HOURGLASS MATRIX: ITS QUADRANT INTERLOCKING FACTORIZATION USING MODIFIED CRAMER’S RULE AND ITS MIXED GRAPH

BABARINSA OLAYIWOLA ISAAC

UNIVERSITI SAINS MALAYSIA

2019

HOURGLASS MATRIX: ITS QUADRANT INTERLOCKING FACTORIZATION USING MODIFIED CRAMER'S RULE AND ITS MIXED GRAPH

by

BABARINSA OLAYIWOLA ISSAC

Thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

June 2019

ACKNOWLEDGEMENT

Pure mathematics is, in its way, the poetry of logical ideas.

Albert Einstein

First, I give glory to Almighty God for His mercy and grace over the examiners, readers and author of this thesis. I would like to express my deepest gratitude to my supervisor, Professor Hailiza Kamarulhaili - Dean, School of Mathematical Sciences - for giving me the confidence, encouragement and continuous guidance during my research work and my writing up of this thesis. I am also grateful for the corrections and suggestions by Associate Professor Nader Jafari Rad on the graph theory aspect of my thesis, Professor Beata Bylina on the quadrant interlocking factorization, Mr. Arif Mandangan on the potential application of hourglass matrix in GGH encryption scheme, and my co-supervisor Dr. Maisarah Mohd of her thorough questioning.

The death of my parents during my Ph.D. program made this journey a long boring and lonely one, yet the quest for knowledge is an overwhelming task but at the end is peace and joy. The lamentation of my grandmother for not being educated, in spite her father who was a teacher care less for his children, motivated me to get all from education. Therefore, I dedicate my thesis to my grandmother, siblings and family.

I also appreciate the friends I met during my formal education for their financial support and encouragement. I do appreciate the advice of Professor Jerome Adepoju who was my undergraduate and postgraduate supervisor. Finally, I appreciate Universiti Sains Malaysia for appointing me as a USM's ambassador and as a Graduate Assistant at School of Mathematical sciences.

Babarinsa Olayiwola Isaac.

TABLE OF CONTENTS

CHAPTER 4 – HOURGLASS MATRIX AND ITS QUADRANT INTERLOCKING FACTORIZATION

CHAPTER 5 – MIXED HOURGLASS GRAPH AND ITS MIXED ENERGY

LIST OF TABLES

Page

LIST OF FIGURES

LIST OF ABBREVIATIONS

SIMD	Single-Instruction Multiple-Data
MIMD	Multiple-Instruction Multiple-Data
ABS	Abaffy-Broyden-Spedicato algorithms
BSP	Bulk Synchronous Parallel
QIF	Quadrant Interlocking Factorization
AQIF	Alternate Quadrant Interlocking Factorization
WZ	Matrices in W and Z forms
WH	Matrices in W and H forms
QZ	Matrices in Q and Z forms
QR	Matrices in Q and R forms
QW	Matrices in Q and W forms
RM	Real Multiplication
RA	Real Addition
AMD	Advanced Micro Devices
Intel	Integrated Electronics
IBM	International Business Machines
ABCI	AI Bridging Cloud Infrastructure
OpenMP	Open Multi-Processing
FLOPS	Floating Point Operations Per Second
PIE	Parallel Implicit Elimination
MATLAB	Matrix Laboratory
LU	Lower and Upper triangular matrices
GE	Gaussian Elimination

HMO	Hückel Molecular Orbital
Mod	Modulo
DFS	Depth-First Search
IFP	Integer Factorization Problem
DLP	Discrete Logarithm Problem
ECDLP	Elliptic Curve Discrete Logarithm Problem
RSA	Rivest-Shamir-Adleman
SVP	Shortest Vector Problem
CVP	Closest Vector Problem
GGH	Goldreich-Goldwasser-Halevi

LIST OF SYMBOLS

$\mathbb{R}$	The set of real numbers
$\mathbb{N}$	The set of natural numbers
$\mathbb{C}$	The set of complex numbers
$\mathbb{Z}$	The set of integers
$\mathbb{Z}^+$	The set of positive integers
$\lceil B \rceil$	Ceiling of B
$\lfloor B \rfloor$	Floor of B
$\prod_{i=1}^{n} X_i$	The finite product of spaces
$\sum_{i=1}^{n} X_i$	The finite sum of spaces
$\|B\|$	Modulus of B
$\|\|B\|\|$	Norm of matrix B
$Tr(B)$	Trace of matrix B
B^T	Transpose of matrix B
B^{-1}	Inverse of matrix B
$\mathcal{O}(n)$	Big O of n
$det(B)$	Determinant of matrix B
$deg^-(v)$	In-degree of a vertex
$deg^+(v)$	Out-degree of a vertex
λ	Eigenvalue
$\mathcal{E}_A(G)$	Energy of a graph
$\lim\limits_{n\to\infty} f(n)$	Limit of a function as n tends to infinity
$V(G)$	Vertex set of G

$E(G)$	Edge set of G
$v(G)$	The number of vertices in G or the order of G
$e(G)$	The number of edges in G or the size in G
$d_G(v)$	The degree of vertex v in G
K_n	The complete graph of order n
$\overline{K_n}$	The complement of complete graph
$N_G(v)$	Open neighbourhood of vertex v
$N_G[v]$	Closed neighbourhood of vertex v
$k - graph$	Regular graph
$A(G)$	Adjacency matrix of a graph
c	Number of components of G
C_n	Cycle graph
H	Linear subgraph
P_o	Mixed cycle with odd orientation
P_e	Mixed cycle with even orientation
$T(n)$	Total number of arithmetic operations
δG	Minimum degree of G
ΔG	Maximum degree of G
$\leq$	Less than or equal to
$\forall$	For all
$\in$	Member of
$\neq$	Not equal to
$\cup$	Union
$\exists$	There exist

MATRIKS "*HOURGLASS*": PERFAKTORAN KUADRAN SALING BERKAIT DENGAN MENGGUNAKAN PERATURAN CRAMER TERUBAHSUAI DAN GRAF CAMPURANNYA

ABSTRAK

Selama beberapa dekad, matriks *hourglass* telah secara sinonimnya merujuk kepada matriks-Z tanpa mempertimbangkan komponen-komponen pemasukan mereka secara teliti. Dalam kajian ini, telah ditunjukkan bahawa matriks hourglass sebenarnya merupakan subset kepada matriks-Z. Suatu matriks *hourglass* diperoleh dengan melakukan pertukaran baris pada setiap peringkat Pemfaktoran Saling Bersambung Kuadran (QIF), jika perlu, untuk memastikan agar pemasukan matriks yang dikira adalah bukan sifar. Secara umum, sebarang sistem linear 2×2 dalam algoritma QIF diselesaikan dengan menggunakan peraturan Cramer. Peraturan Cramer digunakan untuk memastikan agar QIF tidak mengalami masalah pada setiap peringkat proses pemfaktoran. Sementara peraturan Cramer membenarkan pertukaran lengkap vektor lajur kepada matriks pekali, kaedah peraturan Cramer terubahsuai yang diperoleh dalam tesis ini mempertimbangkan vektor lajur bersama dengan pekali matriks untuk menyelesaikan sistem persamaan linear. Kaedah peraturan Cramer terubahsuai yang dicadangkan adalah sama dengan peraturan Cramer klasik, namun berbeza daripada segi penentuan sisa relatif mereka. Keputusan yang diperolehi menunjukkan bahawa tidak terdapat perbezaan ketara dalam prestasi masa di antara peraturan Cramer dengan peraturan terubahsuai dalam QIF matriks segiempat sama tak-singular yang padat. Seterusnya, telah ditunjukkan bahawa norma Frobenius bagi kaedah terubahsuai dalam pemfaktoran adalah lebih baik berbanding dengan peraturan

Cramer, tanpa mengambil kira versi perisian MATLAB yang digunakan. Potensi aplikasi matriks *hourglass* dan QIF dalam rantaian Markov dan dalam kriptografi berasaskan *Lattice* berbanding dengan matriks-Z dan pemfaktoran WZnya telah ditekankan. Kedudukan pemasukan sifar dan bukan sifar dalam setiap matriks *hourglass* membolehkan matriks tersebut diwakili dalam bentuk graf campuran, tidak seperti matriks-Z klasik. Walaupan perwakilan matriks *hourglass* secara langsung diberi oleh graf *hourglass* campuran berpemberat, penetapan nilai pemberat 1 akan menghasilkan graf *hourglass* campuran. Dengan itu, penentu, tenaga campuran dan tenaga Laplacian bagi graf tersebut telah diperoleh. Akhirnya, matriks *hourglass* dan QIF menunjukkan kesan yang menjanjikan dalam pengkomputeraan saintifik menggunakan *multiprocessors* selari atau *mesh*.

HOURGLASS MATRIX: ITS QUADRANT INTERLOCKING FACTORIZATION USING MODIFIED CRAMER'S RULE AND ITS MIXED GRAPH

ABSTRACT

Hourglass matrix has been synonymously referring to Z-matrix for decades without properly considering the components of their entries. In this research, it is established that hourglass matrix is, in fact, a subset of Z-matrix. An hourglass matrix is obtained by carrying out row interchange at every stage of the Quadrant Interlocking Factorization (QIF), when necessary, to ensure the computed entries of the matrix are restricted to be nonzero. In general, any 2×2 linear systems in QIF algorithm is solved using Cramer's rule. Cramer's rule is used to ensure that the QIF does not breakdown at every stage of the factorization process. Though Cramer's rule allows complete substitution of column vector to the coefficient matrix, the modified Cramer's rule derived in this thesis considered the column vector together with the coefficient matrix for solving simple linear systems. The proposed methods are efficient for 2×2 linear system and are shown to be equivalent to classical Cramer's rule, but differ in their relative residual measurement. The presented results show that there is no tangible difference in performance time between the Cramer's rule and its modifications in the QIF of dense nonsingular square matrices. Furthermore, the Frobenius norm of the modified methods in the factorization are shown to be better than Cramer's rule, irrespective of the version of MATLAB used. Besides, the potential applications of hourglass matrix and its QIF in Markov chains and in lattice-based cryptography over Z-matrix and its WZ factorization are highlighted. The position of

the number of zero and nonzero entries in every hourglass matrix allows the matrix to be represented in mixed graph, unlike classical Z-matrix. Though the representation of hourglass matrix in graph gives a weighted mixed hourglass graph, assigning numerical values of 1 to the weight produces a mixed hourglass graph. Hence, the determinant, mixed energy and Laplacian energy of mixed hourglass graph are established. Finally, hourglass matrix and its QIF show promising impact in scientific computing using parallel or mesh multiprocessors.

CHAPTER 1

INTRODUCTION

1.1 General Introduction

Firstly, quadrant interlocking factorization (QIF) or WZ factorization of nonsingular matrix to yield a butterfly (hourglass) shaped dense square matrix called Z-matrix is posited by Evans and Hatzopoulos (1978). However, the appellation word "hourglass matrix" is coined by Demeure (1989) in describing the matrix derived from factorizing a square matrix, predominantly from real symmetric Toeplitz matrix ($T_n = \left[a_{i-j}\right]_{i,j=1}^{n}$) or Hankel matrix ($H_n = \left[h_{i+j-1}\right]_{i,j=1}^{n}$) by computing the entries column by column via WZ factorization or bowtie-hourglass factorization. It was further elucidated that hourglass matrix is synonymous to Z-matrix which can be partitioned into Z_{system} - a system of 2×2 triangular blocks structured (Heinig and Rost, 2005). Nevertheless, the applications and similarities between Z-matrix and hourglass matrix are indistinguishable. For instance, the matrix has not been constituted in graph theory until now, nor its utilization in lattice-based cryptography has been hinted. Although, the flaw is due to unrestricted condition of the computed entries during the factorization to be nonzero, the factorization has been modified and applied, such as in Markov models, together with its block factorization being discussed (Evans, 2002; Rhofi, 2016; Bylina, 2018). Z-matrix exists together with W-matrix during WZ factorization of non-singular matrix B, such that

$$B = WZ. \tag{1.1}$$

That is,

$$\overbrace{\begin{bmatrix} b_{1,1} & b_{1,j} & b_{1,n} \\ b_{i,1} & B_{i,j} & b_{i,n} \\ b_{n,1} & b_{n,j} & b_{n,n} \end{bmatrix}}^{B} = \overbrace{\begin{bmatrix} 1 & 0 & 0 \\ w_{i,1}^{*(1)} & W_{n-2} & w_{i,n}^{*(1)} \\ 0 & 0 & 1 \end{bmatrix}}^{W} \times \overbrace{\begin{bmatrix} z_{1,1}^{*(1)} & z_{1,j}^{*(1)} & z_{1,n}^{*(1)} \\ 0 & Z_{n-2} & 0 \\ z_{n,1}^{*(1)} & z_{n,j}^{*(1)} & z_{n,n}^{*(1)} \end{bmatrix}}^{Z},$$

where $B_{i,j}$, W_{n-2} and Z_{n-2} are matrix of size $(n-2)\times(n-2)$, where Z_{n-2} is obtained from

$$z_{i,j}^{*(k)} = z_{i,j}^{*(k-1)} + w_{i,k}^{*(k)} z_{k,j}^{*(k-1)} + w_{i,n-k+1}^{*(k)} z_{n-k+1,j}^{*(k-1)}, \tag{1.2}$$

and the entries in W are computed from $w_{i,k}^{*(k)}$ and $w_{i,n-k+1}^{*(k)}$ as

$$\begin{cases} z_{k,k}^{*(k-1)} w_{i,k}^{*(k)} + z_{n-k+1,k}^{*(k-1)} w_{i,n-k+1}^{*(k)} = -z_{i,k}^{*(k-1)} \\ z_{k,n-k+1}^{*(k-1)} w_{i,k}^{*(k)} + z_{n-k+1,n-k+1}^{*(k-1)} w_{i,n-k+1}^{*(k)} = -z_{i,n-k+1}^{*(k-1)}, \end{cases} \tag{1.3}$$

for $k = 1,2,\ldots,\lfloor\frac{n-1}{2}\rfloor; i,j = k+1,\ldots,n-k$ (Bylina, 2003).

In addition, WZ factorization exists for every nonsingular matrix, often with pivoting. Pivoting results in swapping rows or columns in a matrix or by multiplying the matrix with permutation matrices (Bornemann, 2018). The factorization mostly depends on the use of an old method, called Cramer's rule, for solving its 2×2 linear systems in Equation (1.3). Cramer's rule is one of the direct methods for solving linear systems. The rule checks every stage of QIF to avoid breakdown by applying pivoting. Though, Cramer's rule is assumed to be less practical due to its high computational time on large linear systems yet many modifications have been made on Cramer's rule to solve restricted matrix equation, Quaternionic systems, and simple and large-scale linear systems (Ufuoma, 2013; Song and Dong, 2017). If greater precision is used only for the determinant calculations, Cramer's rule offers accuracy comparable to that of LU decomposition (Habgood and Arel, 2012).

Furthermore, QIF or butterfly factorization is known for the adaptability of its direct method to solve system of linear equations (Heinig and Rost, 2011). Thus, the factorization gives rise to the use of implicit matrix elimination algorithm (that is, Parallel implicit elimination - PIE) for the solution of linear system to simultaneously compute two matrix elements (two columns at a time) for parallel implementation, unlike Gaussian elimination (GE) which computes one column at a time. This makes WZ factorization suitable for parallel computing. The efficiency of WZ factorization depends on an efficacious use of the memory echelon because computational cost often relies on both the total number of arithmetic operations used and the data transferring time between different memory levels. If there is no sufficient fast memory, then the processor will create waiting time for the data and thereby reducing its efficiency (Bylina and Bylina, 2016). Thus, an optimized algorithm for a single or dual processor may not produce the best parallel implementation as the processor scaling hit the power wall. When parallel processing is used, one processor may need output generated by another processor and hence the processors need to be interconnected. The model for the connections of these processors is from the application of graph theory (for example as path graph which has a linear array of processors). Then, Yalamov and Evans (1995) presented that WZ factorization is faster on computer with a parallel architecture than any other matrix factorization methods, such as LU factorization. Besides, WZ factorization is better than the GE and LU factorization irrespective of the number of processors used (Evans, 1993; Evans and Abdullah, 1994). It has also been shown that the factorization did better on Intel processor than AMD processor. The more processors used in the factorization the better the results to conclude on (Bylina and Bylina, 2013).

Moreover, the necessary and sufficient condition for matrix $B = \left[b_{jk}\right]_{j,k=1}^{n}$ to be a quadrant interlocking factorization or WZ factorization is that the central submatrices (also known as centro-nonsingular) $B^{c}_{n+2-2l} = \left[b_{jk}\right]_{j,k=1}^{n+1-l}$ are nonsingular, where n is even order of matrix B (the assumption also holds for odd order) and c the centered submatrix of B, for $l = 1, \ldots, m = \frac{n}{2}$. Therefore, matrix A with such property is referred to as centro-nonsingular (Rao, 1997). A centro-nonsingular matrix has every central submatrix to be nonsingular. A square matrix is nonsingular (its inverse exists) if and only if its determinant is nonzero, otherwise it is called singular or degenerate (Lipschutz et al., 2009).

Lastly, it is important to know that QIF, bowtie-hourglass factorization, butterfly factorization and WZ factorization are literally the same. Also, other notions on hourglass matrix that do not portray what we discuss in this thesis are based on hourglass stabilization techniques to preserve a full-rank stiffness for single-point element in order to reduce the hourglass effect - zero energy modes - over del operator, can be found in (Li et al., 2011; McGann et al., 2012; Warburton and Maddock, 2015).

1.2 Problem Statement

Due to the general structure of Z-matrix, many authors had classified all kinds of Z-matrix as hourglass matrix without considering the components of their entries. Unfortunately, there are changes in structure of the position of zero and nonzero entries of Z-matrix from QIF which depend on the type of matrix (Toeplitz, Hankel, Hermitian, centrosymmetric, diagonally dominant or tridiagonal matrix) being factorized. Consequently, the similarity between hourglass matrix and Z-matrix was gradually dropped over time without a cogent reason. More so, Z-matrix even if analogously refer to hourglass matrix has not been represented in graph theory because the computed entries $z_{i,j}^{*(k)}$ of Equation (1.2) are not restricted to be nonzero.

1.3 Research Objectives

The following are the objectives of this study:

1. To modify Cramer's rule in solving simple linear systems, especially for 2×2 system of linear equations.

2. To develop an algorithm for obtaining hourglass matrix from quadrant interlocking factorization of nonsingular matrix.

3. To apply the modified Cramer's rule in solving 2×2 linear systems in the quadrant interlocking factorization's algorithm to obtain optimized hourglass matrix.

4. To represent the established hourglass matrix as a mixed graph.

1.4 Motivation of the Study

The primary motivation of the research was to study an algorithm similar to Cramer's rule for solving simple scale linear systems and to apply the algorithm in quadrant interlocking factorization to obtain an optimized hourglass matrix. More so, the obtained matrix from quadrant interlocking factorization should have a direct representation in graph theory unlike the traditional Z-matrix.

1.5 Significance of the Study

The basic knowledge of matrix is required to enjoy the aesthetic and ambient nature of this research, yet there is no limitation to whosoever is interested in the beauty of hourglass matrix, not only matrix theorists and graph theorists but also cryptographers. Hourglass matrix is for the first time differentiated from the classical Z-matrix. Unlike other structured matrices which are not in resemblance with any earthly body, the nonzero entries of hourglass matrix resembles an hourglass device. We restrict the computed entries of hourglass matrix of order $n(n \geq 3)$ during QIF to be nonzero, unlike Z-matrix. This restriction allows us to know the total number of zero and nonzero entries for every hourglass matrix of order $n(n \geq 3)$, its structured block matrix and its representation as mixed graph.

1.6 Limitations of the Study

In spite of the fact that this research was carefully arranged, we are still mindful of its limitation. Due to the lack of parallel computer or mesh multiprocessors with high multicores, we limit our MATLAB codes used in this thesis on Intel processor (Core i7-4600U 2.1GHz, 8GB RAM) with standard hardware.

1.7 Thesis Organization

The following chapters provide review, background and detail on the proposed matrix and it factorization algorithm and various results of the research. Chapter 2 presents a contextual preamble and literature appraisal of Cramer's rule and its importance in quadrant interlocking factorization or WZ factorization. The WZ factorization algorithm for Z-matrix is examined and how it is synonymously refer to hourglass matrix is discussed. especially mixed graph. This review on graph theory is concentrated on unweighted mixed graphs including its mixed graph energy. In Chapter 3, we examine two modifications of Cramer's rule. The two methods are derived from one of the properties of determinant. We give the computational time of the methods and compare the residual error of the methods with Cramer's rule. Chapter 4 presents vivid explanation of quadrant interlocking factorization algorithm for obtaining hourglass matrix, the conditions to generate its entries and the application of the modified Cramer's rule to solve its 2×2 systems of linear equations during the factorization. Furthermore, the matrix norm and the performance time of the modified methods of Cramer's rule in QIF are examined. The term filanz submatrix and epicenter element are introduced to give clear proof about the determinant of hourglass matrix. The differences between hourglass matrix and Z-matrix are made known and properties of hourglass matrix are explain to justify that hourglass matrix is a subset of Z-matrix. Chapter 5 studies how hourglass matrix is represented in mixed graph. The representations lead us to many proofs about its (unweighted) mixed graph either with loops and without loops, where we examine its mixed energy. Chapter 6 gives the concluding part of the study by summarizing the results we obtained from Chapter 3 to Chapter 5. Then we suggest further investigation for interested readers and give an open problem.

CHAPTER 2

PRELIMINARIES AND LITERATURE REVIEW

2.1 Introduction

This chapter discusses the review and the leading preamble of quadrant interlocking factorization (QIF) or WZ factorization. In this study, QIF and WZ factorization will be used interchangeably and a dense nonsingular matrix B will be considered unless otherwise stated. The appraisal focuses on the Z-matrix, its QIF algorithm and the possibility of the factorization to breakdown. The proclivity of WZ factorization for parallel computing is highlighted. The group axioms of Z-system from Z-matrix are briefly scrutinized. We anatomize the reason of using QIF to LU factorization of nonsingular matrix and why Cramer's rule is used to solve the linear systems of QIF instead of LU decomposition. Over two centuries of the existence of Cramer's rule, we explore its unfaded popularity in theoretical and practical world. The review will also hinge on the graph theory. Though the matrix obtained from QIF has never been represented in graph theory, it does have the tenacity to be represented as mixed graph with fascinating results - on its spectrum and energy - and to be used in lattice-base cryptography if the conditions of the factorization are met.

2.2 Cramer's rule as direct solver

There are many direct methods for solving linear systems. The most common is LU-factorization because it breaks up the solution into two matrices by solving multiple right hand sides with only minimal effort (Galoppo et al., 2005) . There are also other linear systems solvers (also called accurate ones) such as Cramer's rule, Gaussian Elimination or Gauss Jordan. Though, Cramer's rule is least consider for practical use but has some theoretical applications and work where other direct solvers may fail (Habgood and C., 2011). Inverse matrix method for solutions of linear system has a correlation with Cramer's rule, because both are applicable when the coefficient matrix is invertible, otherwise a unique solution is not available. The rule has proven to be reliable and efficient for solving system of n linear equations in n variables. If for n linear equations in n unknowns $x_1, x_2, x_3, ..., x_n$ is defined by

$$\begin{cases} b_{1,1}x_1 + b_{1,2}x_2 + b_{1,3}x_3 + \cdots + b_{1,n}x_n = c_1 \\ b_{2,1}x_1 + b_{2,2}x_2 + b_{2,3}x_3 + \cdots + b_{2,n}x_n = c_2 \\ b_{3,1}x_1 + b_{3,2}x_2 + b_{3,3}x_3 + \cdots + b_{3,n}x_n = c_3 \\ \quad \vdots \quad + \quad \vdots \quad + \quad \vdots \quad + \cdots + \quad \vdots \quad = \vdots \\ b_{n,1}x_1 + b_{n,2}x_2 + b_{n,3}x_3 + \cdots + b_{n,n}x_n = c_n \end{cases} \tag{2.1}$$

then Equation (2.1) can equivalently be written as matrix equation of the form

$$Bx = c, \tag{2.2}$$

where the $n \times n$ matrix B (coefficient matrix) has a nonzero determinant, c the column constant term and the vector $x = (x_1, x_2, ..., x_n)^T$ is the column vector of the variables. That is,

$det(B) \neq 0,\ B = (b_{i,j})\ 1 \leq i, j \leq n,\ x = (x_1, ..., x_n)^T,\ c = (c_1, ..., c_n)^T;\ B \in \mathbb{R}^{n \times n};\ x, c \in \mathbb{R}^n$. Since matrix B must be nonsingular ($det(B) \neq 0$), then we can compute its adjoint,

$Adj(B)$, such that

$$B^{-1} = \frac{Adj(B)}{det(B)}. \tag{2.3}$$

Once the determinant of the coefficient matrix can be computed, then the adjoint of the matrix is the transpose of cofactors $(D_{i,j})$ given as

$$D_{i,j} = (-1)^{i+j} det(M_{i,j}),$$

where $M_{i,j}$ is the minor corresponding to matrix entry $b_{i,j}$ for the i^{th} row and j^{th} column of B are eliminated to have

$$Adj(B) = \begin{bmatrix} D_{1,1} & D_{1,2} & D_{1,3} & \cdots & D_{1,n} \\ D_{2,1} & D_{2,2} & D_{2,3} & \cdots & D_{2,n} \\ D_{3,1} & D_{3,2} & D_{3,3} & \cdots & D_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ D_{n,1} & D_{n,2} & D_{n,3} & \cdots & D_{n,n} \end{bmatrix}^T.$$

It is obvious that Cramer's rule provides solutions for given unknowns x_i, in linear systems.

$$x_i = B^{-1}c = \frac{Adj(B)}{det(B)}c = \frac{1}{det(B)} \begin{bmatrix} D_{1,1} & D_{1,2} & D_{1,3} & \cdots & D_{1,n} \\ D_{2,1} & D_{2,2} & D_{2,3} & \cdots & D_{2,n} \\ D_{3,1} & D_{3,2} & D_{3,3} & \cdots & D_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ D_{n,1} & D_{n,2} & D_{n,3} & \cdots & D_{n,n} \end{bmatrix}^T \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{bmatrix}.$$

Therefore,

$$x_i = \frac{D_{1,i}c_1 + D_{2,i}c_2 + D_{3,i}c_3 + \cdots + D_{n,i}c_n}{det(B)}. \tag{2.4}$$

Theorem 2.2.1. *(Higham, 2002)[Cramer's rule] Let $Bx = c$ be an $n \times n$ system of linear equations and B is an $n \times n$ matrix of x such that $det(B) \neq 0$, then the unique solution $(x_1, x_2, ..., x_n)$ to the system in Equation (2.1) is given by*

$$x_i = \frac{det(B_{i|c})}{det(B)}, \tag{2.5}$$

where $B_{i|c}$ is the matrix obtained from B by substituting the vector column of c to the ith column of B, for $i = 1, 2, ..., n$.

It is now easy to see that Equation (2.4) and Equation (2.5) are equivalent. That is,

$$x_i = \frac{D_{1,i}c_1 + D_{2,i}c_2 + D_{3,i}c_3 + \cdots + D_{n,i}c_n}{det(B)} = \frac{det(B_{i|c})}{det(B)}.$$

2.2.1 Historical background of Cramer's rule

Cramer's rule was named after a Swiss mathematician Cramer Gabriel when he hinted that resultants (determinants) might be useful in analytical geometry (Cramer, 1750). He published the technique in his paper "*Introduction à l'analyse des lignes courbes algébriques*" for solving simultaneous system of linear equations on his contribution to the theory of determinant. He further explained how to calculate the terms using his rule for determining the sign and for obtaining the numerator, and explained what happens if the denominator vanishes (Debnath, 2013a). Earlier in 1545, an Italian mathematician Gerolamo Cardano in his paper entitled in "*ars magna*" gave a clue for solving a system of linear equations which he termed as "*regula de modo*" – mother of rules (Cardano et al., 2007). Though his method was practically based on 2×2 resultants, the rule later gave what we essentially know as Cramer's rule. It was MacLaurin, a Scottish mathematician, that gave the first published results on solving two and three simultaneous linear equations in a book titled "*Treatise of Algebra*" (MacLaurin, 1748). In Maclaurin's posthumous, Boyer (1966) revealed that Cramer's rule was published two years earlier. In fact, Hedman (1999) examined a manuscript that produces undeniable proof that Maclaurin taught his students "Cramer's rule" over two decades before Cramer published it. However, Kosinski (2001) argued that the rule Maclaurin chose to allocate sign for every summand was actually wrong, though his assertion of "opposite" coefficient was right, and this was corrected by Cramer by computing the number of shifts, *dérangements*, in the permutation. Günther (1908) pointed that, for lack of good notation, Maclaurin missed the general rule for solving system of linear equations.

Over a century, Cramer's rule quickly found its way into the textbooks. Nowadays, Cramer's rule is taught not only in undergraduate mathematics but also in additional mathematics for Secondary (High) School Students (Meyer, 2000). Cramer's rule is known for solving systems of n linear equations in n variables for over two centuries and thus it gives a clear assertion for the solution of nonsingular linear equations (Hogben, 2014). For this reason, Cramer's rule has been applied in differential geometry (especially in Ricci calculus), to compute totally unimodular in integer programming, to derive the general solution to an inhomogeneous linear differential equation and to obtain Z-matrix from WZ factorization of nonsingular matrix.

2.2.2 Older modifications of Cramer's rule

According to Halmos (1980), "it may be a new proof of an old fact or it may be a new approach to several facts at the same time. If the new proof establishes same previously unsuspected connections between two ideas; it often leads to a generalization". Thus, the understanding of a particular area of interest is suddenly advanced by the discovery of a single basic equation or idea (Pickover, 2011). Therefore, many invented methods of solving systems of n linear equations in n variables have been linked to Cramer's rule (Robinson, 1970; Ehrenborg, 2004). Cramer's rule has gotten a lot of modifications than all other direct methods combined together, because the rule indicates if a system is incompatible or indeterminate without completely solving the system.

There are many previous work on Cramer's rule that made use of properties of determinants, especially cofactor, in their proofs which includes Jacobi's proof that led to Turdi's proof and rediscovered in Whitford and Klamkin (1953) . Recently, Cramer's rule has been proved via adjoint matrix and the proof by identity matrix was adopted to solve system of linear equations using elementary row operations make Cramer's rule invariant - remains unchanged after transformation (Li et al., 2014). Gauss elimination, Jacobi method and Gauss-Jordan elimination are other efficient iterative and numerical methods that have succeeded Cramer's rule, including parallel Cramer's rule (PCR) for solving singular linear systems by creating a tree like structure (Sridhar, 1987; Gu et al., 2006; Watkins, 2004).

Cramer's rule has brought debate among scholars on page in Wikipedia (Talk:Cramer's rule) regarding its complexity time, numerical stability and round off error. Some of the statements against Cramer's rule have been shown to be untrue. For instance, Habgood and Arel (2012) method depends on Chio's condensation by mirroring for solving large-scale linear systems. His work outline how Cramer's rule can be applied in a scalable manner. He showed the stability of the algorithm including its forward and backward error analysis. Unique utilization of Cramer's rule and matrix condensation techniques give an effective process that is applicable to parallel computing architectures. Then he concluded that if an accurate method for evaluating determinants is used then Cramer's rule can, in fact, be numerically stable. However, Ufuoma (2013) method also employed Dodgson's condensation with partial mirroring but with several 2×2 determinants to be computed. Thus, much advancement has been made on Cramer's rule to solve Quaternionic systems (Song and Wang, 2011; Song and Dong, 2017), least-squares solution of linear equations (Kyrchei, 2012), class of singular equations (Wang, 1989), matrix iteration (Srivastava and Gupta, 2015), WZ factorization (Bylina and Bylina, 2013), Cramer's rule for the solution of restricted matrix equation (Gu and Xu, 2008) as well as integrating Dodgson's condensation and Sylvester's determinantal identity with Cramer's rule (Li et al., 2014; Radhakrishnan et al., 2014).

2.2.3 Accuracy and Stability of Cramer's rule

For any numerical method, an algorithm should provide an accurate answer. But an admissible level of accuracy depends on application of the algorithm. Not only should an algorithm display accuracy and stability, but also efficiently balance the workloads and memory optimization (Habgood and C., 2011). In scientific computation, if solutions to problems vary enormously in magnitude, then the relative error is considered due to independent scaling. There are two categories of errors that reduce accuracy thereby causing instability: round-off error and truncation error or discretization errors (Collins, 1990). Round-off error is unavoidable but can be overcome using higher precision - more bits to store data (storing floating point values from singles to doubles). Truncations errors arise from the inability of a process to exactly represent the solution to an individual calculation due to operations that involve a finite number of steps (Quarteroni et al., 2010). If computational errors can be made small and remain small even as the problem size grows, such an algorithm is considered stable. This is because less rounding errors increases the accuracy of the algorithm's solution.

An algorithm can be stable for solving a specific problem but unstable when applied to another problem. Now, the challenge becomes presenting an argument which proves that an algorithm is stable. The two most common strategies are forward analysis and backward analysis (Higham, 2002). Forward error is a relatively intuitive measure as it bounds the algorithm's solution and the actual correct solution. However, the challenge is to provide a known correct solution for comparison. On the other hand, the backward analysis estimates the potential perturbations that the algorithm would affect the original data (Habgood and Arel, 2012). An example of a method that is forward stable but not backward stable is Cramer's rule for solving 2×2 linear systems (Higham, 2002). Nevertheless if the system is well-conditioned for a nonsingular linear system, then Cramer's rule is backward stable (Stummel, 1981). The stability of Equation (2.2) is given as

$$B(x + \delta x) = c + \delta c. \tag{2.6}$$

Due to numerical instability, Moler (1974) expressed that Cramer's rule is unsatisfactory even for 2×2 ill-conditioned linear systems because of round-off error. However, Dunham (1980) gave counter example to the statement to show that Cramer's rule is satisfactory. If better precision is utilized only for the determinant calculations, Cramer's rule offers accuracy comparable to that of LU-factorization. The rule yields a highly accurate answer than Gaussian elimination even with pivoting, especially for 2×2 linear systems. Thus, accurate method to evaluate determinants makes Cramer's rule numerically stable (Habgood and C., 2011).

2.2.4 The advantages and disadvantages of Cramer's rule

Cramer's rule has many disadvantages as it fails when the determinant of the coefficient matrix is zero, requires $n+1$ determinants each of $n \times n$ order, round off error may become significant on large problems with non-integer coefficients and also numerically unstable (Debnath, 2013a; Vein and Dale, 1999). Cramer's rule via Laplace expansion method of determinant has time complexity of $\mathscr{O}(n.n!)$, however it has been improved with other fast and concise methods such as K-Chio's method (Kaltofen and Villard, 2005; Habgood and Arel, 2012; Shores, 2007). Debnath (2013b) pointed out that to solve a system of linear equations that possesses a distinct solution, Cramer's rule requires polynomial time of

$$\left(\frac{1}{3}n^4 + \frac{1}{3}n^3 + \frac{2}{3}n^2 + \frac{1}{3}n - 1\right) \qquad \text{multiplications} \tag{2.7}$$

and

$$\left(\frac{1}{3}n^4 - \frac{1}{6}n^3 - \frac{1}{3}n^2 + \frac{1}{6}n\right) \qquad \text{additions and subtractions} \tag{2.8}$$

which gives the total number of arithmetic operations, $T(n)$, used as $T(n) \approx \frac{2}{3}n^4$ and its the asymptotic time complexity as $\mathscr{O}(n^4)$. Notwithstanding its high computational complexity, Cramer's rule is truly intriguing and it is of hypothetical significance for solving linear systems (Brunetti and Renato, 2014).

Another disadvantage of Cramer's rule is the high residual error for ill-conditioned systems. Conditioning measures the sensitivity of the solution to perturbations in the data (Higham, 2002). The condition number for coefficient matrix B of ill-conditioned system (with respect to the sensitivity of its inverse) should satisfy

$$\kappa(B) = \|B\|\|B^{-1}\|.$$

The condition number depends on the matrix B and on the norm used. If the condition number is large ($\kappa(B) \geq 1$) then matrix B is ill-conditioned, otherwise it is well-conditioned. An ill-conditioned system does not warrant a minute error of the solution when it has small residuum, since its matrix is nearly singular. A system is singular if the condition number is infinite, and ill-conditioned if it is too large because it estimates worst-case loss of precision which are difficult to solve on a computer (Strakos and Liesen, 2005). The condition number is more precisely defined to be the maximum ratio of the relative error in x to the relative residual error in c. We can check how good our result after computing $\hat{x}$ (and the symbol $||\hat{x}||$ represents the norm of the calculated solution vector $\hat{x}$) by finding the residual as $B\hat{x} - c$. The relative residual error

$$\frac{||B\hat{x} - c||}{||B||.||\hat{x}||} \leq \frac{||E||}{||B||},$$

where E is the round-off error, which almost measure accuracy (numerical stability) when working with system of linear equations (Davis, 1984; Neumaier, 1998). Roughly, the relative error norm is about the condition number times the machine precision. If a method is backward stable, then it gives a small residual base on machine precision, even for ill-conditioned problems (Datta, 2004). Stability is connected to an algorithm applied to a precise system because in the ill-conditioned system the errors on the data are amplified by the system. For ill-conditioned system, the solution can be inaccurate even if the residual is small or have an accurate solution even if the residual is large.

Cramer's rule gives a clear representation of an individual component unconnected to all other components. Another advantage of Cramer's rule is its application in quadrant interlocking factorization or WZ factorization to check if the matrix being factorized is nonsingular and to solve the linear systems of the factorization (Levin and Evans, 1991).

2.3 Z-matrix from quadrant interlocking factorization or WZ factorization

Over a century, square matrix has been the research interest of matrix theorists and graph theorists. The properties and applications of square matrices have spread their tentacles to many fields of studies (Debnath, 2013a). Nowadays, studies show that the position of zero and nonzero entries in a square matrix reflects the structure (shape) of the matrix such as Z-matrix (Garnett et al., 2014; Antony and Alemayehu, 2015).

Definition 2.3.1. *(Bylina and Bylina, 2013) Z-matrix of order $n\,(n \geq 3)$ is generally defined as*

$$Z = \begin{cases} (\underbrace{0,\ldots,0}_{i-1}, z_{i,i},\ldots,z_{i,n-i+1},0,\ldots,0)^T, & i = 1,\ldots,\lfloor\frac{(n+1)}{2}\rfloor; \\ (\underbrace{0,\ldots,0}_{n-i}, z_{i,n-i+1},\ldots,z_{i,i},0,\ldots,0)^T, & i = \lfloor\frac{(n+1)}{2}\rfloor+1,\ldots,n, \end{cases} \tag{2.9}$$

The definition of Z-matrix depends on the form on the factorized nonsingular matrix.

Definition 2.3.2. *(Bylina and Bylina, 2013) W-matrix or a bow-tie matrix of order $n\,(n \geq 3)$ is*

$$W = \begin{cases} (w_{i,1},\ldots,w_{i,i-1},1,\underbrace{0,\ldots,0}_{n-2i+1},w_{i,n-i+2},\ldots,w_{i,n}), & i = 2,\ldots,\lfloor\frac{(n+1)}{2}\rfloor; \\ (w_{i,1},\ldots,w_{i,n-i},\underbrace{0,\ldots,0}_{2i-n-1},1,w_{i,i+1},\ldots,w_{i,n}), & i = \lfloor\frac{(n+1)}{2}\rfloor+1,\ldots,n-1, \end{cases}$$

The W-matrix is called a unit W-matrix if in addition $w_{i,i} = 1$ and $w_{i,n+1-i} = 0$ for $i = 1,2,\ldots,n$ (Golpar-Raboky, 2012). Using WZ factorization of Equation (1.1) to solve equation (2.2) will result in

$$\begin{cases} Wy & = c \\ Zx & = y, \end{cases} \tag{2.10}$$

where y is an auxiliary intermediate vector. If the main diagonal entries of Z-matrix are 1's and the anti-diagonal entries are 0's, then it is referred to as unit Z-matrix. It is called unit split Z-matrix if the anti-diagonal entries are replaced with 1's (Heinig and Rost, 2004). A matrix which is both a Z-matrix and a W-matrix is called an X-matrix (Han and Kye, 2016). Thus, a factorization which is either a ZW factorization or WZ factorization is known as butterfly factorization (Heinig and Rost, 2011). A matrix which is either a Z-matrix or a W-matrix is called butterfly matrix. These names are suggested by the shapes of the set of all possible positions for nonzero entries, which are as follows:

$$W = \begin{bmatrix} \bullet & & & & & & & \bullet \\ \bullet & \circ & & & & & \circ & \bullet \\ \bullet & \circ & \circ & & & \circ & \circ & \bullet \\ \bullet & \circ & \circ & \circ & \circ & \circ & \circ & \bullet \\ \bullet & \circ & \circ & \bullet & \bullet & & \circ & \bullet \\ \bullet & \circ & \bullet & & & \bullet & \circ & \bullet \\ \bullet & \bullet & & & & & \bullet & \bullet \\ \bullet & & & & & & & \bullet \end{bmatrix} \qquad Z = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\ & \circ & \circ & \circ & \circ & \circ & \bullet & \\ & & \circ & \circ & \circ & \bullet & & \\ & & & \circ & \bullet & & & \\ & & & \bullet & \circ & & & \\ & & \bullet & \circ & \circ & \circ & & \\ & \bullet & \circ & \circ & \circ & \circ & \circ & \\ \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix}$$

$$X = \begin{bmatrix} \bullet & & & & & & & \bullet \\ & \bullet & & & & & \bullet & \\ & & \bullet & & & \bullet & & \\ & & & \bullet & \bullet & & & \\ & & & \bullet & \bullet & & & \\ & & \bullet & & & \bullet & & \\ & \bullet & & & & & \bullet & \\ \bullet & & & & & & & \bullet \end{bmatrix}.$$

A representation of a nonsingular matrix B in the form $B = ZXW$ (or $B = WXZ$) in which Z is a Z-matrix, W is a W-matrix and X is a X-matrix is called a ZW factorization of B. If X-matrix and Z-matrix are of the same order, then ZX and XZ are Z-matrices. It is called unit ZW factorization if Z is a unit Z-matrix and W is a unit W-matrix. Obviously, if B admits a ZW factorization, then it admits a unique unit ZW factorization. There are other forms of QIF or WZ factorization such as alternate quadrant interlocking factorization ($AQIF$), Cholesky QIF, QZ, QW where the matrix Q is an orthogonal matrix (Evans, 1999; Khazal, 2002). Among all, $AQIF$ possesses many general properties of QIF. Furthermore, $AQIF$ is a variant of Gaussian elimination that works from inner submatrix to outer submatrix whereas QIF is a variant of Gaussian elimination that works from outer submatrix to inner submatrix.

2.3.1 Quadrant interlocking factorization (QIF) algorithm

We assume that matrix B is a square nonsingular matrix. Furthermore, we assume matrix B is centro-nonsingular which has to be factorized into Z-matrix should be of an even size (the assumption also holds for odd order yet easier to work with even order).

If matrix B is singular then interchange columns or rows of the matrix by suitable permutation to avoid breakdown of the factorization method. For the establishment of elements in W-matrix (with 1's in its main diagonal and 0's in the antidiagonal), column ith and $(n-1)$th are solved by simultaneous equation using Cramer's rule which requires matrix B to be successfully updated and this update changes matrix B to Z-matrix. The matrix update of WZ factorization indicates the most time consuming part of the factorization. The advantage of using Cramer's rule to solve for 2×2 system of linear equations during the factorization is to check if the matrix is centro-singular and to adopt least matrix norm (Levin and Evans, 1991; Bylina and Bylina, 2014). The steps to obtain Z-matrix is as follows:

Step 1: Let $B^{(0)} = Z^{(0)}$ for initial update and obtain the first and last rows of Z-matrix as $b^{(0)}_{1,1} = z^{(0)}_{1,1}$, $b^{(0)}_{1,i} = z^{(0)}_{1,i}$, $b^{(0)}_{1,n} = z^{(0)}_{1,n}$, $b^{(0)}_{n,1} = z^{(0)}_{n,1}$, $b^{(0)}_{n,i} = z^{(0)}_{n,i}$, $b^{(0)}_{n,n} = z^{(0)}_{n,n}$, where $i = 2, \ldots, n-1$. Now, we compute $w^{(1)}_{i,1}$ and $w^{(1)}_{i,n}$ from $(n-2)$ sets of 2×2 linear system in Equation (2.11) of matrix B using Cramer's rule

$$\begin{cases} z^{(0)}_{1,1} w^{(1)}_{i,1} + z^{(0)}_{n,1} w^{(1)}_{i,n} = -z^{(0)}_{i,1} \\[2ex] z^{(0)}_{1,n} w^{(1)}_{i,1} + z^{(0)}_{n,n} w^{(1)}_{i,n} = -z^{(0)}_{i,n}. \end{cases} \tag{2.11}$$

The values of $w^{(1)}_{i,1}$ and $w^{(1)}_{i,n}$ are put in matrix form as:

$$W^{(1)} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ w^{(1)}_{2,1} & 1 & \ddots & \vdots & w^{(1)}_{2,n} \\ \vdots & 0 & \ddots & 0 & \vdots \\ w^{(1)}_{n-1,1} & \vdots & \ddots & 1 & w^{(1)}_{n-1,n} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}.$$

Step 2: We update matrix B (let $B^{(1)} = Z^{(1)}$ for the first update) and compute:

$$Z^{(1)} = W^{(1)} Z^{0}.$$

We, therefore, proceed analogously for the inner square matrices of $Z^{(1)}$ of size $(n-2)$ and so on.

Step 3: Next, we compute $w_{i,k}^{(k)}$ and $w_{i,n-k+1}^{(k)}$ from Equation (2.12) by solving its 2×2 linear equations using Cramer's rule, where $k = 1, 2, ..., \frac{n}{2} - 1;\ i = k+1, ..., n-k$.

$$\begin{cases} z_{k,k}^{(k-1)} w_{i,k}^{(k)} + z_{n-k+1,k}^{(k-1)} w_{i,n-k+1}^{(k)} = -z_{i,k}^{(k-1)} \\ z_{k,n-k+1}^{(k-1)} w_{ik}^{(k)} + z_{n-k+1,n-k+1}^{(k-1)} w_{i,n-k+1}^{(k)} = -z_{i,n-k+1}^{(k-1)}. \end{cases} \tag{2.12}$$

Then, we put the values of $w_{i,k}^{(k)}$ and $w_{i,n-k+1}^{(k)}$ in a matrix form as:

$$W^{(k)} = \begin{bmatrix} 1 & & & \\ w_{k+1,k}^{(k)} & \ddots & & w_{k+1,n-k+1}^{(k)} \\ \vdots & & \ddots & \vdots \\ w_{n-1,k}^{(k)} & & \ddots & w_{n-k,n-k+1}^{(k)} \\ & & & 1 \end{bmatrix}.$$

Step 4: We further compute for kth such successful steps as:

$$Z^{(k)} = W^{(k)} Z^{(k-1)}.$$

To arrive at the Z-matrix, we let $Z^{(k)} = Z$. Thus,

$$Z = \begin{bmatrix} z_{1,1}^{(0)} & z_{1,2}^{(0)} & z_{1,3}^{(0)} & \cdots & z_{1,n-2}^{(0)} & z_{1,n-1}^{(0)} & z_{1,n}^{(0)} \\ 0 & z_{2,2}^{(1)} & z_{2,3}^{(1)} & \cdots & z_{2,n-2}^{(1)} & z_{2,n-1}^{(1)} & 0 \\ 0 & 0 & z_{k,k}^{(k-1)} & \cdots & z_{k,n+1-k}^{(k-1)} & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & z_{n+1-k,k}^{(k-1)} & \cdots & z_{n+1-k,n+1-k}^{(k-1)} & 0 & 0 \\ 0 & z_{n-1,2}^{(1)} & z_{n-1,3}^{(1)} & \cdots & z_{n-1,n-2}^{(1)} & z_{n-1,n-1}^{(1)} & 0 \\ z_{n,1}^{(0)} & z_{n,2}^{(0)} & z_{n,3}^{(0)} & \cdots & z_{n,n-2}^{(0)} & z_{n,n-1}^{(0)} & z_{n,n}^{(0)} \end{bmatrix}.$$

The MATLAB code to compute the elements of W-matrix and Z-matrix is given in A.1. Note that the computed $z_{i,j}^{(k)}$ in Z is unbound to be nonzero, for $i, j = k+1, ..., n-k$. A complete one-stage in WZ factorization is when $Z^{(k-1)}$ is computed. Therefore, the factorization has $\lfloor \frac{(n-1)}{2} \rfloor$ stages to compute all the elements of the matrix W and Z (Evans and Hatzopoulos, 1979). After the algorithm of WZ factorization is established, the following theorems were put forward:

Theorem 2.3.1. *Factorization Theorem (Rao, 1997). Let $B \in R^{n\times n}$ be a nonsingular matrix that has a unique QIF factorization, then $B = WZ$ if and only if the submatrices of B are invertible.*

Theorem 2.3.2. *(Rao, 1997). If $B \in R^{n\times n}$ is nonsingular matrix, then there exist a row permutation matrix P for QIF to be carried out with pivoting such that $PB = WZ$.*

Theorem 2.3.3. *(Rao, 1997). Let B be a square matrix, invertible or not. There exists at least one invertible matrix M such that $MB = Z$.*

Theorem 2.3.4. *(Rao, 1997). Every symmetric positive definite and strictly diagonally dominant matrix has a QIF.*

Theorem 2.3.5. *(Rao and Kamra, 2015). Let B be nonsingular tri-diagonal diagonally dominant, then its factored Z-matrix from QIF factorization is also tri-diagonal* *diagonally dominant.*

2.3.2 Suitability and efficiency of QIF for parallel computing

Factorization of matrix B is difficult to compute and applying different optimization techniques couple with parallelism of contemporary computers makes WZ factorization extremely efficient and suitable for parallel computing. While the stability of WZ factorization comes from the centro-nonsingular matrix which is far reliable than any other type of factorization. In fact, WZ factorization is stable for every nonsingular matrix irrespective of their condition number because the algorithm has been preconditioned (Bylina and Bylina, 2007). Thus, for the factorization using Thomas' algorithm, split Levison algorithm, split Schur algorithm, BSP or ABS to speed up and solve linear systems on SIMD or MIMD shared memory parallel computers with many integrated core (MIC) in order to reduce processing time can be found in (Heinig and Rost, 2011; Golpar-Raboky, 2014; Bylina and Bylina, 2016). For parallel implementation, matrix B from Equation (2.2) is expressed as

$$\begin{bmatrix} b_{1,1} & b_{1,j} & b_{1,n} \\ b_{i,1} & B_{i,j} & b_{i,n} \\ b_{n,1} & b_{n,j} & b_{n,n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -w_{i,1} & \mathbb{I}_{n-2} & -w_{i,n} \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} b_{1,1} & b_{1,j} & b_{1,n} \\ 0 & B_{n-2} & 0 \\ b_{n,1} & b_{n,j} & b_{n,n} \end{bmatrix}, \tag{2.13}$$

where B_{n-2} is a matrix of size $(n-2)\times(n-2)$ to update B'_{n-2} as $B'_{n-2} = -w_{i,1}\,b_{1,j} + B_{n-2} - w_{i,n}\,b_{n,j}$, for $i, j = 2, 3, \ldots, n-1$.

Parallel computing is a type of computation in which many calculations (the computation can be divided into smaller subproblems) are performed at the same time. In a parallel algorithm, a single instruction stream commands the execution of the algo-