let us now examine each of the blocks W_i, $1 \leq i \leq 4$, as follows:

We can see that

$$W_9^* = \omega^{-\phi(\gamma^k \gamma^{2m} \gamma^l)} = \omega^{\phi(\gamma^{k+l})} = W_1 \, .$$

similarly we can prove $W_{13}^* = W_5$, completing the theorem 2.

Lemma 2 *Following the theorem 2, the matrix*

$$H_1 \; = \; \Big[\; A + A^* + B + B^* \; \Big] \, . \tag{28}$$

is a circulant matrix of pure real number.

Prove

A, A^*, B and B^* are $m \times m$ matrix, so H_1 is also $m \times m$ matrix. Denote the k-th row l-th column entry of the $m \times m$ matrix M by $M(k,l)$. Then

(1) For $0 \leq k, l < m-1$, we knew that the matrix $W(P)$ is a circulant matrix by the corollary 1 of section 3.9. Since the matrix H_1 is the linear combination of submatrices A, A^*, B and B^*, so

$$H_1(k+1, l+1) \; = \; H_1(k,l) \, ,$$

(2) Since $W(P)$ is the circulant matrix, we have

$$A(m-1,l) \; = \; B(0,l+1) \, , \quad B(m-1,l) \; = \; A^*(0,l+1) \, ,$$

$$A^*(m-1,l) \; = \; B^*(0,l+1) \, , \quad B^*(m-1,l) \; = \; A(0,l+1) \, .$$

where $1 \leq l < m$. Now look at the matrix H_1, we have

$$H_1(m-1,l) = A(m-1,l) + B(m-1,l) + A^*(m-1,l) + B^*(m-1,l)$$

$$= B(0,l+1) + A^*(0,l+1) + B^*(0,l+1) + A(0,l+1) = H_1(0,l+1)\,.$$

hence H_1 is the circulant matrix. Furthermore since A^* and B^* are conjugates of A and B, H_1 is real, completing the proof of lemma 2.

Similarly as the variant 1, the input data function f as indexed by (7) and (8) with $p4$ symmetry is of the following form

$$\underline{X} = \begin{bmatrix} X_1 \\ X_1 \\ X_1 \\ X_1 \end{bmatrix} \tag{29}$$

where X_1 is the length $m = \frac{P^2-1}{4}$ subvector of $\underline{X}$. The Fourier transform $\underline{Y}$ of $\underline{X}$ has a similar structure. However, the essential information is contained in the first subvector of length m. Denote by $\underline{Y_1}$ the first of the four subvectors of $\underline{Y}$.

Then $\underline{Y}$ can be obtained from Y_1. The computation of Y_1 in turn can be made as follows:

$$Y_1 = \begin{bmatrix} A & B & A^* & B^* \end{bmatrix} \begin{bmatrix} X_1 \\ X_1 \\ X_1 \\ X_1 \end{bmatrix} + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix} \tag{30}$$

$$= H_1 X_1 + \begin{bmatrix} X(0) \\ \vdots \\ X(0) \end{bmatrix}.$$

and the output data is

$$(31) \qquad \begin{bmatrix} Y(0) \\ Y(1) \\ Y(\gamma) \\ \vdots \\ Y(\gamma^{m-1}) \end{bmatrix} = K_s U_s \begin{bmatrix} X(0) \\ X(1) \\ X(\gamma) \\ \vdots \\ X(\gamma^{m-1}) \end{bmatrix}.$$

where K_s and U_s are the same as (19) and (21), but only defference is that

$$(32) \qquad H_1 = \begin{bmatrix} A + A^* + B + B^* \end{bmatrix}.$$

is the circulant matrix of pure real number.

By applying the multiplicative algorithm in variant 2, we find that the computation of Fourier transform with $p4$ symmetry has been changed to asymmetry unit with only one-fourth size. Furthermore the original $P^2 \times P^2$ two-dimensional Fourier transform matrix has change to a $\frac{P^2-1}{4} \times \frac{P^2-1}{4}$ circulant matrix of real number. It is more efficient that we only need compute the size of $\frac{P^2-1}{4}$ cyclic convolution instead of the original $P \times P$ two-dimensional Fourier transform after incorporating of the crystallographic $p4$ symmetry to our multiplicative two-dimensional prime point FFT algorithm.

7.4 An Example

We will continue with our example of the case $P = 3$ of section 4.4. The rest of our discussion will be computing the two-dimensional 3-point Fourier transform of data exhibitting $p4$ symmetry.

Denoting a generator of $p4$ by α, i.e.

$$(1) \qquad \alpha = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

Thus the action of α on $Z/3 \times Z/3$ corresponds to the multiplication action by ρ on $Z/3[\rho]$, i.e.,

$$\alpha(x, y) \rightarrow (2y, x)$$

(2) $$\rho(x + \rho y) \rightarrow \rho x + \rho^2(y) = 2y + \rho(x) .$$

Then the redundancy in X-ray data of $p3$ symmetry is

(3) $$f(x, y) = f(2y, x) .$$

Following the example of section 5.4, we have

(4) $$W(3) = \left[\omega^{\phi(\gamma^{j+k})} \right]_{0 \leq j,k < 8} ,$$

Let the ordering of j and k are

$$j = 0, 1, 2, 3, 4, 5, 6, 7;$$

$$k = 0, 1, 2, 3, 4, 5, 6, 7;$$

By the $p3$ symmetry we can reduce the input data to an asymmetric unit X_s only including $X(0)$ and X_1. The output has the same structure as input. Thus the size of input data and output data reduce from 9 to 3. We have

(5) $$U_s = \begin{bmatrix} 1 & 4 & 4 \\ 4 & 1 & -2 \\ 4 & -2 & 1 \end{bmatrix} .$$

Variant 2

In variant 2 we will choice different index ordering on $Z/P \times Z/P$. To find circulant structure we order j and k of (4) as follows:

$$j = 0,1,2,3,4,5,6,7;$$

$$k = 7,6,5,4,3,2,1,0;$$

7.5 Summary

In this chapter we have described the multiplicative two-dimensional prime point FFT algorithm for $p4$ symmetry. The computation of Fourier transform with $p4$ symmetry has been changed to asymmetry unit with only one-fourth size. Since the data has been compressed to one-fourth size, it is more efficient that we only need compute the $\frac{P^2-1}{4} \times \frac{P^2-1}{4}$ circulant or skew-circulant matrix of real number instead of the original $P^2 \times P^2$ two-dimensional Fourier transform matrix of complex number.

Chapter 8

Multidimensional FFT Algorithm for $N = P_1P_2$ with $p3$ and $p4$ Symmetry

8.1 Introduction

We have developed the multiplicative two-dimensional FFT algorithm of the prime number for $p3$ and $p4$ symmetry and the composite number $N = 3L$, $(3, L) = 1$, for $p3$ symmetry in the last two chapters. As an important application we will describe Orbit Exchange method(OEX) in this chapter to present $N = P_1P_2$ two-dimensional FFT algorithm with $p3$ and $p4$ symmetry, which will use the above multiplicative two-dimensional prime point algorithm as routines, including both non-symmetry and symmetry cases that we have developed in this thesis. Orbit Exchange(OEX) is a method for determining asymmetric units and mappings between them. The method is based on the group theoretic properties of the crystallographic groups and the ring structure of the sampling lattices. Asymmetric units determined by this method

are algebraic rather than geometric, and an interface between the two is part of the method. For a given group it is easy to find several asymmetric units.

Based on M. An's work of Orbit Exchange method [18], the methods for $p3$ and $p4$ symmetry cases are developed further. Orbit Exchange method generates a symmetrized P_1P_2-point Fourier transform algorithm, for $(P_1, P_2) = 1$, from P_1-point and P_2-point Fourier transform algorithms. This method decomposes the problem into small modules that are independent of each other, which can be parallelized. These modules also can be shared by other problems: Problems involving different crystallographic groups and different sampling grids. So the efficient subroutines developed in this thesis using multiplicative structure can be incorporated to OEX method. Thus we will get more efficient $p3$ and $p4$ subroutines of P_1P_2 size which can be used further into more complex groups. The method presented here is only concerned with two-dimensional Fourier transform of $p3$ and $p4$ cases. However this theory also can be extended to three-dimensional Fourier transform of more complicated groups.

8.2 Theorem of Asymmetric Unit

Definition 1 *For two sets U and V, the Cartesian product of U and V, denoted by $U \times V$ is the following set of ordered pairs.*

$$U \times V = \{(u, v) \mid u \in U, v \in V\}.$$

Definition 2 *For natural numbers P_1 and P_2, $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$ is the cartesian product of $\mathcal{Z}/P_1$ and $\mathcal{Z}/P_2$ with the componentwise arithmetic modulo the respective natural numbers. We will use the elements of $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$ to label the points in two-dimensional unit cells.*

Definition 3 *For $u \in U$,*

$$G(u) = \{gu \mid g \in G\}$$

is called the G−orbit of u. It is the set of elements equivalent to u modulo G.

U is partitioned into distinct G−orbits. A subset of U consisting of one element from each G−orbit is called an asymmetric unit in U by the group G, denoted U/G.

Let $g_1, g_2, \cdots, g_k$ be the elements of G. Then

$$U = g_1(U/G) \cup g_2(U/G) \cup \cdots \cup g_k(U/G),$$

where

$$g_i(U/G) = \{g_i u \mid u \in U/G\}.$$

Definition 4 *For $u \in U$, the isotropy subgroup at u, denoted by $Iso(u)$, is the subgroup of G that fixes u, i.e.,*

$$Iso(u) = \{g \in G \mid gu = u\}.$$

Definition 5 *Let D be a group acting on $U \times V$. D is said to act diagonally on $U \times V$ if*

$$d(u, v) = (d_U(u), d_V(v)),\ d \in D,\ (u, v) \in U \times V,$$

where d_U and d_V are actions on U and V respectively.

Now we present the theorem of how to find out the asymmetric unit of a given symmetric group.

Theorem 1 *Let D be a diagonal group acting on $U \times V$ and $\{v_1, v_2, \ldots, v_k\}$ be a V/D_V. Then*

(1) $$\cup_{i=1}^{k} (U \times \{v_i\})/Iso(v_i)$$

is an $(U \times V)/D$. We also can interchange the roles of U and V, we find another $(U \times V)/D$,

(2) $$\cup_{i=1}^{h} (\{u_i\} \times V)/Iso(u_i),$$

where $\{u_1, u_2, \ldots, u_h\}$ is an U/D_U.

Asymmetric unit is not unique in general. However, the isotropy property is uniquely determined by the set and the group that acts on the set. In particular, it is independant of the choice of asymmetric units.

To determine asymmetric units by nondiagonal groups, we first diagonalize the group, then proceed as in the case of diagonal groups. We will illustrate this later.

8.3 Diagonalizing Two-dimensional Ratation Groups

In this section we will describe a procedure for diagonalizing crystallographic rotation groups like $p3$ and $p4$ which do not act diagonally. Before this we would like go over the Chinese Remainder Theorem which We have described in chapter 6 already. It will be also important to develop Orbit Exchange method we are going to present.

Theorem 1 *Chinese remainder theorem(CRT)*

Let $N = P_1 P_2$, with $(P_1, P_2) = 1$, then there exists a ring-isomorphism

(1) $$\mathcal{Z}/N \cong \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \,.$$

We will construct the ring-isomorphism using idempotents. Since $(P_1, P_2) = 1$, there exists integer C_1 and C_2 satisfy

$$C_1P_1 + C_2P_2 = 1 \tag{2}$$

Now we define

$$e_1 \equiv C_2P_2 \mod N , \tag{3}$$

$$e_2 \equiv C_1P_1 \mod N . \tag{4}$$

Rewrite (6) as

$$e_1 = C_2P_2 + NM , \quad M \in \mathcal{Z} . \tag{5}$$

We can find out from (2) and (5) that

$$e_1 \equiv 1 \mod P_1 , \quad e_1 \equiv 0 \mod P_2 . \tag{6}$$

Similarly we can find out continuely that

$$e_2 \equiv 1 \mod P_2 , \quad e_2 \equiv 0 \mod P_1 . \tag{7}$$

The idempotents e_1 and e_2 are uniquely determined by conditions (6) and (7). The set

$$\{e_1, e_2\} \tag{8}$$

is called the system of idempotents corresponding to the factorization $N = P_1P_2$, where $(P_1, P_2) = 1$.

Besides (6) and (7), e_1 and e_2 have the following more properties:

$$e_1e_2 \equiv 0 \mod N . \tag{9}$$

$$e_1 + e_2 \equiv 1 \mod N . \tag{10}$$

$$e_1^2 \equiv e_1 \mod N , \quad e_2^2 \equiv e_2 \mod N . \tag{11}$$

These properties above uniquely determine e_1 and e_2 in $\mathcal{Z}/N$. Hence we can find f_1 and f_2 in $\mathcal{Z}/N$ with

$$e_1 = f_2 P_2 \in \mathcal{Z}/N\,, \quad e_2 = f_1 P_1 \in \mathcal{Z}/N\,. \tag{12}$$

where $(f_2, P_1) = 1$ and $(f_1, P_2) = 1$.

We now show that the existence of system of idempotents gives rise to a way of identifying $\mathcal{Z}/N$ and $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$. Define the following mapping

$$\phi : \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \to \mathcal{Z}/N\,, \tag{13}$$

by the formula

$$\phi(u_1, u_2) = (u_1 e_1 + u_2 e_2) \ mod\ N\,, \tag{14}$$

where $u_1 \in \mathcal{Z}/P_1, u_2 \in \mathcal{Z}/P_2$, the set $\{e_1, e_2\}$ is the system of idempotents corresponding to the factorization $N = P_1 P_2$.

The mapping ϕ preserves the arithmetic structure of rings and is a ring homomorphism.

This means that every ordered pair (u_1, u_2), $u_1 \in \mathcal{Z}/P_1, u_2 \in \mathcal{Z}/P_2$ can be written uniquely as

$$u_1 e_1 + u_2 e_2 \equiv u \quad mod\ N\,. \tag{15}$$

From the above description we see that the inverse mapping ϕ^{-1} is given by the following way,

$$\phi^{-1} : \mathcal{Z}/N \to \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \tag{16}$$

and

$$\phi^{-1}(u) = (u\ mod\ P_1, u\ mod\ P_2)\,, \quad u \in \mathcal{Z}/N\,. \tag{17}$$

which identifies that every element $u \in \mathcal{Z}/N$ can be identifyed uniquely with the ordered pair $(u\ mod\ P_1, u\ mod\ P_2)$ in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$.

Concisely, by the Chinese remainder theorem presented above, for $N = P_1P_2$ and $(P_1P_2) = 1$, we have defined the following mappings

$$\phi : \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \to \mathcal{Z}/N \,, \tag{18}$$

by the formula

$$\phi(u_1, u_2) \;=\; (u_1e_1 \;+\; u_2e_2) \;\; mod\, N \,,$$

where $u_1 \in \mathcal{Z}/P_1$ and $u_2 \in \mathcal{Z}/P_2$, the $\{e_1, e_2\}$ is the system of idempotents corresponding to the factorization $N = P_1P_2$. And the inverse mapping ϕ^{-1} is given by

$$\phi^{-1} : \mathcal{Z}/N \to \mathcal{Z}/P_1 \times \mathcal{Z}/P_2 \,, \tag{19}$$

that is

$$\phi^{-1}(u) \;=\; (u \;mod\, P_1,\; u \;mod\, P_2) \,, \quad u \in \mathcal{Z}/N \,.$$

which identifies that every element $u \in \mathcal{Z}/N$ can be identifyed uniquely with the ordered pair $(u \;\; mod\, P_1, u \;\; mod\, P_2)$ in $\mathcal{Z}/P_1 \times \mathcal{Z}/P_2$.

Furthmore, denote $\mathcal{Z}/N \times \mathcal{Z}/N$ by $(\mathcal{Z}/N)^2$. The mapping ϕ and ϕ^{-1} can be extended to

$$\psi : (\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2 \to (\mathcal{Z}/N)^2 \,, \tag{20}$$

$$\psi((u_1, v_1), (u_2, v_2)) \;=\; (\phi(u_1, u_2), \phi(v_1, v_2)) \,.$$

and

$$\psi^{-1} : (\mathcal{Z}/N)^2 \to (\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2 \,, \tag{21}$$

$$\psi^{-1}(u, v) \;=\; (\phi^{-1}(u), \phi^{-1}(v)) \,,$$

We now describe a procedure for diagonalizing crystallographic groups that do not act diagonally. The procedure is to rewrite the carterian product $\mathcal{Z}/N \times \mathcal{Z}/N$ so as to diagonalize the group action.

For a group G acting on $(\mathcal{Z}/N)^2$, we have a diagonal group (G_1, G_2) acting on $(\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2$ with the following bijective correspondance :

$$g \in G \longleftrightarrow (g_1, g_2) \in (G_1, G_2),$$

$$\psi\left((g_1, g_2)((u_1, v_1), (u_2, v_2))\right) = g\left(\psi((u_1, v_1), (u_2, v_2))\right) , \tag{22}$$

$$\psi^{-1}\left(g(u, v)\right) = (g_1, g_2)\left(\psi^{-1}(u, v)\right) . \tag{23}$$

A rotational symmetry is described by

$$\alpha_{11}u_1 + \alpha_{12}u_2,\ \alpha_{21}u_1 + \alpha_{22}u_2; \tag{24}$$

with $\alpha_{11}, \alpha_{12}, \alpha_{21}$ and α_{22} having values 0 or ± 1. We introduce another way of describing the above for ease of discussion. Denote by $\underline{u}$ the column matrix $\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$. Let $\alpha = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix}$. The matrix multiplication $\alpha\underline{u}$ describes in (24) via the correspondence

$$(u_1, u_2) \longleftrightarrow \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

for

$$\alpha\underline{u} = \begin{bmatrix} \alpha_{11}u_1 + \alpha_{12}u_2 \\ \alpha_{21}u_1 + \alpha_{22}u_2 \end{bmatrix}.$$

Henceforth, by abuse of notation, we will identify elements of $(\mathcal{Z}/N)^2$ with column matrices. We will identify a group G of rotations with a group of 2×2 matrices. The action of G on $(\mathcal{Z}/N)^2$ is the matrix multiplication modulo N. In this setting, the actions of diagonl components G_1 and G_2 are the matrix multiplications modulo P_1 and P_2 respectively.

Example 1

Consider the group $p3$ acting on $(\mathcal{Z}/15)^2$.

$$p3 = \{u, v;\ \overline{v}, u-v;\ v-u, \overline{u}.\} \longleftrightarrow \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & 1 \\ -1 & 0 \end{bmatrix}. \right\}$$

We can decompose $(\mathcal{Z}/15)^2$ into $(\mathcal{Z}/3)^2$ and $(\mathcal{Z}/5)^2$ by the map ψ^{-1} above. The system of idempotents $\{e_1, e_2\}$ of the mapping ψ corresponding to the factorization $15 = 3 \times 5$ are $e_1 = 10$ and $e_2 = 6$ (see the table 1 of section 6.2). Take the element (8,2) $\in (\mathcal{Z}/15)^2$.

(1) $\psi^{-1}(8,2) = ((2,2),(3,2)),\quad (2,2) \in (\mathcal{Z}/3)^2,\quad (3,2) \in (\mathcal{Z}/5)^2.$

(2) $p3$ identifies (2,2) with (1,0) and (0,1) in $(\mathcal{Z}/3)^2$.

(3) $p3$ identifies (3,2) with (3,1) and (4,2) in $(\mathcal{Z}/5)^2$.

(4)

$$\psi((2,2),(3,2)) = (10 \times 2 + 6 \times 3, 10 \times 2 + 6 \times 2)\ mod\ 15 = (8,2),$$

$$\psi((1,0),(3,1)) = (10 \times 1 + 6 \times 3, 10 \times 0 + 6 \times 1)\ mod\ 15 = (13,6),$$

$$\psi((0,1),(4,2)) = (10 \times 0 + 6 \times 4, 10 \times 1 + 6 \times 2)\ mod\ 15 = (9,7),$$

Note that $p3$ identifies (8,2) with (13,6) and (9,7) in $(\mathcal{Z}/15)^2$.

Example 2

Consider the symmetric group $p4$ acting on $(\mathcal{Z}/21)^2$.

$$p4 = \{u, v; -v, u; -u, -v; v, -u\}$$
$$\longleftrightarrow \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \right\}$$

We can decompose $(\mathcal{Z}/21)^2$ into $(\mathcal{Z}/3)^2$ and $(\mathcal{Z}/7)^2$ by the mapping ψ^{-1} also. The system of idempotents $\{e_1, e_2\}$ of the mapping ψ corresponding to the factorization $21 = 3 \times 7$ are $e_1 = 7$ and $e_2 15$ (see table 1 of section 6.2). We take the element $(5,9) \in (\mathcal{Z}/21)^2$.

(1) $\psi^{-1}(5,9) = ((2,0),(5,2)), \quad (2,0) \in (\mathcal{Z}/3)^2\ ,\ (5,2) \in (\mathcal{Z}/7)^2\ .$

(2) $p4$ identifies $(2,0)$ with $(0,2)$, $(1,0)$ and $(0,1)$ in $(\mathcal{Z}/3^2$.

(3) $p4$ identifies $(5,2)$ with $(5,5)$, $(2,5)$ and $(2,2)$ in $(\mathcal{Z}/7)^2$.

(4)

$$\psi((2,0),(5,2)) = (7 \times 2 + 15 \times 5, 7 \times 0 + 15 \times 2) \ \ mod\, 21 = (5,9)\ ,$$

$$\psi((0,2),(5,5)) = (7 \times 0 + 15 \times 5, 7 \times 2 + 15 \times 5) \ \ mod\, 21 = (12,5)\ ,$$

$$\psi((1,0)(2,5)) = (7 \times 1 + 15 \times 2, 7 \times 0 + 15 \times 5) \ \ mod\, 21 = (16,12)\ ,$$

$$\psi((0,1)(2,2)) = (7 \times 0 + 15 \times 2, 7 \times 1 + 15 \times 2) \ \ mod\, 21 = (9,16)\ .$$

Note that $p4$ identifies $(5,9)$ with $(12,5)$, $(16,12)$ and $(9,16)$.

Formulas (22) and (23) describe a way of determining the action of the group G on $(\mathcal{Z}/N)^2$ in terms of (G_1, G_2) acting on $(\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2$ via the maps ψ and ψ^{-1}.

By the theorem 1 of section 9.2, we know how to find out the asymmetric unit of a symmetric group.

To determine asymmetric units of nondiagonal groups like $p3$ and $p4$ rotation groups, we first diagonalize the group, then proceed as in the case of diagonal groups. We illustrate this with the following example.

Example 3

Determination of $(\mathcal{Z}/15)^2/p3$.

(1) Reindex the elements of $(\mathcal{Z}/15)^2$ with elements in $(\mathcal{Z}/3)^2 \times (\mathcal{Z}/5)^2$ via ψ^{-1}.

(2) $p3$–orbit decompositions of $(\mathcal{Z}/3)^2$and $(\mathcal{Z}/5)^2$ are given in tables 1 and 2.

(3) Selecting one element from each row of the tables, we find asymmetric units in $(\mathcal{Z}/3)^2$ and $(\mathcal{Z}/5)^2$. To this end, we will choose the first columns in each case.

$$A_3 = \{(0,0),(0,1),(0,2),(1,2),(2,1)\}$$

is a $(\mathcal{Z}/3)^2/p3$.

$$Iso(\{(0,0),(1,2),(2,1)\}) = p3, \quad Iso(\{(0,1),(0,2)\}) = p1.$$

$$A_5 = \{(0,0),(0,1),(0,2),(0,3),(0,4),(1,2),(1,3),(2,1),(3,1)\}$$

is a $(\mathcal{Z}/5)^2/p3$.

$$Iso(0,0) = p3, \quad Iso(u_5,v_5) = p1, \text{for } (u_5,v_5) \neq (0,0).$$

(4) Applying formulas (4) and (5), we find asymmetric units in the product $(\mathcal{Z}/3)^2 \times (\mathcal{Z}/5)^2$.

$$A_{3,5} = (A_3 \times \{(0,0)\}) \cup ((\mathcal{Z}/3)^2 \times (A_5 - \{(0,0)\})$$

$$A_{5,3} = (\{(0,0),(1,2),(2,1)\} \times A_5) \cup (\{(0,1),(0,2)\} \times (\mathcal{Z}/5)^2)$$

are $((\mathcal{Z}/3)^2 \times (\mathcal{Z}/5)^2)/(p3,p3)$.

(5) $j^{-1}(A_{3,5})$ as well as $j^{-1}(A_{5,3})$ are $(\mathcal{Z}/15)^2/p3$.

A_3	αA_3	$\alpha^2 A_3$
(0,0)	(0,0)	(0,0)
(0,1)	(2,2)	(1,0)
(0,2)	(1,1)	(2,0)
(1,2)	(1,2)	(1,2)
(2,1)	(2,1)	(2,1)

Table 1 $p3$-orbit decomposition of $(\mathcal{Z}/3)^2$.

A_5	αA_5	$\alpha^2 A_5$
(0,0)	(0,0)	(0,0)
(0,1)	(4,4)	(1,0)
(0,2)	(3,3)	(2,0)
(0,3)	(2,2)	(3,0)
(0,4)	(1,1)	(4,0)
(1,2)	(3,4)	(1,4)
(1,3)	(2,3)	(2,4)
(2,1)	(4,1)	(4,3)
(3,1)	(4,2)	(3,2)

Table 2 $p3$-orbit decomposition of $(\mathcal{Z}/5)^2$.

Chinese remainder theorem extends to any finite number of factors that are relatively prime to each other. Thus the above method can be iterated.

The applications of the theorem of astmmetric unit we show are all examples of algorithms and implementation ideas. The objective in designing an algorithm for a given symmetry group is to reduce the size of data set by approximately the order of the group. As we will see, computationally efficient

data set is not a precise asymmetric unit. In most cases, we will use a set that contains an asymmetric unit. Theses aspects will be pointed out in the examples that follow.

8.4 Orbit Exchange for FFT with Rotation Symmetries

We have presented the prime factor method in section 6.3.2 already. It provides a way using Chinese remainder theorem to replace a large size one-dimensional Fourier transform computation with a small size two-dimensional computation.

By the Chinese remainder theorem, for $N = P_1P_2$ and $(P_1P_2) = 1$, we have defined the following mappings ϕ, ϕ^{-1}, ψ and ψ^{-1} in (18) to (21) of section 9.3.

Let $f(u)$ be a function on $\mathcal{Z}/N$. The one-dimensional Fourier transform is defined as

$$(1) \qquad \hat{f}(v) = \sum_{u \in \mathcal{Z}/N} f(u) e^{\frac{-2\pi i}{N} uv} .$$

For $u, v \in \mathcal{Z}/N$, we can use the system of idempotents $\{e_1, e_2\}$ to change u and v by

$$(2) \qquad u = u_1e_1 + u_2e_2 \ mod\, N ,$$

and

$$(3) \qquad v = v_1e_1 + v_2e_2 \ mod\, N .$$

Then

$$(4)\ \hat{f}(v_1e_1 + v_2e_2) = \sum_{u_1 \in \mathcal{Z}/N, u_2 \in \mathcal{Z}/N} f(u_1e_1 + u_2e_2) e^{\frac{-2\pi i}{N}(u_1e_1+u_2e_2)(v_1e_1+v_2e_2)} ,$$

By the properties of the system of idempotents $\{e_1, e_2\}$,

$$e^{\frac{-2\pi i}{N}(u_1e_1+u_2e_2)(v_1e_1+v_2e_2)} = e^{\frac{-2\pi i}{N}u_1v_1e_1}e^{\frac{-2\pi i}{N}u_2v_2e_2} .$$

by (15) of section 8.2, $e_1 = f_2P_2$ and $e_2 = f_1P_1$, the above equality can be changed to

$$e^{\frac{-2\pi i}{P_1}u_1v_1f_2}e^{\frac{-2\pi i}{P_2}u_2v_2f_1}$$

(4) can be rewritten as

$$\hat{f}(v_1e_1 + v_2e_2) = \sum_{u_1\in Z/P_1 u_2\in Z/P_2} f(u_1e_1 + u_2e_2)e^{\frac{-2\pi i}{P_1}u_1v_1f_2}e^{\frac{-2\pi i}{P_2}u_2v_2f_1} .$$

This method is known as Prime Factor algorithm, which also can be applied to the computation of multidimensional Fourier transform as well.

We now describe the orbit exchange method for incorporating symmetry groups to the Fourier transform algorithms. The main idea is to exploit the existence of several stages of computation and data transposition. Symmetry is incorporated into the data transposition only; Using the symmetry, data is mapped from one asymmetric unit onto another asymmetric unit. In each stage, an asymmetric unit is chosen in a way to make the computation using existing Fourier transform routines of our algorithms in this thesis.

We will describe the procedures for a nondiagonal rotation symmetric group to the prime factor method with examples.

We begin by listing the property of the Fourier transform in the presence of rotation symmetries.

By the section 7.2 we have shown the property of Fourier transform in the presence of rotation symmetries and we have the following theorem

Theorem 1 *Denote the Fourier transform of f by $\hat{f}$. For a function f defined on $Z/N \times Z/N$, if $f(u,v) = f(\alpha(u,v))$, then*

(5) $$\hat{f}(w,z) = \hat{f}(\alpha^{\#}(w,z)),$$

where $\alpha^{\#}$ denotes the inverse transpose of the matrix α, $\alpha^{\#} = (\alpha^{-1})^t$.

Let g be a function on $(\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2$ and define for $(\underline{w},\underline{v}),(\underline{w},\underline{z}) \in (\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2$

$$g_1(\underline{w},\underline{v}) = \sum_{\underline{u}} g(\underline{u},\underline{v}) e^{-\frac{2\pi i}{P_1}<\underline{u},\underline{w}>},$$

$$g_2(\underline{w},\underline{z}) = \sum_{\underline{v}} g_1(\underline{w},\underline{v}) e^{-\frac{2\pi i}{P_2}<\underline{v},\underline{z}>}.$$

Thus g_2 is the Fourier transform of g. Assume now that g is (G_1,G_2) symmetric, where (G_1,G_2) is the diagonalized group of rotation symmetries. The theorem below describes the intermediate symmetries.

Theorem 2 *If g is (G_1,G_2) symmetric, then for $(\alpha_1,\alpha_2) \in (G_1,G_2)$,*

$$g_1(\underline{w},\underline{v}) = g_1(\alpha_1^*\underline{w}, \alpha_2\underline{v}),$$

$$g_2(\underline{w},\underline{z}) = g_2(\alpha_1^*\underline{w}, \alpha_2^*\underline{v}).$$

Example 4

Intermediate symmetries of the diagonalized $p3$.

$$p3 = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & 1 \\ -1 & 0 \end{bmatrix}. \right\},$$

$$p3^* = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}. \right\}.$$

$$
\begin{aligned}
g(u_1,u_2,v_1,v_2) &= g(P_1-u_2,u_1-u_2,P_2-v_2,v_1-v_2) \\
&= g(u_2-u_1,P_1-u_1,v_2-v_1,P_2-v_1) \qquad (6) \\
g_1(w_1,w_2,v_1,v_2) &= g_1(P_1-w_1-w_2,w_1,P_2-v_2,v_1-v_2) \\
&= g_1(w_2,P_1-w_1-w_2,v_2-v_1,P_2-v_1) \qquad (7) \\
g_2(w_1,w_2,z_1,z_2) &= g_2(P_1-w_1-w_2,w_1,P_2-z_1-z_2,z_1) \\
&= g_2(w_2,P_1-w_1-w_2,z_2,P_2-z_1-z_2) \qquad (8)
\end{aligned}
$$

Now we describe Orbit Exchange incorporating the prime factor method to compute FFT with rotation symmetries. We will use $(\mathcal{Z}/N)^2$ to index the data set. Assume $N = P_1P_2$, with $(P_1,P_2) = 1$. Let G be a group of rotation symmetries and let data be $G-$symmetric.

1. Reindex the data to be on $(\mathcal{Z}/P_1)^2 \times (\mathcal{Z}/P_2)^2$. Set $U = (\mathcal{Z}/P_1)^2$, $V = (\mathcal{Z}/P_2)^2$. We will denote the elements of $U \times V$ using the column matrix notation. Denote the diagonalized group action by (G_U, G_V).

2. Using the diagonal action, determine V/G_V.

3. With data on $U \times V/G_V$, compute $F_2(P_1)$ for each $v \in V/G_V$.

4. Determine U/G_U^*.

5. Map the partially transformed data from step 3 onto $V \times U/G_U^*$.

6. With the data on $V \times U/G_U^*$ compute $F_2(P_2)$ for each $u \in U/G_U^*$.

7. Reindex the data on $V \times U/G_U^*$ to be on an asymmetric unit in $(\mathcal{Z}/N)^2$ by the ψ map.

Example 5

FFT Computation Incorporating $p3$

We will use $(\mathcal{Z}/12)^2$ as indexing the data set with $p3$ symmetry.

1. Decompose $(\mathcal{Z}/12)^2$ into $(\mathcal{Z}/4)^2 \times (\mathcal{Z}/3)^2$. Set $U = (\mathcal{Z}/4)^2$ and $V = (\mathcal{Z}/3)^2$.

2. The asymmetric units in the product $(\mathcal{Z}/4)^2 \times \mathcal{Z}/3)^2$ is

$$A_{4,3} = (A_4 \times (A_3 - \{(1,0),(2,0)\})) \cup ((\mathcal{Z}/4)^2 \times \{(1,0),(2,0)\}),$$

   with the set A_3 in Table 1 of section 8.3 as $V/p3$.

3. Compute $p3$ symmetry $F_{2s}(4)$ three times and non-symmetry $F_2(4)$ two times of the data on $A_{4,3}$.

4. Determine $U/p3^*$.

   $p3^*$orbit decomposition of U is given in table 1. The elements in the first column provides an $U/p3^*$.

$$A_4 = \{(0,0),(1,0),(2,0),(3,0),(1,1),(3,2)\}.$$

5. Map $U \times V/p3 \longrightarrow V \times U/p3^*$.

6. Compute $p3$ symmetry $F_{2s}(3)$ one time and non-symmetry $F_2(3)$ eight times of data on $(A_3 \times \{(0,0)\}) \cup ((\mathcal{Z}/3)^2 \times (A_4^* - \{(0,0)\}))$.

7. Reindex $V \times A_4^*$ to obtain the tranformed data indexed by elements in $(\mathcal{Z}/12)^2$.

A_4	α^*	$(\alpha^2)^* A_4$
(0,0)	(0,0)	(0,0)
(1,0)	(3,1)	(0,3)
(2,0)	(2,2)	(0,2)
(3,0)	(1,3)	(0,1)
(1,1)	(2,1)	(1,2)
(3,2)	(3,3)	(2,3)

Table 1 $p3^*$-orbit decomposition of $(\mathcal{Z}/4)^2$.

8.5 Summary

In this chapter we have described the Orbit Exchange method for the composite number $N = P_1 P_2$ with $(P_1, P_2) = 1$ for $p3$ and $p4$ symmetry. Orbit Exchange method is a method for determining asymmetric units and mapping between them. This method decomposes the problem into small modules that are independent of each other, which can be parallelized. Hence the new multiplicative algorithm developed in this thesis, especially for $p3$ and $p4$ symmetry subroutines, can be nested into OEX method efficiently. We only concern with two-dimensional Fourier transform of the $p3$ and $p4$ cases in this chapter, however the Orbit Exchange method is powerful to be extended to three-dimensional Fourier transform of more complicated crystallographic symmetry groups.

Chapter 9

Implementation

9.1 Introduction

The multiplicative two-dimensional fast Fourier transform algorithms developed in this thesis can be implemented efficiently in different computer architectures depending on different variants of our algorithm. The performance of the implementation of the algorithms will be effected by the different machine architectures and parameters. In fact, an efficient algorithm will have a good performance based on that we have a better knowledge of the machine architecture and the effective use of programming language.

Computers may be categorized according to whether they have one or many instruction streams and one or many data streams. SISD defines the simplest conventional computers, and the SIMD and MIMD contain all supercomputers.

- SISD describes the relatively simple sequential computers that perform each instrution of a program to completion before beginning the next instruction. There is no possibility of overlap within the machine, and therefore only one stream of data through the CPU.

- SIMD defines a computer system having a single instruction processor and multiple arithmetic and logical processors, thereby allowing simultaneous computation to be performed on different streams of data.

- MIMD has multiple instruction processors as well as a means to overlap execution of instructions. A more exciting and more complicated application of such systems is to assign several CPUs to execute the instructions of a single program. Here it is again necessary to cause each CPU to work on different segments of the program data, the MIMD CPUs each perform a unique version of the instruction stream independent of the others. At critical points in the program, the CPUs must be forced to synchronize with one another, either to properly pass information among themselves or to correctly share a common memory location.

SISD is the sequential machine such as the VAX/780 and the Micro VAX II. The following is a derivation of the performance of an algorithm working on SISD machine relying on the arithmetic account. Let

T1 = the time required to perform an floating-point operation of addition,

N_a = the number of additions,

N_m = the number of multiplications,

k = the ratio of the time spent by an floating-point operation of multiplication to addition, then the total time T to perform the total operational count is

$$T = T1(N_a + kN_m)\,. \tag{1}$$

Since in the sequential machine the time of float-point multiplication is more large than addition, the main point for a good performance on the sequential computer is to minimize the number of multiplications.

But (1) can not be used to evaluate the performance of an algorithm on SIMD and MIMD supercomputer architectures on which we are more interesting in implementation of our algorithms in this thesis. For example, Cray-1 and Cray X-MP computers have eight vector registers, each having 64 elements that are 64 bits wide. When a loop of arbitrary length is "vectorized", it is done in vector strips of length 64. Consider the loop:

```
      DO  10   I = 1,  N
      A(I) = B(I) + R* C(I)
10    CONTINUE
```

In effect, this is performed on a CRAY in the following way:

```
      J = MOD(N, 64)
      DO  20   I = 1, J
      A(I) = B(I) + R* C(I)
20    CONTINUE
      I = J
      DO  30   K = J+1, N, 64
      DO  30   COUNT = 1, 64
      I = I + 1
      A(I) = B(I) +  R * C(I)
30    CONTINUE
```

If N is not evenly divisible by 64, loop 20 does the "remainder", or else it is not executed(J=0). Loop 30 then performs a series of loops, each of exactly 64 in length to complete the computation. This is known as vector pipelining. The number of elements in a vector register obviously determines the length of a "strip". The IBM 3090 Vector Facility has vector registers of 128 elements.

Another technique for vector porcessing is vector chaining. The vector functional units can be "chained" together, thus allowing overlap of related operations. For example the multiply-add function as in the above program can be chained. It means that the add and multiply pipelines can be linked together, one producing a result fed directly into the other-thus again doubling the result rate. Generally there are other combinations of operations which can be chained.

By the above example we know that in order to obtain the good performance on vector processor, it is different from the conventional machine. We can not use arithmetric account simply. In order to obtain good performance on supercomputer it is more complicate than on the sequential computer. We should use the architecture of the vector pipelining and vector chaining as more as possible. Since vector processing is inherently fast than scalar, Amdahl's law tells us that the system will be dominated by scalar performance. Amdahl's law provides performance as a function of the fraction of operations vectorized and allows us to dertermine for an existing program how much code must be vectorized to achieve performance goals.

Here we would introduce some features about supercomputer which will mainly effect the performance of implementation.

9.2 Supercomputer Architecture

9.2.1 Introduction

To achieve optimal performance on a supercomputer, it is essential for us to understand the architecture of the target machine. On the majority of commercially successful supercomputers there are fast registers, a large banked memory, and segmented functional units.

A supercomputer is the biggest, fastest computer available at the moment.

What distinguishes the supercomputers from others is their ability to perform many operations simultaneously.

Some supercomputers accomplish many simultaneous operations by "vector"processing [25], that is, by using powerful instructions to feed arrays of operands through a "pipeline" This pipeline concept is a recognition that the most intense use of a computer is almost always in a loop, doing the same operations to many different operands. The Cray X-MP, Cray-2, the IBM 3090 Vector Facility are examples of pipelined vector processors.

Other supercomputers accomplish many simultaneous operations by having many processsors working in parallel on a program. Some machines combine both parallel and vector architectures, such as the Cray X-MP, the IBM 3090/600F.

All of the supercomputers are characterized by their ability to perform much faster in "vector" or "parallel" mode than in "scalar" mode. Pipelining of operations and simultaneous execution of instructions are the mainstay of supercomputers. The performance might be from two to one thousand times faster, but only you have efficient algorithms and the effective use of languages, such as Fortran, on supercomputers.

9.2.2 some features of supercomputer

Functional units and segmented functional units

A computer has hundreds of instructions. It might be partitioned into functional units, each one of which executes a family of related instructions. Because each of the major functions of the CPU has been realized in a wholly independent unit of hardware, compilers take advantage of multiple functional units by attempting to schedule as many independent operations as possible to achieve maximum overlap of instruction execution.

A computer might be able to issue a new instruction in each clock cycle, but there are very few instructions that complete execution in just one clock cycle. For example, a floating-point add instruction might take four clock cycles to complete; a multiply or a divide might take even longer. For this reason, each functional unit is itself further partitioned into a number of independent segments, preferably one segment for each clock cycle of execution. By this means, a computer may issue several identical instructions in sequence as long as the operands are independent. Consider the execution of the following Fortran statements:

$$Z1 = X1 + Y1$$
$$Z2 = X2 + Y2$$
$$Z3 = X3 + Y3$$
$$Z4 = X4 + Y4$$

Assuming that the operands have already been fetched to registers, then the steps through time shown in Figure 1 indicate the operation of the segmented floating-point add unit.

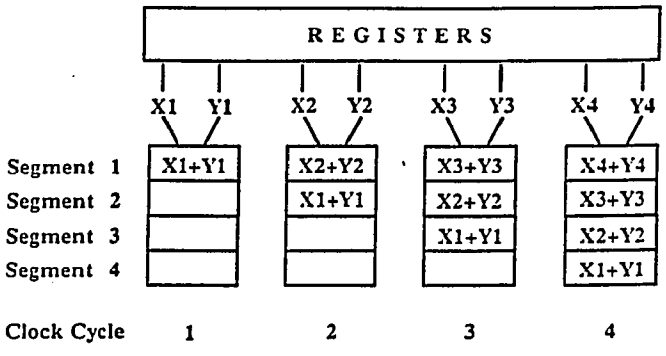

Fig. 1 Segmented Functional-Unit Operation

The adder depicted in Figure 1 performs as an assembly line with four stations. The segmentation of the adder generates one result per clock cycle instead of one result every four clock cycles achievable without segmentation.

The performance of a segmented functional unit is characterized by two features:

* Startup Time. This is the number of clock cycles prior to the generation of the first result.

* Result Rate. This is directly related to the longest segment(every segment may be one or several clock cycles long) in a functional unit. After the startup time, a functional unit can deliver one result each time the longest segment completes its task.

Memory Caches

Considering that most functional units can produce one result per clock cycle, and that several can run in parallel, a tremendous burden is placed on memory access to fetch operands and store results. One way to alleviate this burden is the use of memory banks.

A memory cache is a small, fast, expensive memory that placed between the very fast CPU registers and the large slow main memory of a machine. When the CPU requests a data item from memory, the hardware checks to see if the item is resident in the cache, and, if so, it delivers it to the CPU, typically in two clock cycles. If the item is not in the cache, the hardware requests a packet of data from main memory to the cache that includes the item requested plus(usually) three more items as well, on the assumption that the data will be referenced contiguously. Assuming that the transfer from memory to cache takes 12 clock cycles, then the time to transfer one item

from memory to CPU is: 12+2=14 clock cycles. If the other three items are subsequently referenced from the cache, then the total cost in time to transfer data to the CPU is: 20 clock cycles, or 5 clock cycles per item transferred.

Most cache systems also use the cache for instructions as well as data operands.

The Vector Processor

There are two major categories of vector processors; memory-to-memory machines and register-to-register machines. The memory-to-memory machines fetch vector operands directly from memory to the CPU and store vector results directly back into memory, with no intervening registers.

The register-to-register machines move data from memory to vector registers and perform computations with vector-register operands, placing results again into vector registers. These results are either retained for further use or stored back into memory. The first register-to-register vector processor was the Cray-1. In fact, all other vector processors except the CYBER 205 and ETA 10 are register-to-register machines.

The major characteristics affecting performance of programs on register-to-register machines are

* Clock cycle
* Instruction issue rate
* Size and number of vector registers
* Memory size
* Number of concurrent paths to memory
* Ability to fetch/store vectors with a stride
* Number of duplicate arithmetic functional units(multiple vector pipelines)

* Whether functional units can be "chained" together
* Indirect addressing capability
* Handling of conditional blocks of code

9.3 Implementation

By implementing the new multiplicative two-dimensional FFT algorithms of size P and $N = P_1P_2$ cases on representative machines, we are building some libraries as follows:

- The library for two-dimensional FFT
- The library for two-dimensional FFT with p3 symmetry
- The library for two-dimensional FFT with p4 symmetry

The following programming technique were used to optimize performance when we implement our algorithm on vector machine:

- The algorithms should be tailored to specific operational characteristics of the vector machine, such as cache size, vector section size, number of vector registers, and page size.
- Access data that are stored contiguously; that is, use stride-1 computetions.
- Reuse data in vector registers, minimizing vector loads and stores.
- Manage the cache efficiently to maximize data reuse; i.e., algorithms are structured to operate on subblocks that are sized to remain in the cache until all computations involving the subblock are complete. For example,

the number of rows in the subblock might be equal to the VSS, while the number of columns is chosen so that the subblock will fit in the cache.

- Use the most efficient machine instructions-for example, the multiply-add, multiply-subtract, and multiply-accumulate instructions (the compound vector instructions). Neglecting overhead, these instructions generate two floating-point results every cycle. Other vector instructions, such as multiply, add, and subtract, generate one floating-point result per cycle.

- Perform fewer loads and stores for short-precision data by using long-precision instructions.

- Use algorithms that minimize paging; for example, alternate forward and backward sweeps through the columns of a matrix.

The performance generally correspond to the programmer being able to take advantage of vector instructions, reuse of data in vector registers, and reuse of data in cache.

Some test programs of our multiplicative two-dimensional FFT algorithms have been implemented on IBM 3090 using vector facility with complex input data. The timing results shown in table 1 is as follows:

	ESSL SCFT2	Multiplicative Algorithm	Multi. Algo. $p3$ Symmetry	Multi. Algo. $p4$ Symmetry
3×3		0.023 ms	0.007 ms	0.004 ms
4×4		0.030 ms	0.008 ms	
5×5		0.171 ms	0.063 ms	
6×6	0.481 ms			
7×7		0.320 ms		0.064 ms
10×10	0.603 ms			
11×11		0.947 ms	0.234 ms	0.162 ms

Table 1. Timing of 2-dimensional FFT

There is the Engineering and Scientific Subroutine Library(ESSL) on IBM 3090. It is a set of high-performance mathematical subroutines. In order to prove the efficiency of our new algorithms, we list the timing of the subroutine SCFT2 of ESSL on IBM 3090 which computes the two-dimensional FFT with the input data of complex number. The minimum number of input data of SCFT2 is 6×6. In general case it is more hard to compute the prime point FFT than non-prime case. Since there is no prime number subroutines in ESSL, we only put the numbers which close to the prime numbers to compare each other.

We have done the prime cases of 5×5, 7×7 and 11×11 with our multiplicative two-dimensional FFT algorithm with field structure. We have also done 3×3 and 4×4 cases two-dimensional FFT using ring structure. For $p3$ symmetry case, we have done 3×3, 4×4, 5×5 and 11×11 cases. For $p4$ case, we have done 3×3, 7×7 and 11×11 cases of prime points. From

table 1, the timing of both $p3$ and $p4$ symmetry cases is good comparing with non-symmetry cases. It also looks good for timing comparing with ESSL subroutine.

	ESSL SCFT2	Multiplicative $p3$ Symmetry	OEX $p3$ Symmetry
12×12	0.651 ms	0.206 ms	0.216 ms

Table 2. Timing of 12×12 FFT

For composite number case, we have done 12×12 case with two methods. The first one is the multiplicative algorithm with ring structure shown in section 6.5. The second one is the OEX method described in chapter 8, which is nested with 3×3 and 4×4 multiplicative ring structure algorithm shown in section 6.5. Comparing with 12×12 of SCFT2 subroutine in ESSL, the timing is also good by incorporating the crystallographic symmetry to our efficient algorithms. Althrough we only show implementation results of some test programs, it has been shown the efficiency of our new multiplicative two-dimensional FFT algorithms developed in this thesis.

Reference

[1] R. E. Blabut, "Fast Algorithms for Digital Signal Processing", Addison-Wesley, Reading, Mass., 1985

[2] M. T. Heideman, D.H. Johnson and C.S. Burrus, "Gauss and the History of the Fast Fourier Transform", IEEE ASSP Maganize, October 1984, pp.14-21.

[3] J. Cooley and J. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", Math. Comput., Vol. 19, No. 2, pp.297-301, 1965.

[4] I. J. Good, "The Interaction Algorithm and Practical Fourier Analysis", J. Royal Statist. Soc., Ser. B 20, No. 2, (1958), pp.361-372; addendum, 22 (1960), pp.372-375.

[5] L. H. Thomas, "Using a Computer to Solve Problems in Physics", Applications of Digital Computers. Ginn and Co., Boston, Mass., 1963.

[6] D. P. Kolba and T.W. Parks, "Prime Factor FFT Algorithm Using High Speed Convolution", IEEE Trans. Acoust., Speech, Signal Proc., ASSP-25 (1977), pp.281-294.

[7] C. S. Burrus, and P. W. Eschenbacher, "An In-place, In-order Prime Factor FFT Algorithm", IEEE Trans. Acoust., Speech, Signal Proc., ASSP-29 (1981), pp.806-817.

[8] C. M. Rader, "Discrete Fourier Transforms When the Number of Data Samples is Prime", Proc. IEEE 56, 1968, pp.1107-1108.

[9] S. Winograd, "On Computing the Discrete Fourier Transform", Proc. Nat. Acad. Sci. USA, Vol 73. No. 4, April 1976, pp.1005-1006.

[10] S. Winograd, "On Computing the Discrete Fourier Transform", Math. of Computation, Vol. 32. No. 141, Jan. 1978, pp.175-199.

[11] S. Winograd, "Arithmetic Complexity of Computations", CBMS-NSF Conference Series in Applied Math. No. 33, SIAM, 1980.

[12] L. Auslander, E. Feig and S. Winograd, "New Algorithms for the Multidimensional Discrete Fourier Transform", IEEE Trans. Acoust., Speech, Signal Proc. Vol. ASSP-31, No. 2, April 1983, pp.388-403.

[13] R. C. Agarwal and J.W. Cooley, "New Algorithms for Digital Convolution", IEEE Trans. Acoust., Speech, Signal Proc. ASSP-25 (1977), pp.392-410.

[14] Norman F. M. Henry and K. Lonsdale, "International Tables for X-ray Crystallography", Vol. I, The Kynoch Press, Birmingham, England, 1952.

[15] L. Auslander and M. An, "Fourier Transforms that Respect Crystallographic Symmetries", IBM Journal of Research and Development, 31(2), pp.213-223, March 1987.

[16] L. F. TenEyck, "Crystallographic Fast Fourier Transforms", Acta. Cryst., pp.183-191, 1973.

[17] G. Bricogne and R. Tolimieri, "Symmetrized FFT Algorithms", IMA Springer-Verlag, to appear.

[18] M. An, J. Cooley and R. Tolimieri, "Factorization Method for Crystallographic Fourier Transforms", Advances in Applied Mathematics 11, pp.358-371(1990).

[19] K. Ireland and M. Rosen, "A Classical Introduction to Modern Number Theory", Springer-Verlag.

[20] R. Tolimieri, M. An and C. Lu, "Algorithms for discrete Fourier Transform and Convolution", Springer-Verlag, 1989.

[21] J. Johnson, R. Johnson, D. Rodriguez, R. Tolimieri, "A Methodology for Designing, Modifying, and Implementing Fourier Transform Algorithms on Various Architectures", IEEE Trans. on Circuits and Systems, In Press.

[22] R. C. Agarwal and C. S. Burrus, "Fast One-dimensional Digital Convolution by Multidimensional Techniques", IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-22, No. 1, PP.1-10, Feb. 1974.

[23] R. C. Agarwal and T. W. Cooley, "Vectorized Mixed Radix Discrete Fourier Transform Algorithms", Proceedings of the IEEE, Vol. 75, No. 9, Sep. 1987. pp.1283-1292.

[24] H. Tian, M. An and R. Tolimieri, "Multiplicative Algorithm of Computation Prime Factor FFT for P3 Symmetry", Proceedings of IEEE Fourth Digital Signal Processing Workshop, Sept. 1990.

[25] Hui Cheng, "Vector Pipelining, Chaining, and Speed on the IBM 3090 and Cray X-MP", Computer, IEEE, Sept. 1989. pp.31-46.