A NEW PRECONDITIONER FOR SOLVING LINEAR SYSTEMS WITH ILL-CONDITIONED Z-MATRICES

By
Isnardo Arenas Navarro

A thesis submitted in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE
in
APPLIED MATHEMATICS

UNIVERSITY OF PUERTO RICO
MAYAGÜEZ CAMPUS
July, 2011

Approved by:

ACAR Robert, Ph.D Member, Graduate Committee	Date
CASTILLO Paul, Ph.D Member, Graduate Committee	Date
YONG Xuerong, Ph.D President, Graduate Committee	Date
CEDEÑO José, Ph.D Representative of Graduate Studies	Date
COLÓN Omar, Ph.D. Chairperson of the Department	Date

Abstract of Thesis Presented to the Graduate School
of the University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master of Science

A NEW PRECONDITIONER FOR SOLVING LINEAR SYSTEMS WITH ILL-CONDITIONED Z-MATRICES

By

Isnardo Arenas Navarro

July 2011

Chair: YONG, Xuerong

Major Department: Department of Mathematical Sciences

The class of Z-matrices is the set of matrices whose off-diagonal entries are non positive. Solving a linear system of the form $Ax = b$ is possible if the coefficient matrix A has certain characteristics such as being of full rank. Linear systems in which the coefficient matrix A is a Z-matrix can be found in many processes used in different applied fields from engineering to economics. For instance, they can be found when approximating the solution of a partial differential equation (PDE) by finite difference methods. Usually, the resulting linear systems are very large and using direct methods is impractical. Then it is necessary to use iterative methods. The success of iterative methods depends on the condition number of the system. The condition number is the maximum ratio of the relative error in x divided by the relative error in b. This is a measurement that relates the behavior of the system given small changes on its right

hand side (RHS). If the condition number in a linear system is small then it is said that the system is well-conditioned, otherwise it is ill-conditioned. In many cases, large linear systems with Z-matrices as coefficient matrices have large condition numbers, which can cause iterative methods to fail. To alleviate this problem, a technique known as preconditioning is used. Basically, preconditioning is any form of modification of an original linear system that produces an equivalent system that is faster to solve than the original system.

The Gauss-Seidel is one of the most reliable and oldest iterative methods for solving linear systems, but it tends to converge slowly for ill-conditioned systems. In 2002, Hisashi Kotakemori [9] proposed a preconditioner for the Gauss-Seidel of the form $P = I + S_{max}$ for the particular case where the coefficient matrix is a diagonally dominant Z-matrix, with unit diagonal elements. Then, the problem $Ax = b$ is changed to an equivalent one, which is $PAx = Pb$.

This thesis will investigate the properties of the preconditioner $P = I + S_{max}$. As it will be shown, using this preconditioner preserves the convergence characteristics of the problem and keeps PA as a Z-matrix. Then, based on these properties, a new preconditioner will be proposed based on P, which can be used for diagonally dominant Z-matrices with positive diagonal elements, not only for unit diagonal elements. In addition, this new preconditioner can be used iteratively to improve the convergence characteristics of the problem.

Resumen de Tesis Presentada a Escuela Graduada

de la Universidad de Puerto Rico como Requisito Parcial de los

Requerimientos para el Grado de Maestría en Ciencias

UN NUEVO PRECONDICIONADOR PARA RESOLVER SISTEMAS LINEALES CON MATRICES Z MAL CONDICIONADAS

Por

Isnardo Arenas Navarro

Julio 2011

Consejero: YONG, Xuerong

Departamento: Departamento de Ciencias Matemáticas

La clase de las matrices Z es el conjunto de matrices que sus entradas fuera de la diagonal no son positivas. Resolver un sistema lineal de la forma $Ax = b$ es posible si la matriz de coeficientes tiene ciertas características como ser de rango completo. Sistemas lineales en los que la matriz de coeficientes A es una matriz Z se puede encontrar en muchos de los procesos utilizados en diferentes campos aplicados de la ingeniería y la economía. Por ejemplo, se puede encontrar cuando se aproxima la solución de una ecuación diferencial parcial (PDE) por métodos de diferencias finitas. Por lo general, los sistemas lineales resultantes son de gran tamaño lo cual hace que el uso de métodos directos sea impráctico. Entonces es necesario el uso de métodos iterativos. El éxito de los métodos iterativos depende del número de condición del sistema. El número de condición es la relación máxima del error relativo de x dividido por el error relativo

en b. Esta es una medida que relaciona el comportamiento del sistema ante cambios pequeños en su lado derecho (RHS). Si el número de condición en un sistema lineal es pequeño, entonces se dice que el sistema está bien acondicionado, de lo contrario es mal condicionado. En muchos casos los sistemas lineales grandes con matrices Z como matrices de coeficientes tienen un gran número condición, esto puede causar que los métodos iterativos fallen. Para aliviar este problema, se utiliza una técnica conocida como precondicionamiento. Básicamente, precondicionamiento es cualquier forma de modificación de un sistema lineal en uno equivalente que es más rápido para resolver que el sistema original.

El Gauss-Seidel es uno de los métodos iterativos más antiguos y confiables para resolver sistemas lineales, pero tiende a converger lentamente. En 2002, Hisashi Kotakemori en [9] propuso un precondicionador de Gauss-Seidel de la forma $P = I + S_{max}$, para el caso particular en que la matriz de coeficientes es una matriz Z diagonalmente dominante, con una diagonal principal unitaria. Entonces, el problema $Ax = b$ se cambia por un equivalente, que es $PAx = Pb$.

En esta tesis se investigarán las propiedades del precondicionador $P = I + S_{max}$. Como se verá, el uso de este precondicionador preserva las características de convergencia del problema y mantiene PA como una matriz Z. Luego, basado en estas propiedades, se propone un nuevo precondicionador fundamentado en P, el cual se puede utilizar para matrices Z diagonalmente dominantes con diagonal positiva, no sólo para las de diagonal unitaria. Además, este nuevo precondicionador puede ser utilizado de manera iterativa para mejorar las caracteristicas de convergencia del problema.

Dedicated to:

My parents: Isnardo Arenas Caicedo and Carmen Cecilia Navarro de Arenas

ACKNOWLEDGMENTS

To Dr. Paul Castillo, for everything I learned in each of his courses.

To Dr. Daniel McGee, for giving me the opportunity of being a part of his team.

To Dr. Robert Acar, for revising my work and making valuable remarks.

To my classmates Jesus Cajigas and Maider Marin, for helping with the translation of this document.

To my advisor Dr. Xuerong Yong, for his guidance and support.

CONTENTS

LIST OF SYMBOLS AND NOTATIONS

ρ	Spectral radius of a matrix.
σ	Singular values of a matrix.
λ	Eigenvalues of a matrix.
κ	Condition Number of a system.
$\|\|\|$	Norm of a vector or a matrix.
α	Scalar value.
β	Scalar value.
$*$	Denotes an entry of a matrix whose value is of no interest.
θ	Denotes an angle in radians.
A	Denotes a matrix.
x, y, z	Denote vectors.
$\tilde{x}$	Denotes a vector, which is an approximation to the solution of a system.

b	Denotes a vector or right hand side of a system.
$P, P_C, P_S, P_U, P_R, P_{SM}$	Used to denote a particular preconditioner.
P'	Used to denote a step previous of the construction the preconditioner $\tilde{P}$.
$\tilde{P}$	Used to denote the new preconditioner.
I, I_k, I_{kk}	Denotes an identity matrix.
S_{max}	Denotes a matrix used in preconditioner P.
S_M	Denotes a part used in precontioner P_{SM}.
C	Denotes a part used in preconditioner P_C.
S	Denotes a part of preconditioners P_S, P_R, P_{SM}.
R	Denotes a part of preconditioner P_R.
U	Denotes an upper triangular matrix or a part of preconditioner P_U.
L	Denotes a lower triangular matrix.
D	Denotes a diagonal matrix or the diagonal part of a matrix.
E	Denotes a strict lower triangular matrix.
F	Denotes a strict upper triangular matrix.
M	Denotes a matrix or Denotes a matrix obtained in the regular splitting for the Gauss-Seidel method.
N	Denotes a matrix obtained in the regular splitting for the Gauss-Seidel method.
M_P, N_P	Denote the matrices obtained in the regular splitting for the Gauss-Seidel method when is preconditioned with P.
$\mathcal{A}$	Denotes a precontioned matrix.
$\mathcal{M}, \mathcal{N}$	Denote the matrices obtained in the regular splitting for the Gauss-Seidel method when is preconditioned.
J	Denotes jacobian operator.
t, k, n	Denote nonnegative integers.

p	Denotes a type of norm.
ω	Denotes a scalar used with the SSOR preconditioner.
f	Denotes a function.
1D	Used to abbreviate one-dimensional.
2D	Used to abbreviate two-dimensional.
3D	Used to abbreviate three-dimensional.
RHS	Used to abbreviate right hand side.
PDE	Used to abbreviate partial diferential equations.
#	Used to abbreviate number, or numbers.

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

Solving a linear system of the form $Ax = b$ where A is a nonsingular square matrix arises in many processes and in different applied fields from engineering to economics. Also, many numerical methods rely on the solution of those systems. For example, when the Newton method is applied to find $f(x^*) = 0$ where $f : \mathbb{R}^n \to \mathbb{R}^n$, the iteration can be written as

$$x_{k+1} = x_k - (J(f)(x_k))^{-1} f(x_k)$$

where $J(f)$ is a $n \times n$ matrix. Then, for each Newton iteration, the expression $(J(f)(x_k))^{-1} f(x_k)$ needs to be computed. This is equivalent to solving the linear system $J(f)(x_k)z = f(x_k)$, where $z = x_{k+1} - x_k$.

In many applications the size of the matrix may become very large. An example of this is when the domain of a partial differential equation is discretized by partitioning it. If the domain is a cube and it is partitioned into 100 parts on each side, this will generate a matrix of size 1000000×1000000. Therefore it is impractical to calculate the inverse of the matrix or to use direct methods like Gaussian elimination. A first aproach to solve this problem is to use iterative methods to approximate the solution of the linear system.

Another problem to be considered when solving a linear system is the condition number of the matrix. The condition number gives the relationship between a disturbance on the right hand side and the approximation of the solution. To solve this problem, preconditioners are used (see Chapter 2). When the condition number of the matrix is very large, the system is called ill-conditioned. For these systems, the convergence rate of the iterative methods tends to be slow.

A special case of a linear system $Ax = b$ is when A is a Z-matrix. Z-matrices are matrices whose off-diagonal entries are less than or equal to zero. The purpose of this thesis is to study ill-conditioned systems of Z-matrices and show that iterative preconditioning can be used to improve the convergence characteritics of these systems. In that direction, the properties of the preconditioner P proposed by Hisashi Kotakemori [9] for a nonsingular diagonally dominant Z-Matrix with unit diagonal will be studied.

The main goal of this thesis is to lay down all the theoretical background and proof necessary to construct an iterative preconditioner based on P that can be used for nonsingular diagonally dominant Z-Matrices with positive diagonal. This includes extending the usage of the preconditioner P from the class of Z-matrices with unit diagonal to a wider class with positive diagonal.

The preconditioner proposed in Chapter 4 will be validated by using it to solve linear systems with Laplacian matrices from the finite difference method applied in one, two and three dimensions. The results obtained will be presented in Chapter 5. Finally, conclusions and future work will be provided in Chapter 6.

CHAPTER 2

PRECONDITIONING

There are theoretical convergence conditions that guarantee that iterative methods can obtain a good approximation to the solution of the problem $Ax = b$. These convergence conditions are different for each iterative method. However, in practice, iterative methods suffer from slow convergence. To reduce the number of iterations needed for those methods to achieve a good accuracy, preconditioning is used.

The quantity

$$\kappa_p(A) = \|A\|_p \left\|A^{-1}\right\|_p \tag{2.1}$$

which is called the condition number of the linear system with respect to the norm $\|.\|_p$, $p = 1, \ldots, \infty$ plays an important role in the solution of linear systems by iterative methods. Preconditioning is used in systems where $\kappa_p(A)$ is very large and its purpose is to make this value as small as possible. In that way, it can be guaranteed that the linear system is not unstable to small changes in the RHS. Roughly speaking, a preconditioner is any form of implicit or explicit modification of an original linear system that produces an equivalent system that is faster to solve than the original system. For example, scaling all the rows of a linear system to make the diagonal elements equal to one is an explicit form of preconditioning. The resulting system can be solved by a Krylov subspace method, shuch as Conjugate Gradient, and may require fewer steps to converge than solving the original system.

In general, the condition number $\kappa_p(A)$ obeys the following relation

$$\|r\|_p = \|b - A\tilde{x}\|_p$$

$$\frac{\|x - \tilde{x}\|_p}{\|x\|_p} \leq \kappa_p(A) \frac{\|r\|_p}{\|b\|_p} \tag{2.2}$$

When the condition number of a matrix is large, this could mean that even if the norm of the residual $\|r\|_p$ is small, the obtained approximation $\tilde{x}$ of the solution is not good.

When $p = 2$ the condition number is given by

$$\kappa_2(A) = \frac{\sigma_{max}(A)}{\sigma_{min}(A)} \tag{2.3}$$

where $\sigma_{max}(A)$ and $\sigma_{min}(A)$ are the maximum and minimum singular values of A.

2.1 Some Classic Preconditioners

This section discusses some of the preconditioners that appear in classical literature of numerical linear algebra.

Many preconditioners are based on different decompositions or splittings of the coefficient matrix A. In this section, some classical decompositions and how the preconditioners are produced from them will be discussed.

The first idea is to build an equivalent system to $Ax = b$ as follows

$$M^{-1}Ax = M^{-1}b.$$

This idea is based on decomposing the matrix into the form $A = M - R$ where R is a residual matrix, and M can be, for example, triangular. Instead of computing explicitly the result of M^{-1} or $M^{-1}A$, it is more appropiate to write the product of M^{-1} and a vector as the solution of a linear system. This process is less expensive, computationally speaking.

One easy way of defining a preconditioner is to perform an incomplete factorization of A. This implies a decomposition of the form $A = LU - R$, but the incomplete factorization can not always be achieved. Even when you have the incomplete factorization there is no guarantee that either the condition number or the number of iterations can be improved. To compute $M^{-1}x = z$, where M is factored as $M = LU$ the following steps are used:

1. Solve $Ly = x$.

2. Solve $Uz = y$.

This returns the value of the vector z. Algorithms 2.1 and 2.2 show how to solve upper triangular and lower triangular linear systems respectively. The upper triangular system is solved using the fact that the last row of the system is $z_n = \frac{y_n}{A_{n,n}}$. Then, this value can be substituted in the row before that leaving only one unknown in that row. This value can be easily obtained and the process is repeated until all the unknowns are determined. This process is known as *back substitution.* A similar approach can be followed with lower triangular systems by beginning in the first row. In that case, the process is known as *forward substitution.*

Algorithm 2.1 Solver for The Lower Triangular Part

Require: A and b

```
for i = 1 : n do
    x(i) = b(i)
    for k = 1 : i - 1 do
        x(i) = x(i) - A(i,k)x(k)
    end for
    x(i) = x(i)/A(i,i)
end for
```

Algorithm 2.2 Solver for The Upper Triangular Part

Require: A and b

```
for i = n : -1 : 1 do
    x(i) = b(i)
    for k = i + 1 : n do
        x(i) = x(i) - A(i,k)x(k)
    end for
    x(i) = x(i)/A(i,i)
end for
```

Another typical preconditioner is the Symmetric Successive Overrelaxation Method (SSOR). Consider the splitting $A = D - E - F$ where D is the diagonal part of A, $-E$ is the strict lower triangular part of A and $-F$ is the strict upper triangular part of A, then

$$M_{SSOR} = (D - \omega E)D^{-1}(D - \omega F)$$

with $L \equiv (D - \omega E)D^{-1} = (I - \omega ED^{-1})$ and $U \equiv (D - \omega F)$, when $\omega = 1$ this preconditioner is called Symmetric Gauss-Seidel. These preconditioners require that all entries of the diagonal of the matrix A are nonzero.

2.2 Preconditioner $P = I + S_{max}$

In this section the use of the preconditioner P will be considered with the Gauss-Seidel method (see Definition 2.1). The preconditioner P is used as follows,

$$PAx = Pb$$

Definition 2.1. *[15] Let $Ax = b$ and $A = M - N$ where M is the lower triangular part of A then the Gauss-Seidel iteration is given by $x_{k+1} = M^{-1}Nx_k + M^{-1}b$*

Note that when M is the lower triangular part of A and N is the strictly upper part of $-A$, then the splitting $A = M - N$ is called a Regular Splitting. Now since $N = M - A$, then replacing it in Definition 2.1 produces,

$$x_{k+1} = x_k - M^{-1}Ax_k + M^{-1}b.$$

Algorithm 2.3 shows all the steps for the Gauss-Seidel method.

Algorithm 2.3 The Gauss-Seidel Method

Require: A, x_0 and b

1: Calculate M

2: $r_0 = b - Ax_0$

3: **for** $k = 0, 1 \ldots,$ until convergence **do**

4: Solve for x_{k+1}, $M(x_{k+1} - x_k) = r_k$

5: $r_{k+1} = b - Ax_{k+1}$

6: **end for**

The convergence condition for Gauss-Seidel is given by

$$\|x_k - x^*\| \leq \rho(M^{-1}N)^k \|x_0 - x^*\|$$

where x^* is the solution for the system $Ax = b$. This method converges to a solution for any x_0 when $\rho(M^{-1}N) < 1$. When $\rho(M^{-1}N)$ is very close to 1 the number of iterations increases. So it is necessary to use a preconditioner P such that $PA = \mathcal{A}$ and for the new splitting $\mathcal{A} = \mathcal{M} - \mathcal{N}$, $\rho(\mathcal{M}^{-1}\mathcal{N})$ is smaller than $\rho(M^{-1}N)$.

In the following definitions the concepts of a Z-matrix and diagonal dominance are introduced.

Definition 2.2. *Let $A = (a_{ij})$ be a matrix, then A is called Z-Matrix if and only if $a_{ij} \leq 0$ when $i \neq j$*

Definition 2.3. *Let $A = (a_{ij})$ be a matrix, then A is called diagonally dominant if and only if $|a_{ii}| \geq \sum_{j=1\, j\neq i}^{n} |a_{ij}|$, $i = 1, \ldots, n$.*

In 2002 Hisashi Kotakemori [9] proposed $P = I + S_{max}$ as a preconditioner for a nonsingular diagonally dominant Z-Matrix with unit diagonal, which is based on Gunawardena 1991 [5]. This work proposes an extension of the result in [9] by modifying the preconditioner $P = I + S_{max}$ so that it can be used iteratively. For completeness, the construction of the preconditioner $P = I + S_{max}$ as built in [9] will be presented next.

Let A be an $n \times n$ Z-Matrix with unit diagonal, then $S_{\max} = (s_{i,j})$ is an $n \times n$ matrix such that

$$s_{i,j} = \begin{cases} -a_{i,j} & \text{if } j > i \text{ and } j = k_i \\ 0 & \text{otherwise} \end{cases} \tag{2.4}$$

where

$$k_i = \min\left\{ j \middle| \max_{k>i} |a_{i,k}| = |a_{i,j}| \right\}$$

Example. 2.1. *Let*

$$A = \begin{pmatrix} 1 & 0 & -\frac{1}{3} & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & 0 & -\frac{1}{2} \\ -\frac{1}{4} & -\frac{1}{4} & 0 & 1 & -\frac{1}{2} \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

For the splittings $A = M - N$ the result obtained is that $\rho(M^{-1}N) \approx 0.8582932135683774$, where

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 & 0 \\ -\frac{1}{4} & -\frac{1}{4} & 0 & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}, \quad N = \begin{pmatrix} 0 & 0 & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Now using Equation 2.4, the preconditioner is obtained as follows

$$S_{max} = \begin{pmatrix} 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad P = \begin{pmatrix} 1 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$PA = \begin{pmatrix} \frac{7}{8} & -\frac{1}{8} & -\frac{1}{3} & 0 & -\frac{1}{4} \\ -\frac{1}{8} & \frac{7}{8} & 0 & 0 & -\frac{3}{4} \\ -\frac{1}{2} & -\frac{1}{6} & \frac{3}{4} & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

For the splittings of $PA = \mathcal{A}$ where $\mathcal{A} = \mathcal{M} - \mathcal{N}$ and $\mathcal{M}$ is the lower triangular part

$$\mathcal{M} = \begin{pmatrix} \frac{7}{8} & 0 & 0 & 0 & 0 \\ -\frac{1}{8} & \frac{7}{8} & 0 & 0 & 0 \\ -\frac{1}{2} & -\frac{1}{6} & \frac{3}{4} & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}, \quad \mathcal{N} = \begin{pmatrix} 0 & \frac{1}{8} & \frac{1}{3} & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & \frac{3}{4} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

the obtained result is that $\rho(\mathcal{M}^{-1}\mathcal{N}) \approx 0.7377715884967286$.

2.3 Some Preconditioners Similar to $P = I + S_{max}$

In the review of literature made for this work, some similar preconditioners to $P = I + S_{max}$ were found. In many of these cases the preconditioners were proposed by the same authors of [9]. But they differ from this work in the sense that they do not use them iteratively. Some of these preconditioners are:

2.3.1 The preconditioner $P_C = I + C$

This preconditioner is proposed by Milaszewicz [11]. It is used with the Gauss-Seidel method and can be applied to Z-matrices with unit main diagonal. $C = (c_{i,j})$ is a matrix defined as

$$c_{i,j} = \begin{cases} 0 & \text{if } j \neq 1 \text{ or } i = 1 \\ -a_{i,1} & \text{otherwise} \end{cases}.$$

The structure of the preconditioner P_C is as follows,

$$P_C = I + C = \begin{pmatrix} 1 & 0 & \dots & & \dots & 0 \\ -a_{2,1} & 1 & \dots & & \dots & 0 \\ \vdots & 0 & 1 & 0 & \dots & 0 \\ & \vdots & & \ddots & & \vdots \\ \vdots & 0 & \dots & 0 & 1 & 0 \\ -a_{n,1} & 0 & \dots & \dots & 0 & 1 \end{pmatrix} \tag{2.5}$$

Example. 2.2. *Using the same matrix of Example 2.1, the preconditioner P_C is as follows*

$$P_C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 1 & 0 & 0 \\ \frac{1}{4} & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$P_C A = \begin{pmatrix} 1 & 0 & -\frac{1}{3} & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 0 & \frac{5}{6} & -\frac{1}{4} & -\frac{1}{2} \\ 0 & -\frac{1}{4} & -\frac{1}{12} & \frac{7}{8} & -\frac{1}{2} \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

Now the regular splitting for $P_C A = \mathcal{A}$, where $\mathcal{A} = \mathcal{M} - \mathcal{N}$, is as follows

$$\mathcal{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{5}{6} & 0 & 0 \\ 0 & -\frac{1}{4} & -\frac{1}{12} & \frac{7}{8} & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}, \quad \mathcal{N} = \begin{pmatrix} 0 & 0 & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

the obtained result is that $\rho(\mathcal{M}^{-1}\mathcal{N}) \approx 0.8348742347875103$.

2.3.2 The preconditioner $P_S = I + S$

This preconditioner is proposed by Gunawardena [5] for the same method and the same type of matrices as in Subsection 2.3.1. It focuses on eliminating the elements of the first upper codiagonal of A. $S = (s_{i,j})$ is a matrix defined as

$$s_i j == \begin{cases} -a_{i,j} & \text{if } j = i+1 \\ 0 & \text{otherwise} \end{cases}.$$

The structure of the preconditioner P_S is as follows,

$$P_S = I + S = \begin{pmatrix} 1 & -a_{1,2} & 0 & & \dots & 0 \\ 0 & 1 & -a_{2,3} & 0 & \dots & 0 \\ \vdots & 0 & 1 & -a_{3,4} & \ddots & 0 \\ & \vdots & & \ddots & & \vdots \\ \vdots & 0 & \dots & 0 & 1 & -a_{n-1,n} \\ 0 & 0 & \dots & \dots & 0 & 1 \end{pmatrix} \tag{2.6}$$

Example. 2.3. *Using the same matrix of Example 2.1, the preconditioner P_S is as follows*

$$P_S = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$P_S A = \begin{pmatrix} 1 & 0 & -\frac{1}{3} & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & 0 & -\frac{1}{2} \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

Now the regular splitting for $P_S A = \mathcal{A}$, where $\mathcal{A} = \mathcal{M} - \mathcal{N}$, is as follows

$$\mathcal{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}, \quad \mathcal{N} = \begin{pmatrix} 0 & 0 & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

the obtained result is that $\rho(\mathcal{M}^{-1}\mathcal{N}) \approx 0.8328351721763375$.

2.3.3 The preconditioner $P_U = I + \beta U$

This preconditioner is proposed by Kotakemori [10] for the same method and the same type of matrices as in Subsection 2.3.1. For this preconditioner, U is the strict upper triangular part of $-A$ and $\beta \geq 1$ is a scalar factor of U.

Example. 2.4. *Using the same matrix of Example 2.1, the preconditioner P_U with $\beta = 1$ is as follows*

$$P_U = \begin{pmatrix} 1 & 0 & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & 1 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$P_U A = \begin{pmatrix} \frac{17}{24} & -\frac{1}{8} & 0 & 0 & -\frac{5}{12} \\ -\frac{1}{8} & \frac{17}{24} & -\frac{1}{4} & 0 & -\frac{1}{4} \\ -\frac{1}{2} & -\frac{1}{6} & \frac{3}{4} & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

Now the regular splitting for $P_U A = \mathcal{A}$, where $\mathcal{A} = \mathcal{M} - \mathcal{N}$, is as follows

$$\mathcal{M} = \begin{pmatrix} \frac{17}{24} & 0 & 0 & 0 & 0 \\ -\frac{1}{8} & \frac{17}{24} & 0 & 0 & -0 \\ -\frac{1}{2} & -\frac{1}{6} & \frac{3}{4} & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}, \quad \mathcal{N} = \begin{pmatrix} 0 & \frac{1}{8} & 0 & 0 & \frac{5}{12} \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

the obtained result is that $\rho(\mathcal{M}^{-1}\mathcal{N}) \approx 0.6703795542311850$.

2.3.4 The preconditioner $P_R = I + S + R$

This preconditioner is proposed by Niki and Kohno, as described in [14], for the same method and the same type of matrices as in Subsection 2.3.1. For this preconditioner, $R = (r_{i,j})$ is a matrix defined as

$$r_{i,j} = \begin{cases} 0 & \text{if } i \neq n \text{ or } j = n \\ -a_{n,j} & \text{otherwise} \end{cases}.$$

The structure of the preconditioner P_R is as follows,

$$P_S = I + S + R = \begin{pmatrix} 1 & -a_{1,2} & 0 & & \dots & 0 \\ 0 & 1 & -a_{2,3} & 0 & \dots & 0 \\ \vdots & 0 & 1 & -a_{3,4} & \ddots & 0 \\ & \vdots & & \ddots & & \vdots \\ \vdots & 0 & \dots & 0 & 1 & -a_{n-1,n} \\ -a_{n,1} & -a_{n,2} & \dots & \dots & -a_{n,n-1} & 1 \end{pmatrix} \tag{2.7}$$

Example. 2.5. *Using the same matrix of Example 2.1, the preconditioner P_R is as follows*

$$P_R = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & \frac{1}{3} & \frac{1}{2} & 0 & 1 \end{pmatrix}$$

and

$$P_R A = \begin{pmatrix} 1 & 0 & -\frac{1}{3} & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & 0 & -\frac{1}{2} \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ -\frac{1}{4} & 0 & 0 & -\frac{1}{6} & \frac{7}{12} \end{pmatrix}$$

Now the regular splitting for $P_R A = \mathcal{A}$, where $\mathcal{A} = \mathcal{M} - \mathcal{N}$, is as follows

$$\mathcal{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 & 0 \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ -\frac{1}{4} & 0 & 0 & -\frac{1}{6} & \frac{7}{12} \end{pmatrix}, \quad \mathcal{N} = \begin{pmatrix} 0 & 0 & \frac{1}{3} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

the obtained result is that $\rho(\mathcal{M}^{-1}\mathcal{N}) \approx 0.7750459262368632$.

2.3.5 The preconditioner $P_{SM} = I + S + S_M$

This preconditioner is proposed by Sakakihara [12] for the same method and the same type of matrices as in Subsection 2.3.1. This preconditioner is based in the preconditioner P_S discussed in Subsection 2.3.2, where S is the first upper codiagonal of $-A$. $S_M = (s_{i,j})$ is obtained as follows,

$$s_{i,j} = \begin{cases} -a_{i,j} & \text{if } j > i+1 \text{ and } j = k_i \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

where

$$k_i = \min \left\{ j | \max_{k>i+1} |a_{i,k}| = |a_{i,j}| \right\}$$

Example. 2.6. *Using the same matrix of Example 2.1, the preconditioner P_{SM} is as follows*

$$P_{SM} = \begin{pmatrix} 1 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and

$$P_{SM}A = \begin{pmatrix} 1 & 0 & -\frac{1}{3} & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & 0 & -\frac{1}{2} \\ -\frac{1}{4} & -\frac{5}{12} & -\frac{1}{4} & 1 & 0 \\ -\frac{1}{4} & 0 & 0 & -\frac{1}{6} & \frac{7}{12} \end{pmatrix}$$

In particular for this example, $P_{SM} = P = I + S_{max}$, so the results are identical to those of Example 2.1.

The following example shows that the preconditioner $P = I + S_{max}$ is different from P_{SM}.

Example. 2.7. *Let*

$$A = \begin{pmatrix} 1 & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 \\ 0 & 1 & -\frac{1}{3} & 0 & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 1 & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & 0 & 1 & -\frac{1}{2} \\ 0 & -\frac{1}{3} & -\frac{1}{2} & 0 & 1 \end{pmatrix}$$

Then

$$P_{SM} = \begin{pmatrix} 1 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 1 & \frac{1}{3} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad P = I + S_{max} = \begin{pmatrix} 1 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

CHAPTER 3

PROPERTIES OF Z-MATRICES AND THE PRECONDITIONER $P = I + S_{MAX}$

This chapter will present a series of lemmas that will lay the background necessary to understand the preconditioner $P = I + S_{max}$ and achieve the modification. From now on, it is assumed that A is a nonsingular and diagonally dominant Z-matrix, P is a preconditioner with nonnegative real entries, and x and b are vectors of suitable sizes. Lemma 3.1 is a classical result that allows to bound the spectral radius of matrices with all entries positive. Lemma 3.2 was proven during this work and it guarantees that the product PA is still a Z-matrix. So, using this product iteratively will maintain that property. Lemmas 3.3 and 3.4 are used to extend the usage of preconditioner P to matrices with positive diagonal. Those lemmas show that any Z-matrix with a positive main diagonal can be transformed to one with unitary diagonal, thus conserving the properties of being a Z-matrix and without losing the quality of its spectral radius. Lemmas 3.5 and 3.7 are used to extend the usage of preconditioner P to matrices with positive diagonal iteratively. Lemma 3.5 shows that the product PA preserves the diagonal dominant property. Therefore P can be used iteratively preserving that property. Lemma 3.7 was proven during this study and it says that the splitting of the product PA is bounded by the splitting of the original coefficient matrix. This last result is of importance since it will be used to show that using P in an iterative fashion will produce a better spectral radius.

In the following Lemma the notation $A \geq 0$ is used to indicate that the matrix or vector is nonnegative i.e. that all entries of A are greater than or equal to zero. The notation $A > 0$ is used to indicate that the matrix or vector is positive i.e. that all entries of A are greater than zero.

Lemma 3.1 ([6, Corollary 8.1.29]). *Let A an $n \times n$ Matrix, $x \in \mathbb{R}^n$, and suppose that $A \geq 0$ and $x > 0$. If $\alpha \geq 0$ is such that $Ax \leq \alpha x$, then $\rho(A) \leq \alpha$. From this it follows that if $Ax < \alpha x$, then $\rho(A) < \alpha$.*

Lemma 3.2. *If A is an $n \times n$ Z-matrix with unit diagonal and $P = (I + S_{\max})$, then PA is a Z-matrix.*

Proof. Since

$$P = \begin{pmatrix} 1 & \dots & -a_{1,k_1} & & & & \dots \\ 0 & 1 & & \dots & -a_{2,k_2} & & \dots \\ 0 & 0 & & 1 & \dots & -a_{3,k_3} & \dots \\ \vdots & \dots & & & \ddots & & \\ 0 & 0 & & \dots & 0 & 1 & -a_{n-1,k_{n-1}} \\ 0 & 0 & & \dots & 0 & 0 & 1 \end{pmatrix}$$

A is a Z-matrix and $k_i \neq i$ then $a_{i,k_i} \leq 0$. Morever $(PA)_{i,j} = a_{i,j} - a_{i,k_i} a_{k_i,j} \leq 0$ when $i \neq j$, so PA is a Z-matrix. ■

Lemma 3.3. *If A is a $n \times n$ Z-Matrix and $D = \operatorname{diag}(\mu_1, \mu_2, \dots, \mu_n)$ with $\mu_i \geq 0$, then DA is a Z-matrix.*

Proof. Since

$$DA = (\mu_i a_{i,j})$$

when $i \neq j$, $\mu_i > 0$ and $a_{i,j} \leq 0$ then $\mu_i a_{i,j} \leq 0$. ■

Lemma 3.4. *Let A an $n \times n$ Matrix, $D = \operatorname{diag}(\mu_1, \mu_2, \dots, \mu_n)$ then $\rho(M^{-1}N) = \rho(M_D^{-1}N_D)$ where $A = M - N$ and $DA = M_D - N_D$ are the regular Gauss-Seidel splittings.*

Proof. Note that for diagonal matrix D, the regular splittings of DA are $M_D = DM$ and $N_D = DN$, so $\rho(M_D^{-1}N_D) = \rho((DM)^{-1}DN) = \rho(M^{-1}D^{-1}DN) = \rho(M^{-1}N)$. ■

Note that the result used ($M_D = DM$ and $N_D = DN$) in 3.4 is not true when D is not diagonal, which would have a different splitting.

Lemma 3.5. *If A is an $n \times n$ Z-matrix diagonally dominant with unit diagonal and non-singular, $P = (I + S_{\max})$, then PA is a diagonally dominant Z-matrix with positive diagonal and non-singular.*

Proof. Since $(PA)_{i,j} = a_{i,j} - a_{i,k_i} a_{k_i,j}$, using that $0 \leq a_{i,k_i} a_{k_i,i} < 1$ for $i \neq j$ then $(PA)_{i,i} = 1 - a_{i,k_i} a_{k_i,i} > 0$. Hence PA has a positive diagonal, and $1 \geq \sum_{j=1\, j\neq i}^{n} |a_{i,j}|$, but A is a Z-matrix so

$$1 \geq \sum_{j=1\, j\neq i}^{n} -a_{i,j}. \tag{3.1}$$

Hence $\sum_{j=1}^{n} a_{i,j} \geq 0$ for all i.

Now using the result in Lemma 3.2

$$\begin{aligned}
|(PA)_{i,i}| - \sum_{j=1\, j\neq i}^{n} |(PA)_{i,j}| &= 1 - a_{i,k_i} a_{k_i,i} - \sum_{j=1\, j\neq i}^{n} -(PA)_{i,j} \\
&= 1 - a_{i,k_i} a_{k_i,i} + \sum_{j=1\, j\neq i}^{n} a_{i,j} - a_{i,k_i} a_{k_i,j} \\
&= \sum_{j=1}^{n} a_{i,j} - a_{i,k_i} \sum_{j=1}^{n} a_{k_i,j} \\
&\geq 0 \qquad (3.2)
\end{aligned}$$

So $|(PA)_{i,i}| \geq \sum_{j=1\, j\neq i}^{n} |(PA)_{i,j}|$. Hence PA is a diagonally dominant Z-matrix. ∎

Note that if the inequality in 3.1 is strict, then 3.2 has a strict inequality too.

The following result is classical in numerical linear algebra and is true for any irreducibly diagonally dominant matrix. But in this case, only a short version of the same for Z-matrices is needed.

Lemma 3.6. *Let A be an irreducibly diagonally dominant Z-matrix, then Gauss-Seidel splitting $A = M - N$ yields convergence and*

$$\rho(M^{-1}N) < 1$$

Proof. See the proof in [15, Theorem 4.5] ■

Lemma 3.7. *Let A be a diagonally dominant Z-matrix with unit diagonal, non-singular using regular splittings $A = M - N$, $PA = M_P - N_P$ and with $\rho(M^{-1}N) < 1$ then*

$$\rho(M_P^{-1}N_P) \leq \rho(M^{-1}N)$$

Proof. By putting $A = P^{-1}(M_P - N_P)$, one has that $A = M - N = P^{-1}(M_P - N_P)$. Since $\rho(M^{-1}N) < 1$ there exists a positive vector, x, satisfying $M^{-1}Nx = \rho(M^{-1}N)x$. Then

$$Ax = (M - N)x = M(I - M^{-1}N)x = \frac{1 - \rho(M^{-1}N)}{\rho(M^{-1}N)}Nx \geq 0$$

Since $M_P^{-1} \geq 0$ and $P \geq I \geq 0$, then $M_P^{-1}P \geq M_P^{-1} \geq M^{-1}$ and it follows that

$$\begin{aligned}
(M_P^{-1}P - M^{-1})Ax &\geq 0 \\
M_P^{-1}PAx - M^{-1}Ax &\geq 0 \\
M_P^{-1}P(P^{-1}(M_P - N_P))x - M^{-1}(M - N)x &\geq 0 \\
(I - M_P^{-1}N_P)x - (I - M^{-1}N)x &\geq 0 \\
M^{-1}Nx - M_P^{-1}N_Px &\geq 0 \\
\rho(M^{-1}N)x &\geq M_P^{-1}N_Px.
\end{aligned}$$

Finally, using Lemma 3.1, $\rho(M_P^{-1}N_P) \leq \rho(M^{-1}N)$. ■

CHAPTER 4

THE PROPOSED METHOD

By the previous results, a new preconditioner can be introduced. This new preconditioner is based on the preconditioner P but it can be applied iteratively for any irreducibly diagonally dominant Z-matrix with a positive diagonal. This is an improvement on the work done in [9] since the discussions there only apply for Z-matrices with unit diagonal.

Given an irreducibly diagonally dominant Z-matrix with positive diagonal, say A, it can be transformed into an irreducibly diagonally dominant Z-matrix with unit diagonal. This can be achieved by multiplying A by a matrix $D = \operatorname{diag}\left(\frac{1}{a_{1,1}}, \frac{1}{a_{2,2}}, \ldots, \frac{1}{a_{n,n}}\right)$, then the preconditioner P for the product DA is

$$P = \begin{pmatrix} 1 & \dots & & & -\frac{a_{1,k_1}}{a_{1,1}} & & \dots \\ 0 & 1 & \dots & & -\frac{a_{2,k_2}}{a_{2,2}} & & \dots \\ 0 & 0 & 1 & \dots & -\frac{a_{3,k_3}}{a_{3,3}} & & \dots \\ \vdots & \dots & & \ddots & & & \\ 0 & 0 & \dots & 0 & 1 & -\frac{a_{n-1,k_{n-1}}}{a_{n-1,n-1}} \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}.$$

The coefficient matrix of the preconditioned system is $P(DA)$. If the multiplication is associated as $(PD)A$ and $P' = PD$, one has that

$$P' = \begin{pmatrix} \frac{1}{a_{1,1}} & \dots & & & -\frac{a_{1,k_1}}{a_{1,1}a_{k_1,k_1}} & & \dots \\ 0 & \frac{1}{a_{2,2}} & \dots & & -\frac{a_{2,k_2}}{a_{2,2}a_{k_2,k_2}} & & \dots \\ 0 & 0 & \frac{1}{a_{3,3}} & \dots & -\frac{a_{3,k_3}}{a_{3,3}a_{k_3,k_3}} & & \dots \\ \vdots & \dots & & \ddots & & & \\ 0 & 0 & \dots & 0 & \frac{1}{a_{n-1,n-1}} & -\frac{a_{n-1,k_{n-1}}}{a_{n-1,n-1}a_{k_{n-1},k_{n-1}}} \\ 0 & 0 & \dots & 0 & 0 & \frac{1}{a_{n,n}} \end{pmatrix}.$$

The number of operations in P' can be reduced if $D^{-1} = \operatorname{diag}(a_{1,1}, a_{2,2}, \dots, a_{n,n})$ is considered. Then the obtained preconditioned system will be $D^{-1}(PDA)$, associating the multiplication $(D^{-1}PD)A$ and letting

$$\tilde{P} = D^{-1}PD \tag{4.1}$$

yields the following form

$$\tilde{P} = \begin{pmatrix} 1 & \dots & & & -\frac{a_{1,k_1}}{a_{k_1,k_1}} & & \dots \\ 0 & 1 & \dots & & -\frac{a_{2,k_2}}{a_{k_2,k_2}} & & \dots \\ 0 & 0 & 1 & \dots & -\frac{a_{3,k_3}}{a_{k_3,k_3}} & & \dots \\ \vdots & & & \ddots & & & \\ 0 & 0 & \dots & 0 & & 1 & -\frac{a_{n-1,k_{n-1}}}{a_{k_{n-1},k_{n-1}}} \\ 0 & 0 & \dots & 0 & & 0 & 1 \end{pmatrix}$$

$\tilde{P}$ is the new preconditioner proposed in this work.

The next Lemma shows that the iterative use of $\tilde{P}$ does not have an asymptotic behavior. That is, if the preconditioner $\tilde{P}$ is applied several times, then the spectral radius will keep decreasing until it reaches a value of zero.

Lemma 4.1. *Let A be an irreducibly diagonally dominant $n \times n$ Z-matrix with positive diagonal,then there exists $t \in \mathbb{N}$ such that A_t is lower triangular (where $A_t = \tilde{P}_{t-1}A_{t-1}$ and $A_0 = A$).*

Proof. For $n = 2$

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}, \quad \tilde{P}_1 = \begin{pmatrix} 1 & -\frac{a_{1,2}}{a_{2,2}} \\ 0 & 1 \end{pmatrix}$$

Hence

$$A_1 = \begin{pmatrix} a_{1,1} - \frac{a_{1,2}a_{2,1}}{a_{2,2}} & 0 \\ a_{2,1} & a_{2,2} \end{pmatrix}$$

Now suppose that for some k there exists t_k such that for any irreducibly diagonally dominant A $(k \times k)$, Z-matrix with positive diagonal, A_{t_k} is lower triangular.

It will be proved that for $k+1$ there exists t_{k+1} such that for any irreducibly diagonally dominant A $(k+1 \times k+1)$ Z-matrix with positive diagonal, $A_{t_{k+1}}$ is lower triangular.

$$A = \left(\begin{array}{c|ccccc} a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} & \dots & a_{0,k} \\ \hline a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,k} \\ a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,k} \\ a_{3,0} & a_{3,1} & a_{3,2} & a_{3,3} & \dots & a_{3,k} \\ \vdots & \dots & & & \ddots & \\ a_{k,0} & a_{k,1} & a_{k,2} & a_{k,3} & \dots & a_{k,k} \end{array}\right)$$

Note that

$$\tilde{P}_l = \left(\begin{array}{c|ccccc} 1 & \dots & 0 & * & 0 & \dots \\ \hline 0 & & & & & \\ 0 & & & \tilde{\mathcal{P}}_l & & \\ 0 & & & & & \\ 0 & & & & & \end{array}\right)$$

where $\tilde{\mathcal{P}}_l$ is a $k \times k$ preconditioner. By the induction hypothesis A_{t_k} has a lower triangular block as follows,

Then

$$A_{t_k} = \left(\begin{array}{c|ccccc} * & * & * & * & \dots & * \\ \hline * & * & 0 & 0 & \dots & 0 \\ * & * & * & 0 & \dots & 0 \\ * & * & * & * & \ddots & 0 \\ \vdots & \dots & & & \ddots & 0 \\ * & * & * & * & \dots & * \end{array}\right)$$

since A_{t_k} is Z-matrix then $(A_{t_k})_{0,k} \leq 0$

This will be analyzed in two cases when $(A_{t_k})_{0,k} = 0$ and $(A_{t_k})_{0,k} < 0$

- Case 1

  If $(A_{t_k})_{0,k} = 0$, A_{t_k} will be as follows

$$A_{t_k} = \left(\begin{array}{c|cccccc} * & * & * & * & \dots & 0 \\ \hline * & * & 0 & 0 & \dots & 0 \\ * & * & * & 0 & \dots & 0 \\ * & * & * & * & \ddots & 0 \\ \vdots & \dots & & & \ddots & 0 \\ * & * & * & * & \dots & * \end{array}\right)$$

  Thus changing the partition of A_{t_k} as follows

$$A_{t_k} = \left(\begin{array}{cccccc|c} * & * & * & \dots & & * & 0 \\ * & * & 0 & \dots & & 0 & 0 \\ * & * & * & & 0 & \dots & 0 \\ * & * & * & & * & \ddots & \vdots \\ \vdots & & & & & \ddots & 0 \\ \hline * & * & * & & * & \dots & * \end{array}\right)$$

  Finally, using the hypothesis of induction, A_{2t_k} is lower triangular.

- Case 2

  If $(A_{t_k})_{0,k} < 0$, and if through $\tilde{t}$ iterations more with $\tilde{t} < t_k$ the entry becomes 0, then

$$A_{t_k+\tilde{t}} = \left(\begin{array}{ccccc|c} * & * & * & * & \dots & 0 \\ * & * & 0 & 0 & \dots & 0 \\ * & * & * & 0 & \dots & 0 \\ * & * & * & * & \ddots & \vdots \\ \vdots & \dots & & & \ddots & 0 \\ \hline * & * & * & * & \dots & * \end{array}\right).$$

Hence, using the hypothesis of induction, $A_{2t_k+\tilde{t}}$ is lower triangular.

If $(A_{t_k})_{0,k} < 0$, and if through t_k iterations more the entry never becomes 0, then by the hypothesis of induction

$$A_{2t_k} = \left(\begin{array}{ccccc|c} * & 0 & 0 & 0 & \dots & * \\ * & * & 0 & 0 & \dots & 0 \\ * & * & * & 0 & \dots & 0 \\ * & * & * & * & \ddots & \vdots \\ \vdots & \dots & & & \ddots & 0 \\ \hline * & * & * & * & \dots & * \end{array}\right)$$

so A_{2t_k+1} is lower triangular.

Hence, it is sufficient to take $t_{k+1} = 3 * t_k$ and $A_{t_{k+1}}$ is lower triangular. ■

Considering the construction of the preconditioner $\tilde{P}$ and the result in Lemma 4.1, Theorem 4.2 is proved.

Theorem 4.2. *If A is Z-matrix with positive diagonal, irreducibly diagonally dominant and $A_0 = A$, $A_t = \tilde{P}_{t-1}A_{t-1}$, with Gauss-Seidel regular splittings $A_t = M_t - N_t$ then* *there exists a* $k \in \mathbb{N}$ *such that*

$$1 > \rho(M_0^{-1}N_0) \geq \rho(M_1^{-1}N_1) \geq \rho(M_2^{-1}N_2) \geq \ldots \geq \rho(M_k^{-1}N_k) = 0$$

4.1 How to use the proposed preconditioner

Now that all the results have been obtained and that the new preconditioner is contructed, the following discussion explains how to use it.

Given a linear system $Ax = b$, where A is a diagonally dominant Z-matrix with positive main diagonal and irreducible, the first step is to fix a nonnegative integer t. This t will be the amount of times that the preconditioner will be applied iteratively. Now since t is fixed, at each step the following is calculated

$$A_t = \tilde{P}_{t-1}A_{t-1} \quad \text{and} \quad b_t = \tilde{P}_{t-1}b_{t-1}$$

$$(\text{ where } A_0 = A \text{ and } b_0 = b)$$

Finally, after applying the preconditioner t times, the approximation of the solution in the linear system should be calculated. For this, the Gauss-Seidel method (see Algorithm 2.3) is used. The whole idea of this process is summarized in the following algorithm.

Algorithm 4.1 The Proposed Method

Require: A, x_0, b and t

$A_0 = A$ and $b_0 = b$

for $i = 1, \ldots, t$ **do**

  Obtain preconditioner $\tilde{P}_{i-1}$ for A_{i-1}

  $A_i = \tilde{P}_{i-1}A_{i-1}$, $b_i = \tilde{P}_{i-1}b_{i-1}$

end for

Solving $A_t x = b_t$ by Gauss-Seidel 2.3

Note that when $t = 0$, Algorithm 4.1 is equivalent to solving $Ax = b$ without the preconditioner.

CHAPTER 5

RESULTS

Many of the linear systems obtained in different applied fields are ill-conditioned. This makes the preconditioners occupy a privileged place in the study of applied mathematics.

Some examples of these ill-conditioned linear systems in applied fields are:

5.1 One-Dimensional Laplacian

In the problem of the One-Dimensional Laplacian, the matrix that is obtained is a tridiagonal given by:

$$\begin{pmatrix} 2 & -1 & 0 & 0 & & \dots & 0 \\ -1 & 2 & -1 & 0 & & \dots & 0 \\ 0 & -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & & \ddots & \ddots & \ddots & & 0 \\ 0 & \dots & 0 & -1 & 2 & -1 & 0 \\ 0 & \dots & & 0 & -1 & 2 & -1 \\ 0 & \dots & & & 0 & -1 & 2 \end{pmatrix}_{n \times n}$$

The eigenvalues of this matrix are given by

$$\lambda_j = 2 - 2\cos(j\theta) \quad j = 1, \dots, n, \tag{5.1}$$

where

$$\theta = \frac{\pi}{n+1}$$

Note that since $0 < j\theta < \pi$, all eigenvalues λ_j are positive, where λ_1 is the minimum value in the module and λ_n is the maximum value in the module. Hence, using the 2-norm, the condition number (see Equation 2.3) is given by

$$\kappa_2 = \frac{2 - 2\cos(\frac{n\pi}{n+1})}{2 - 2\cos(\frac{\pi}{n+1})}$$

The condition number κ_2 for this matrix is very large when n is big, so this matrix is ill-conditioned. This is shown in Figure 5.1.

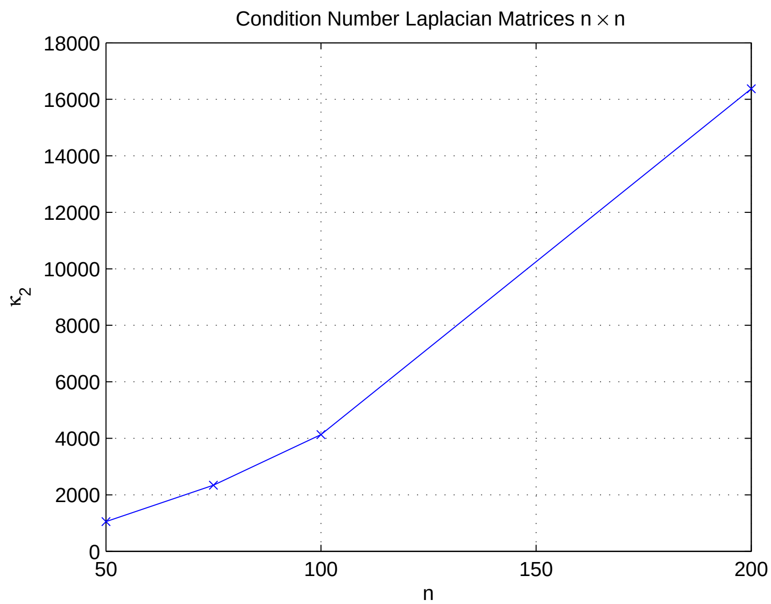

Figure 5.1: Condition Number κ_2 for 1D Laplacian Matrices $n \times n$

Example. 5.1. *For A an $n \times n$ Laplacian matrix with $x^* = [1, 1, \ldots, 1]^T$ and $b = Ax^*$, taking $x_0 = [0, 0 \ldots, 0]$, a tolerance 1×10^{-6}, 4000 as the maximum number of iterations, and the proposed preconditioner $\tilde{P}$ (see Equation 4.1).*

This matrix is obtained from the discretization of the equation $-\frac{d^2u}{dx^2} = f(x)$ using finite differences.

Figure 5.2 shows the number of iterations versus the number of times that the matrix was iteratively preconditioned.

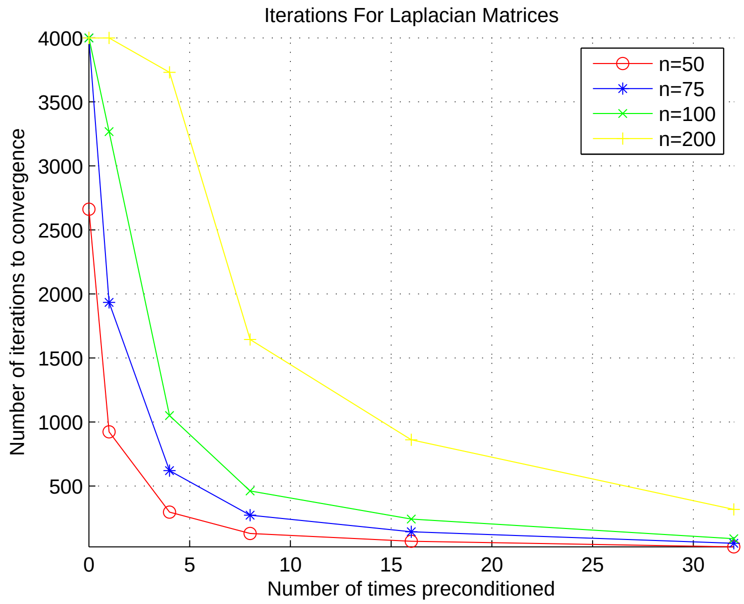

Figure 5.2: Number of iterations for 1D Laplacian matrices

	# of times preconditioned					
n	0	1	4	8	16	32
50	2662	923	297	130	69	26
75	4000	1934	621	273	143	53
100	4000	3268	1051	462	242	89
200	4000	4000	3731	1644	862	318

Table 5.1: Iterations For One-Dimensional Laplacian Matrices

Table 5.1 shows the size of the matrices versus number of times the matrices has been iteratively preconditioned to a maximum of 32 times. Note that when the preconditioner is applied only one time (column 3 in Table 5.1), it is equivalent to the preconditioner proposed in [9]. Using $\tilde{P}$ iteratively improves the number of iterations needed to convergence in the Gauss-Seidel method.

5.2 Two-Dimensional Laplacian

In the problem of the Two-Dimensional Laplacian, the linear system that is obtained has the following structure. Figure 5.3 shows the position of the entries of the coefficient matrix of the Two-Dimensional Laplacian using Matlab's function $spy(A)$.

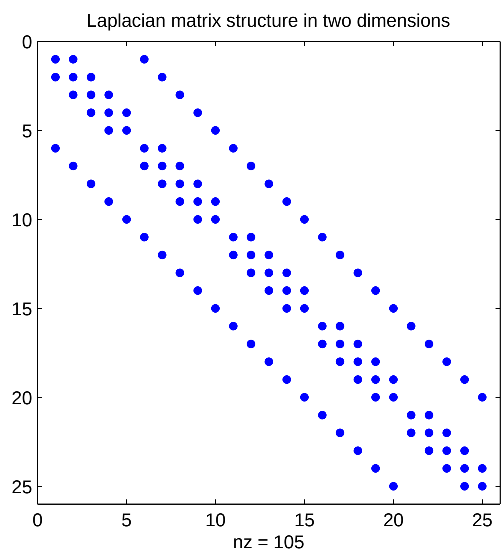

Figure 5.3: 2D Laplacian Matrix Structure

Figure 5.4 shows the condition number estimate of the matrix. Those values are obtained using the function "condest()" in Matlab. The value of the condition number estimate is very large for these matrices and increases with the value of n. Hence this matrix is ill-conditioned.

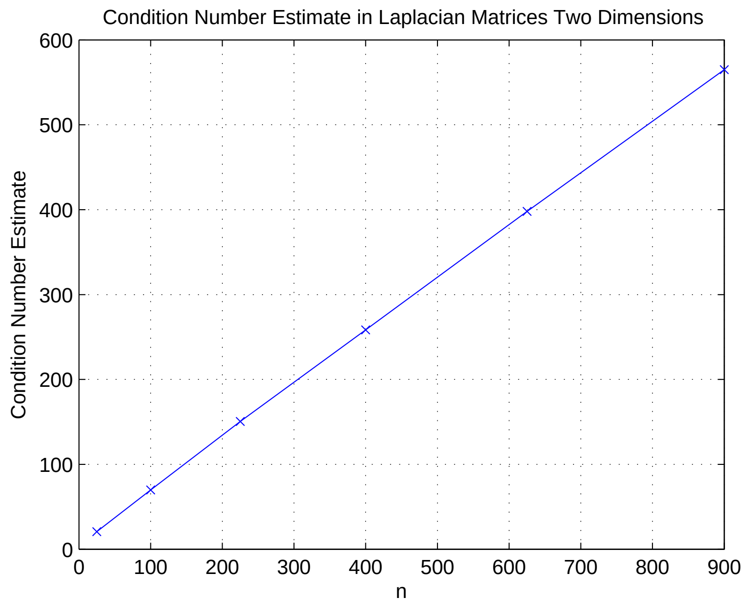

Figure 5.4: Condition Number Estimate for 2D Laplacian Matrices $n \times n$

Example. 5.2. *For $n \times n$ matrices that are from the Two-Dimensional Laplacian using a $k \times k$ grid with $n = k^2$ $x^* = [1, 1, \ldots, 1]^T$ and $b = Ax^*$, taking $x_0 = [0, 0 \ldots, 0]$, a tolerance 1×10^{-6}, 4000 as the maximum number of iterations, and the proposed preconditioner $\tilde{P}$ (see Equation 4.1).*

$$A = \begin{pmatrix} B_k & -I_k & 0 & \ldots & 0 \\ -I_k & B_k & -I_k & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & -I_k & B_k & -I_k \\ 0 & \ldots & 0 & -I_k & B_k \end{pmatrix}_{n \times n}$$

This matrix is obtained from the discretization of the equation $-\left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}\right) =$

$f(x_1, x_2)$ using finite differences, where the block B_k is as follows

$$B_k = \begin{pmatrix} 4 & -1 & 0 & \dots & 0 \\ -1 & 4 & -1 & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & -1 & 4 & -1 \\ 0 & \dots & 0 & -1 & 4 \end{pmatrix}_{k \times k}$$

and I_k is the $k \times k$ identity matrix.

Figure 5.5 shows the number of iterations versus the number of times that the matrix was iteratively preconditioned.

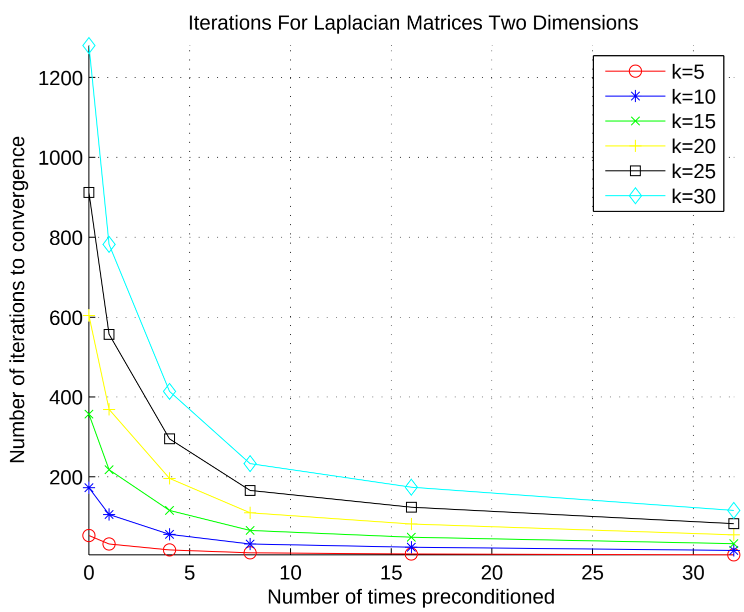

Figure 5.5: Number of iterations for 2D Laplacian matrices

	# of times preconditioned					
k	0	1	4	8	16	32
5	53	32	17	10	7	5
10	173	106	56	32	24	16
15	357	218	116	66	49	33
20	604	369	196	110	82	55
25	912	557	295	166	124	83
30	1280	782	414	233	174	116

Table 5.2: Iterations For Two-Dimensional Laplacian Matrices

As in the case of Example 5.1, using $\tilde{P}$ iteratively makes the Gauss-Seidel method need less iterations to converge, as shown in Table 5.2.

5.3 Three-Dimensional Laplacian

In the problem of the Three-Dimensional Laplacian, the linear system that is obtained has the following structure. As in Figure 5.3, Figure 5.6 shows the position of the entries of the coefficient matrix of the Three-Dimensional Laplacian using Matlab's function $spy(A)$.

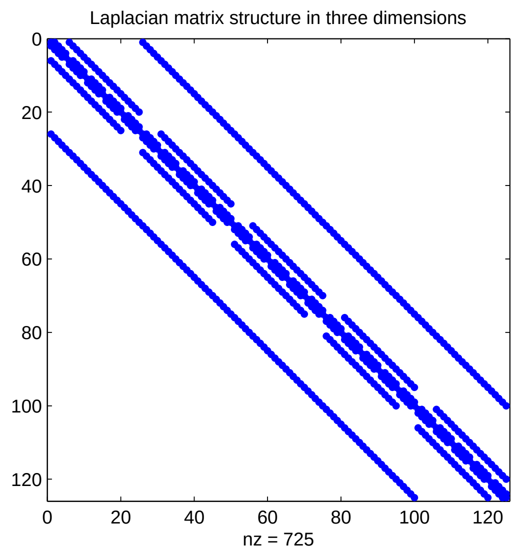

Figure 5.6: 3D Laplacian Matrix Structure

Figure 5.7 shows the condition number estimate of the matrix. Those values are obtained using the function "condest()" in Matlab. The value of the condition number estimate is very large for these matrices and increases with the value of n. Hence this matrix is ill-conditioned.

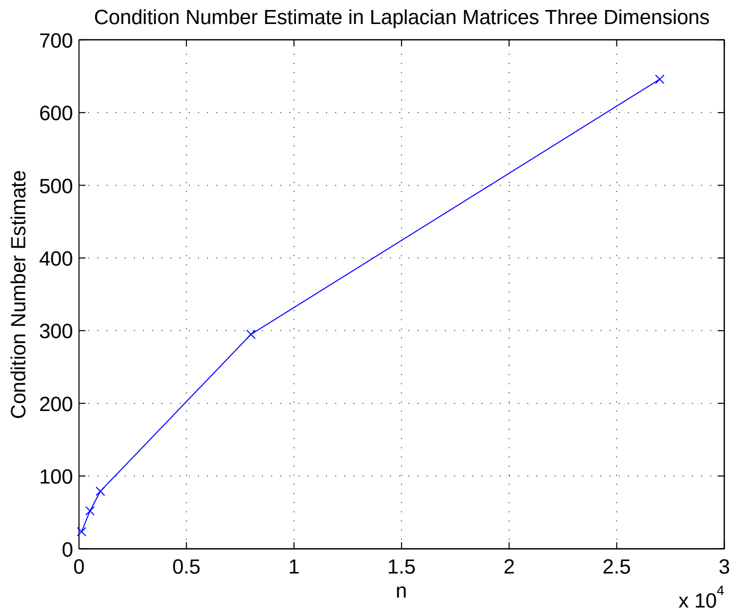

Figure 5.7: Condition Number Estimate for 3D Laplacian Matrices $n \times n$

Example. 5.3. *For $n \times n$ matrices that are from the Three-Dimensional Laplacian using a $k \times k \times k$ grid with $n = k^3$ $x^* = [1, 1, \dots, 1]^T$ and $b = Ax^*$, taking $x_0 = [0, 0 \dots, 0]$, a tolerance 1×10^{-6}, 4000 as the maximum number of iterations, and the proposed preconditioner $\tilde{P}$ (see Equation 4.1).*

$$A = \begin{pmatrix} B_{kk} & -I_{kk} & 0 & \dots & 0 \\ -I_{kk} & B_{kk} & -I_{kk} & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & -I_{kk} & B_{kk} & -I_{kk} \\ 0 & \dots & 0 & -I_{kk} & B_{kk} \end{pmatrix}_{n \times n}$$

This matrix is obtained from the discretization of the equation $-\left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2}\right) =$

$f(x_1, x_2, x_3)$ using finite differences, where the block B_{kk} is as follows

$$B_{kk} = \begin{pmatrix} B_k & -I_k & 0 & \dots & 0 \\ -I_k & B_k & -I_k & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & -I_k & B_k & -I_k \\ 0 & \dots & 0 & -I_k & B_k \end{pmatrix}_{k^2 \times k^2},$$

the block B_k is as follows

$$B_k = \begin{pmatrix} 6 & -1 & 0 & \dots & 0 \\ -1 & 6 & -1 & & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & & -1 & 6 & -1 \\ 0 & \dots & 0 & -1 & 6 \end{pmatrix}_{k \times k},$$

the block I_k is the $k \times k$ identity matrix and the block I_{kk} is the $k^2 \times k^2$ identity matrix.

Figure 5.8 shows the number of iterations versus the number of times that the matrix was iteratively preconditioned.

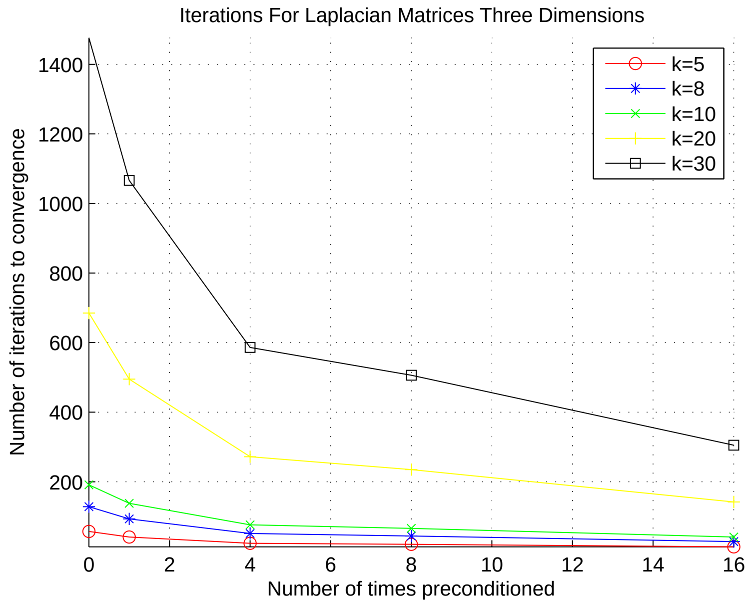

Figure 5.8: Number of iterations for 3D Laplacian matrices

	# of times preconditioned				
k	0	1	4	8	16
5	57	41	23	20	13
8	128	93	51	44	28
10	191	138	76	66	41
20	685	495	272	235	142
30	1476	1066	586	506	305

Table 5.3: Iterations For Three-Dimensional Laplacian Matrices

As in the case of example 5.1 using $\tilde{P}$ iteratively makes the Gauss-Seidel method need less iterations to converge, as shown in Table 5.3.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

- With the properties shown in Chapter 3 for the preconditioner P, an extension $\tilde{P}$ was built. This new preconditioner works for a diagonally dominant Z-matrix with positive diagonal.

- At the beginning of this thesis, it was thought that the spectral radius would have a nonzero limit as the number of times the preconditioner were applied would go to infinity. However, it was shown that this limit is zero and is reached in a finite number of steps.

- The preconditioner $\tilde{P}$ shows a significant reduction in the number of iterations of the Gauss-Seidel method once it is used iteratively.

6.2 Future Work

- The proposed preconditioner inherits the problem of fill-in of the preconditioner P, this problem could be minimized by preprocessing the array with some permutations, and comparing the results of convergence.

- If the matrix A is symmetric, PA is not symmetric. Thus, for future work, one could try to obtain a preconditioner derived from P that keeps the symmetry, and test these results with methods for symmetric matrices, such as Conjugate Gradient.

- Numerical comparisons of the preconditioner obtained in this thesis with other preconditioners for Z-matrices.

BIBLIOGRAPHY

[1] Robert Beauwens, *Iterative solution methods*, Appl. Numer. Math. **51** (2004), 437–450.

[2] Dragos M. Cvetkovic, Michael. Doob, and Horst. Sachs, *Spectra of graphs : theory and application*, Academic Press, New York ; London, 1980.

[3] Andreas Frommer and Daniel B. Szyld, *H-splittings and two-stage iterative methods*, Numerische Mathematik **63** (1992), 345–356, 10.1007/BF01385865.

[4] Gene H. Golub and Charles F. Van Loan, *Matrix computations*, 3rd ed., The Johns Hopkins University Press, 1996.

[5] Ananda D. Gunawardena, S.K. Jain, and Larry Snyder, *Modified iterative methods for consistent linear systems*, Linear Algebra and its Applications **154-156** (1991), 123 – 143.

[6] Roger A. Horn and Charles R. Johnson, *Matrix analysis*, repr. with corr. ed., Cambridge University Press, Cambridge [Cambridgeshire] ; New York, 1985 (English).

[7] Toshiyuki Kohno, Hisashi Kotakemori, Hiroshi Niki, and Masataka Usui, *Improving the modified Gauss-Seidel method for Z-matrices*, Linear Algebra and its Applications **267** (1997), 113 – 123.

[8] Toshiyuki Kohno and Hiroshi Niki, *A note on the preconditioner* $P_M = (I + S_M)$, Journal of Computational and Applied Mathematics **225** (2009), no. 1, 316 – 319.

[9] Hisashi Kotakemori, Kyouji Harada, Munenori Morimoto, and Hiroshi Niki, *A comparison theorem for the iterative method with the preconditioner* $(I + S_{max})$, Journal of Computational and Applied Mathematics **145** (2002), no. 2, 373 – 378.

[10] Hisashi Kotakemori, Hiroshi Niki, and Naotaka Okamoto, *Accelerated iterative method for Z-matrices*, Journal of Computational and Applied Mathematics **75** (1996), no. 1, 87 – 97.

[11] J. P. Milaszewicz, *Improving Jacobi and Gauss-Seidel iterations*, Linear Algebra and its Applications **93** (1987), 161 – 170.

[12] Munenori Morimoto, Kyouji Harada, Michio Sakakihara, and Hideo Sawami, *The Gauss-Seidel iterative method with the preconditioning matrix* $(I + S + S_M)$, Japan Journal of Industrial and Applied Mathematics **21** (2004), 25–34, 10.1007/BF03167430.

[13] Reinhard Nabben, *On relationships between several classes of Z-matrices, M-matrices and nonnegative matrices*, Linear Algebra and its Applications **421** (2007), no. 2-3, 417 – 439, Special Issue in honor of Miroslav Fiedler.

[14] Hiroshi Niki, Kyouji Harada, Munenori Morimoto, and Michio Sakakihara, *The survey of preconditioners used for accelerating the rate of convergence in the Gauss-Seidel method*, Journal of Computational and Applied Mathematics **164-165** (2004), 587 – 600, Proceedings of the 10th International Congress on Computational and Applied Mathematics.

[15] Y. Saad, *Iterative methods for sparse linear systems, 2nd edition*, SIAM, Philadelphia, PA, 2003.

[16] Jin-Liang Shao, Ting-Zhu Huang, and Guo-Feng Zhang, *Linear system based approach for solving some related problems of M-matrices*, Linear Algebra and its Applications **432** (2010), no. 1, 327 – 337.

[17] R.S. Varga, *Matrix iterative analysis, 2nd edition*, Springer, Berlin, 2000.