Fall 12-19-2020

Multigrid for the Nonlinear Power Flow Equations

Enrique Pereira Batista
Southern Methodist University, epereirabatista@smu.edu

MULTIGRID FOR THE NONLINEAR POWER FLOW EQUATIONS

Approved by:

Dr. Barry Lee
Associate Professor of Mathematics

Dr. Mohammad Khodayar
Associate Professor of Electrical Engineering

Dr. Benno Rumpf
Associate Professor of Mathematics

Dr. Yunkai Zhou
Associate Professor of Mathematics

MULTIGRID FOR THE NONLINEAR POWER FLOW EQUATIONS

A Dissertation Presented to the Graduate Faculty of the

Dedman College

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Computational and Applied Mathematics

by

Enrique Pereira-Batista

B.S., Universidad de Cartagena (Colombia)
M.S., Southern Methodist University

December 19, 2020

ACKNOWLEDGMENTS

I would like to acknowledge the Department of Mathematics at Southern Methodist University. Especially, I am highly grateful to my advisor Dr. Barry Lee for his continuous and invaluable support and direction in my development as a researcher. I would also like to thank Dr. Benno Rumpf, Dr. Mohammad Khodayar and Dr. Yunkai Zhou for serving on my committee and the feedback provided during the course of this work. I would like to extend my sincere gratitude to my fellow graduate student for their support and friendship. I want also to express my appreciation to Dr. Shaobu Wang and Dr. Bharat Vyakaranam for their constant support and collaboration during the time I spent at PNNL. Last but not least, I would like to thank my family and friends in Colombia and Puerto Rico, especially my parents Betty and Enrique, and Socorro Echevarria for their kindness and encouragement. Funding for this research was provided by the Division of Mathematical Sciences of the National Science Foundation under award number DMS-1734727 and DOE Advanced Scientific Computing Research under award number 4-BB47-P-00298-00.

Pereira-Batista , Enrique

B.S., Universidad de Cartagena (Colombia)
M.S., Southern Methodist University

Multigrid for the Nonlinear Power Flow Equations

Advisor: Dr. Barry Lee

Doctor of Philosophy degree conferred December 19, 2020

Dissertation completed November 2, 2020

The continuously changing structure of power systems and the inclusion of renewable energy sources are leading to changes in the dynamics of modern power grid, which have brought renewed attention to the solution of the AC power flow equations. In particular, development of fast and robust solvers for the power flow problem continues to be actively investigated. A novel multigrid technique for coarse-graining dynamic power grid models has been developed recently. This technique uses an algebraic multigrid (AMG) coarsening strategy applied to the weighted graph Laplacian that arises from the power network's topology for the construction of coarse-grain approximations to the original model. Motivated by this technique, a new multigrid method for the AC power flow equations is developed using this coarsening procedure. The AMG coarsening procedure is used to build a multilevel hierarchy of admittance matrices, which automatically leads to a hierarchy of nonlinear power flow equations. The hierarchy of power flow equations is then used in a full approximation scheme (FAS) and a multiplicative correction multigrid framework to produce multilevel solvers for the power flow equations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

This work is dedicated to my parents, Betty and Enrique.

Chapter 1

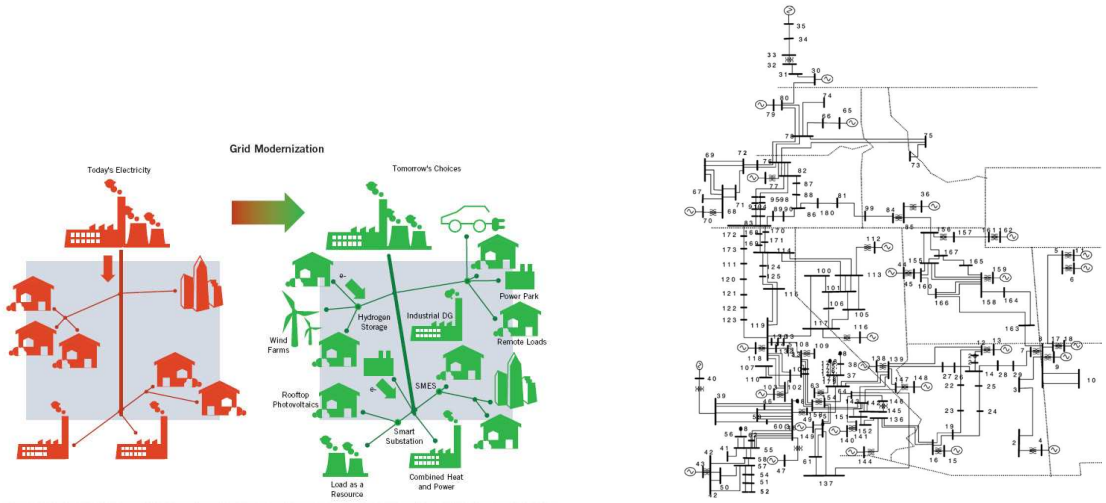
Introduction

The rapid proliferation of renewable energy generation, such as hydro, bio-gas, solar, wind and geothermal sources, leads to substantial changes in the dynamics of power grid networks. Because modern society requires large amounts of energy for use in industry, commerce, transportation, communication and domestic appliances, this consumption affects the dynamics too ([33]). This increasing demand of electric energy causes large power systems to operate under stressed conditions that reach the system's stability limits. For these reasons, it is important to know the state of the system by determining the voltages and currents at every node of the power network to monitor the system under stressed conditions. Moreover, monitoring the system can help in long-term planning designs so that the network components (generators, lines, transformers, etc.) can be appropriately constructed and stationed to withstand the stresses they will be exposed to ([33]).

1.1. Derivation of the Power Flow Equations

A power grid network is composed of r generator buses, $n - r - 1$ load buses, and a slack bus. The network can be represented as a graph denoted by $\mathcal{G}(V, E)$ where the set $V = \{1, \ldots, n\}$ represents the bus nodes and E is the set of edges representing the transmission lines connecting the buses. Behind the network, we have the system of nodal network equations describing the relationships between currents and voltages at every node in the network. It is given by

$$\mathbf{I} = \mathbf{YV}, \tag{1.1}$$



(a) Renewable energy alternatives and new elements taking part into the modern energy market

(b) Large power networks are composed of a large number of buses

Figure 1.1: Modern power grid networks

where $\mathbf{I}$ and $\mathbf{V}$ are the vectors of complex current injections and voltages at every node in the grid, respectively, and $\mathbf{Y}$ is the complex admittance matrix, which is discussed in Chapter 7.

The power injected at node i is given by

$$\mathbf{S}_i = \mathbf{P}_i + \hat{\imath}\mathbf{Q}_i = \mathbf{V}_i\overline{\mathbf{I}}_i, \tag{1.2}$$

where $\overline{(\cdot)}$ denotes complex conjugate and $\hat{\imath} = \sqrt{-1}$ is the imaginary unit. $\mathbf{P}_i$ (the real component of the injected power) is called the active power and $\mathbf{Q}_i$ (the imaginary part of the injected power) is called the reactive power. Expressing the components of the admittance matrix in terms of their real and imaginary parts

$$\mathbf{Y}_{ij} = \mathbf{G}_{ij} + \hat{\imath}\mathbf{B}_{ij}, \tag{1.3}$$

the total current injection at node i can be written as

$$\mathbf{I}_i = \sum_{j=1}^{n} \mathbf{Y}_{ij}\mathbf{V}_j = \sum_{j=1}^{n} (\mathbf{G}_{ij} + \hat{\imath}\mathbf{B}_{ij})\mathbf{V}_j. \tag{1.4}$$

Using polar coordinates for $\mathbf{V}$, i.e. $\mathbf{V}_j = |\mathbf{V}_j|\, e^{\hat{\imath}\delta_j}$, equation (1.2) can be rewritten as

$$\mathbf{P}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{B}_{ij} \sin(\delta_i - \delta_j) + \mathbf{G}_{ij} \cos(\delta_i - \delta_j) \right\} \tag{1.5}$$

$$\mathbf{Q}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{G}_{ij} \sin(\delta_i - \delta_j) - \mathbf{B}_{ij} \cos(\delta_i - \delta_j) \right\} \tag{1.6}$$

where δ_i is the voltage phase angle at the ith bus, and $\mathbf{G}_{ij}$ and $\mathbf{B}_{ij}$ are the conductance and susceptance of the transmission line connecting buses i and j. Equations (1.5)-(1.6) are the power flow equations. There are four quantities defined for every bus i: the active power $\mathbf{P}_i$, the reactive power $\mathbf{Q}_i$, the voltage magnitude $|\mathbf{V}_i|$ and the phase angle δ_i. A bus at which the active power and the voltage magnitude are specified is called a PV bus (usually a generator). A bus at which active and reactive powers are specified is called a PQ bus (usually a load bus). At the slack bus the voltage magnitude and the phase angle are known.

Since only the voltage magnitude $|\mathbf{V}_i|$ and active power $\mathbf{P}_i = Re(\mathbf{S}_i)$ are known for every PV bus i, equation (1.6) is not complete for PV buses. In order to have a complete set of equations for the PV buses, the voltage magnitudes are specified at the PV buses. Thus, a complete set of power flow equations in polar form are

$$\mathbf{P}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \{\mathbf{B}_{ij} \sin(\delta_i - \delta_j) + \mathbf{G}_{ij} \cos(\delta_i - \delta_j)\} \quad (1.7)$$

for $i = 1 \ldots, n-1$,

$$\mathbf{Q}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \{\mathbf{G}_{ij} \sin(\delta_i - \delta_j) - \mathbf{B}_{ij} \cos(\delta_i - \delta_j)\} \quad (1.8)$$

for every PQ bus i,

$$(|\mathbf{V}_i^{sp}|)^2 = |\mathbf{V}_i|^2 \quad (1.9)$$

for every PV bus i

where $|\mathbf{V}_i^{sp}|$ is the specified voltage magnitude at the PV bus i.

Typically, the resistance of a transmission line is significantly smaller in magnitude than the reactance. Therefore, the real part of the admittance matrix is often set to zero resulting in the following simplified system of equations

$$\mathbf{P}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \{\mathbf{B}_{ij} \sin(\delta_i - \delta_j)\} \quad (1.10)$$

for $i = 1, \ldots, n-1$

$$\mathbf{Q}_i = -|\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \{\mathbf{B}_{ij} \cos(\delta_i - \delta_j)\} \quad (1.11)$$

for every PQ bus i

$$(|\mathbf{V}_i^{sp}|)^2 = |\mathbf{V}_i|^2 \quad (1.12)$$

for every PV bus i.

The unknown quantities are determined by solving either nonlinear system (1.7)-(1.9) or (1.10)-(1.12).

1.2. Review of Classical Solution Methods

In general, analytic solutions for the power flow problem are not known. Thus, computational methods have to be used to calculate an approximate solution ([19, 43, 50]).

In the following, the magnitudes (moduli) of voltages, currents and impedances are expressed in per-unit or percent of specified base values. For instance, if 20kV is specified as base voltage, then 19kV corresponds to 19/20=0.95 per unit (p.u.). Calculations are made using per-unit quantities rather than dimensional quantities ([20]). Use of the per-unit system can be thought as a normalization or rescaling of the quantities involved in a power flow model.

1.2.1. Nonlinear Gauss-Seidel (Cartesian Form)

When complex voltages, power injections, and entries of the admittance matrix are represented using cartesian coordinates (i.e., real and imaginary components), the voltages can be computed iteratively by solving the system of equations

$$\mathbf{V}_i = \frac{1}{\mathbf{Y}_{ii}} \left[\frac{\mathbf{P}_i - \hat{\imath}\mathbf{Q}_i}{\overline{\mathbf{V}}_i} - \sum_{j=1,\, j\neq i}^{n} \mathbf{Y}_{ij}\mathbf{V}_j \right] \tag{1.13}$$

for $i = 1, \ldots, n-1$. These equations are obtained by using (1.1) and taking the complex conjugate of (1.2). The most simple iterative method used for solving the power flow equations is Gauss-Seidel. Starting with an initial guess $\mathbf{V}^0$, Gauss-Seidel approximates the complex voltage at node i by iterating the equation

$$\mathbf{V}_i^l = \frac{1}{\mathbf{Y}_{ii}} \left[\frac{\mathbf{P}_i - \hat{\imath}\mathbf{Q}_i}{\overline{\mathbf{V}}_i^{l-1}} - \sum_{j=1}^{i-1} \mathbf{Y}_{ij}\mathbf{V}_j^l - \sum_{j=i+1}^{n} \mathbf{Y}_{ij}\mathbf{V}_j^{l-1} \right] \tag{1.14}$$

for every $i = 1, \ldots, n-1$ and for $l = 1, 2, \ldots$

Since for a PV bus $\mathbf{Q}_i$ is unknown, it can be first approximated using (1.8) giving

$$\mathbf{Q}_i \approx |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{G}_{ij} \sin\left(\delta_i - \delta_j\right) - \mathbf{B}_{ij} \cos\left(\delta_i - \delta_j\right) \right\}. \tag{1.15}$$

Then, this value is used in (1.14) to calculate $\mathbf{V}_i^l$ ([20]).

1.2.2. Newton-Raphson

By rewriting equations (1.5)-(1.6) in the vector form

$$\mathbf{F}(\mathbf{x}) - \mathbf{y} = \mathbf{0}, \tag{1.16}$$

a different approach can be obtained. Here

$$\mathbf{x} = \begin{bmatrix} \delta \\ |\mathbf{V}| \end{bmatrix} = \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_{n-1} \\ |\mathbf{V}_1| \\ \vdots \\ |\mathbf{V}_{n-1}| \end{bmatrix}, \ \mathbf{y} = \begin{bmatrix} \mathbf{P} \\ \mathbf{Q} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_{n-1} \\ \mathbf{Q}_1 \\ \vdots \\ \mathbf{Q}_{n-1} \end{bmatrix}, \ \mathbf{F}(\mathbf{x}) = \begin{bmatrix} \mathbf{F}^1(\mathbf{x}) \\ \mathbf{F}^2(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1^1(\mathbf{x}) \\ \vdots \\ \mathbf{F}_{n-1}^1(\mathbf{x}) \\ \mathbf{F}_1^2(\mathbf{x}) \\ \vdots \\ \mathbf{F}_{n-1}^2(\mathbf{x}) \end{bmatrix}, \tag{1.17}$$

with

$$\mathbf{F}_i^1(\mathbf{x}) = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{B}_{ij} \sin(\delta_i - \delta_j) + \mathbf{G}_{ij} \cos(\delta_i - \delta_j) \right\} \tag{1.18}$$

and

$$\mathbf{F}_i^2(\mathbf{x}) = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{G}_{ij} \sin(\delta_i - \delta_j) - \mathbf{B}_{ij} \cos(\delta_i - \delta_j) \right\} \tag{1.19}$$

for $i = 1, \ldots, n-1$ (the slack bus is omitted since $|\mathbf{V}_n|$ and δ_n are specified).

Problem (1.16) can be then solved by using Newton's iterative method. The Jacobian matrix of $\mathbf{F}(\mathbf{x})$ is

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \mathbf{J}_1(\mathbf{x}) & \mathbf{J}_2(\mathbf{x}) \\ \mathbf{J}_3(\mathbf{x}) & \mathbf{J}_4(\mathbf{x}) \end{bmatrix} = \left[\begin{array}{ccc|ccc} \frac{\partial \mathbf{F}_1^1}{\partial \delta_1} & \cdots & \frac{\partial \mathbf{F}_1^1}{\partial \delta_{n-1}} & \frac{\partial \mathbf{F}_1^1}{\partial |\mathbf{V}_1|} & \cdots & \frac{\partial \mathbf{F}_1^1}{\partial |\mathbf{V}_{n-1}|} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial \mathbf{F}_{n-1}^1}{\partial \delta_1} & \cdots & \frac{\partial \mathbf{F}_{n-1}^1}{\partial \delta_{n-1}} & \frac{\partial \mathbf{F}_{n-1}^1}{\partial |\mathbf{V}_1|} & \cdots & \frac{\partial \mathbf{F}_{n-1}^1}{\partial |\mathbf{V}_{n-1}|} \\ \hline \frac{\partial \mathbf{F}_1^2}{\partial \delta_1} & \cdots & \frac{\partial \mathbf{F}_1^2}{\partial \delta_{n-1}} & \frac{\partial \mathbf{F}_1^2}{\partial |\mathbf{V}_1|} & \cdots & \frac{\partial \mathbf{F}_1^2}{\partial |\mathbf{V}_{n-1}|} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial \mathbf{F}_{n-1}^2}{\partial \delta_1} & \cdots & \frac{\partial \mathbf{F}_{n-1}^2}{\partial \delta_{n-1}} & \frac{\partial \mathbf{F}_{n-1}^2}{\partial |\mathbf{V}_1|} & \cdots & \frac{\partial \mathbf{F}_{n-1}^2}{\partial |\mathbf{V}_{n-1}|} \end{array}\right]. \tag{1.20}$$

Starting with an initial guess

$$\mathbf{x}^0 = \begin{bmatrix} \delta^0 \\ |\mathbf{V}^0| \end{bmatrix},$$

Newton's method approximates the solution of (1.16) as follows:

for $i = 0, 1, \ldots$

1. Compute

$$\Delta \mathbf{y}^i = \begin{bmatrix} \mathbf{P} - \mathbf{F}^1(\mathbf{x}^i) \\ \mathbf{Q} - \mathbf{F}^2(\mathbf{x}^i) \end{bmatrix} \tag{1.21}$$

   using (1.18) and (1.19).

2. Compute the Jacobian matrix of $\mathbf{F}$ evaluated at $\mathbf{x}^i$, denoted $\mathbf{J}\left(\mathbf{x}^i\right)$, using (1.20).

3. Solve

$$\mathbf{J}\left(\mathbf{x}^i\right) \Delta \mathbf{x}^i = \Delta \mathbf{y}^i \tag{1.22}$$

   for $\Delta \mathbf{x}^i$.

4. Update the approximate solution

$$\mathbf{x}^{i+1} = \mathbf{x}^i + \Delta \mathbf{x}^i. \tag{1.23}$$

Since for each PV bus i, $\mathbf{Q}_i$ is unknown, the equation

$$\mathbf{F}_i^2(\mathbf{x}) - \mathbf{Q}_i = 0 \tag{1.24}$$

in (1.16) is not completely defined. Furthermore, $|\mathbf{V}_i|$ is known, which implies that no processing is needed for the corresponding element in $\mathbf{x}$. Hence, $|\mathbf{V}_i|$ and $\mathbf{Q}_i$ can be dropped from $\mathbf{x}$ and $\mathbf{y}$, respectively. Moreover, with $|\mathbf{V}_i|$ dropped from $\mathbf{x}$, the column corresponding to partial derivatives with respect to $|\mathbf{V}_i|$ have to be removed from the Jacobian matrix (1.20). In addition, since equation (1.24) is not completely defined, the row corresponding to partial derivatives of $\mathbf{F}_i^2$ are also removed from the Jacobian matrix of $\mathbf{F}$. After removing $|\mathbf{V}_i|$, $\mathbf{Q}_i$ from $\mathbf{x}$ and $\mathbf{y}$, and the indicated row and column from the Jacobian matrix of $\mathbf{F}$, the size of problem (1.16) is reduced ([20]).

1.2.3. Fast Decoupled Power Flow

Two useful properties of power systems are ([21]):

1. As the transmission lines are more reactive, the conductances are relatively small compared to the susceptances, i.e. $\mathbf{G}_{ij} \ll \mathbf{B}_{ij}$.

2. Under normal steady-state operation, differences between phase angles are very small.

With these facts at hand, let us analyze $\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_i|}$, $\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_j|}$, $\frac{\partial \mathbf{F}_i^2}{\partial \delta_i}$ and $\frac{\partial \mathbf{F}_i^2}{\partial \delta_j}$ in (1.20) with $i \neq j$ for all $i, j = 1, \ldots, n-1$. These terms are given by the expressions

$$\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_i|} = 2|\mathbf{V}_i|\mathbf{G}_{ii} + \sum_{j=1,\, j\neq i}^{n} |\mathbf{V}_j| \left[\mathbf{G}_{ij}\cos(\delta_i - \delta_j) + \mathbf{B}_{ij}\sin(\delta_i - \delta_j)\right], \tag{1.25}$$

$$\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_j|} = |\mathbf{V}_i| \left[\mathbf{G}_{ij}\cos(\delta_i - \delta_j) + \mathbf{B}_{ij}\sin(\delta_i - \delta_j)\right], \tag{1.26}$$

$$\frac{\partial \mathbf{F}_i^2}{\partial \delta_i} = \sum_{j=1,\, j\neq i}^{n} |\mathbf{V}_i|\,|\mathbf{V}_j| \left[\mathbf{G}_{ij}\cos(\delta_i - \delta_j) + \mathbf{B}_{ij}\sin(\delta_i - \delta_j)\right], \tag{1.27}$$

$$\frac{\partial \mathbf{F}_i^2}{\partial \delta_j} = -|\mathbf{V}_i|\,|\mathbf{V}_j|\left[\mathbf{G}_{ij}\cos\left(\delta_i - \delta_j\right) + \mathbf{B}_{ij}\sin\left(\delta_i - \delta_j\right)\right]. \tag{1.28}$$

By the first property, $\mathbf{G}_{ii}$ and $\mathbf{G}_{ij}$ are negligible. Also, since $(\delta_i - \delta_j)$ is small, so is $\sin(\delta_i - \delta_j)$. Hence, $\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_i|} \approx 0$, $\frac{\partial \mathbf{F}_i^1}{\partial |\mathbf{V}_j|} \approx 0$, $\frac{\partial \mathbf{F}_i^2}{\partial \delta_i} \approx 0$ and $\frac{\partial \mathbf{F}_i^2}{\partial \delta_j} \approx 0$. This implies that $\mathbf{J}_2(\mathbf{x}^i) \approx \mathbf{0}$ and $\mathbf{J}_3(\mathbf{x}^i) \approx \mathbf{0}$ in (1.20), reducing (1.22) to two sets of decoupled equations

$$\mathbf{J}_1(\mathbf{x}^i)\Delta\delta^i = \mathbf{P} - \mathbf{F}^1(\mathbf{x}^i) \tag{1.29}$$

$$\mathbf{J}_4(\mathbf{x}^i)\Delta|\mathbf{V}^i| = \mathbf{Q} - \mathbf{F}^2(\mathbf{x}^i) \tag{1.30}$$

This simplified method is known as *Fast Decoupled Power Flow*.

1.2.4. DC Power Flow

Usually, the base value for the voltages in the per-unit representation is chosen in such a way that the voltage magnitudes are close to 1.0 per unit. By assuming this, under normal steady state operation, the voltage magnitudes are exactly equal to 1.0 per unit (i.e. $|\mathbf{V}_i| = 1.0$ for all $i = 1, \ldots, n$), the dependence of $\mathbf{F}^2$ on $|\mathbf{V}|$ in (1.20) can be completely neglected. With this additional simplification the real power balance equations reduce to the linear problem

$$-\mathbf{B}\delta = \mathbf{P}. \tag{1.31}$$

This technique is referred to as *DC Power Flow* ([20]).

Chapter 2

Multigrid

One of the most popular methods for solving nonlinear problems is Newton's method, described in Section 1.2.2 ([26, 46]). Unfortunately, its basin of attraction is generally small compared to other nonlinear solvers, which means it is less robust. Furthermore, for large problems such as the power flow equations on large power grids, Newton's method must solve large systems of equations, increasing its computational cost. In order to overcome these limitations, a multiscale solver will be examined. This solver will use local information of the problem to update the solution values at every node. The method has a basin of attraction that can be larger than Newton's method, and the multiscale feature of the scheme can reduce the actual cost of the solution procedure.

Multigrid (MG) is one of the most successful multiscale solvers for PDE-based problems. It is a methodology for solving an extensive class of problems by constructing a hierarchy of grids and resolving different scales of the solution on each grid of the hierarchy. Multigrid methods were first investigated by Fedorenko, who in 1964 proposed and proved the convergence of the first multigrid algorithm for the Poisson equation on a square domain ([16, 17]). In 1966, Bakhvalov extended its application to more general boundary value problems ([2]). Inspired by the papers of Fedorenko and Bakhvalov, in 1972 Brandt proved the actual efficiency of multigrid algorithms ([4]). He also studied the development of adaptive multilevel methods and introduced the nonlinear multigrid method FAS, which will be discussed in Section 2.3 ([5]). In 1976, Hackbusch analyzed the convergence of the multigrid algorithm first using the Fourier transform and later making use of the smoothing and approximation properties ([22, 23]). The effectiveness of these multigrid algorithms is based on the knowledge

of the geometry of the underlying discretization grid. This approach is known as geometric multigrid. The first investigations of a purely algebraic approach, which uses information from the matrix equation rather than the geometry of the discretization grid, were developed in the early 80's by Brandt, McCormick, Stüben and Ruge ([10, 11, 41]). This approach is known as algebraic multigrid (AMG) and is described in further details in Section 2.2. Since then, several variants of the multigrid algorithms have been developed ([49, 15, 12, 32, 13]) and applied to a vast number of problems such as image processing, combinatorial optimization, flow calculations, statistical mechanics and electrodynamics ([6, 7, 14, 27, 29]).

In this chapter, we start by reviewing geometric multigrid, which is the simplest form of multigrid. We then introduce algebraic multigrid (AMG) and the full approximation scheme (FAS). The last two will be used in the development of the multiscale solver for the power flow equations.

2.1. Geometric Multigrid

Geometric multigrid is the most basic form of multigrid. It is a very effective technique for solving linear systems

$$\mathbf{A}\mathbf{u} = \mathbf{f}, \tag{2.1}$$

where $\mathbf{A}$ is a discretization of a diffusion operator. It employs a hierarchy of grids (which is built from the initial discretization grid) in order to solve different scales of the solution. The number of computational operations employed by geometric multigrid for solving a discrete PDE problem is a small multiple of the number of unknowns of the problem. This makes geometric multigrid highly efficient for solving PDE-based problems.

In order to understand how geometric multigrid works, assume, without loss of generality, a two-level setting. Problem (2.1) and its underlying discretization grid are associated to level 1. Thus, problem (2.1) can be written as

$$\mathbf{A}^1\mathbf{u}^1 = \mathbf{f}^1. \tag{2.2}$$

At level 2, a grid and an equation defined on this grid needs to be determined, i.e. $\mathbf{A}^2$, $\mathbf{f}^2$ and operators to transfer data between grids are required. We examine how this is done by considering a particular problem described in the next section.

2.1.1. Model Problem

Consider the model 1D problem

$$\begin{aligned} -u''(x) &= f(x), \quad 0 < x < 1, && (2.3)\\ u(0) &= 0, && (2.4)\\ u(1) &= 0. && (2.5) \end{aligned}$$

A discretization of this problem using a finite difference method requires a partitioning of the domain $\Omega = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$ into n subintervals with nodes $x_i^h = ih$ for $i = 0, \ldots, n$, where $h = 1/n$. These nodes form a grid denoted by Ω^h (Figure 2.1). Equations (2.3)-(2.5) are replaced by the difference equations

$$\begin{aligned} \frac{-v_{i-1}^1 + 2v_i^1 - v_{i+1}^1}{h^2} &= f\left(x_i^h\right), \quad 1 \leq i \leq n-1, && (2.6)\\ v_0^1 &= 0, && (2.7)\\ v_n^1 &= 0. && (2.8) \end{aligned}$$

where v_i^1 is an approximation to the exact solution u at node x_i^h.

An important concept fundamental in the design of multigrid methods is smooth and oscillatory modes. To define them we start by writing (2.6)-(2.8) in matrix form

$$\mathbf{A}^1\mathbf{v}^1 = \mathbf{f}^1, \tag{2.9}$$

Figure 2.1: One-dimensional grid Ω^h defined over Ω in model problem (2.3)-(2.5).

where

$$\mathbf{A}^1 = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & \dots & \\ -1 & 2 & -1 & \dots \\ \ddots & \ddots & \ddots & \ddots \\ \dots & -1 & 2 & -1 \\ & \dots & -1 & 2 \end{bmatrix}, \quad \mathbf{v}^1 = \begin{bmatrix} v_1^1 \\ v_2^1 \\ \vdots \\ v_{n-2}^1 \\ v_{n-1}^1 \end{bmatrix} \quad \text{and} \quad \mathbf{f}^1 = \begin{bmatrix} f(x_1^h) \\ f(x_2^h) \\ \vdots \\ f(x_{n-2}^h) \\ f(x_{n-1}^h) \end{bmatrix}.$$

A relatively simple iterative method for solving (2.9) is Gauss-Seidel iteration. To explain the effect of Gauss-Seidel iteration on (2.9), consider the eigenvalues and eigenvectors of $\mathbf{A}^1$. The eigenvalues of $\mathbf{A}^1$ are given by

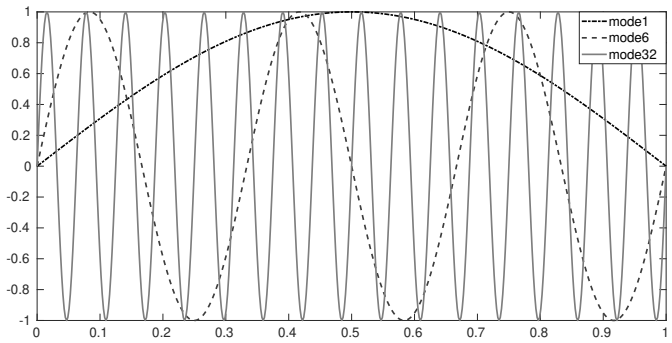
$$\lambda_j = 4\sin^2\left(\frac{j\pi}{2n}\right), \quad j = 1, \dots, n-1, \tag{2.10}$$

and the k-th component of the j-th eigenvector is

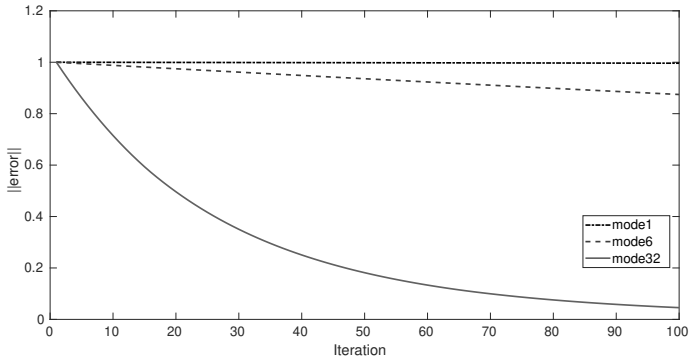
$$\mathbf{w}_{j,k} = \sin\left(\frac{jk\pi}{n}\right), \ j = 1, \dots, n-1, \ k = 0, \dots, n. \tag{2.11}$$

The vectors $\mathbf{w}_j$ are also known as *Fourier modes*. The integer j is called *wavenumber* (or *frequency*) and it indicates the number of half sine waves that constitutes $\mathbf{w}_j$ on the domain of the problem ([14]). For $n = 64$, modes $\mathbf{w}_1$, $\mathbf{w}_6$ and $\mathbf{w}_{32}$ are shown in Figure 2.2a.

These modes can be categorized as *low-oscillatory* (*smooth*) modes if $1 \leq j < \frac{n}{2}$ and *highly-oscillatory* (*high-frequency*) modes if $\frac{n}{2} \leq j \leq n-1$ ([14]). To see how Gauss-Seidel acts on the modes, consider taking $\mathbf{f}^1 = \mathbf{0}$ in (2.9) (so the exact solution is $\mathbf{u}^1 = \mathbf{0}$), and



(a) Fourier modes $\mathbf{w}_1$, $\mathbf{w}_6$ and $\mathbf{w}_{32}$ of $\mathbf{A}^1$ in (2.9) with $n = 64$.



(b) Sup-norm of the error $\mathbf{e}^1 = \mathbf{u}^1 - \mathbf{v}^1$ after 100 Gauss-Seidel sweeps with different initial guesses $\mathbf{w}_1$, $\mathbf{w}_6$, $\mathbf{w}_{32}$ and $n = 64$.

Figure 2.2: Modes $\mathbf{w}_1$, $\mathbf{w}_6$ and $\mathbf{w}_{32}$ of $\mathbf{A}^l$, with $n = 64$, and the corresponding error norms after 100 Gauss-Seidel sweeps.

a random initial guess. After several sweeps of Gauss-Seidel, an approximate solution $\mathbf{v}^1$ is obtained. It can be observed that Gauss-Seidel damps out the components of the error $\mathbf{e}^1 = \mathbf{v}^1$ corresponding to oscillatory modes very quickly, but hardly reduces the components of the error corresponding to smooth modes. Figure 2.2b shows the errors after 100 sweeps starting with initial guesses $\mathbf{v}^{1,0} = \mathbf{w}_1$, $\mathbf{v}^{1,0} = \mathbf{w}_6$ and $\mathbf{v}^{1,0} = \mathbf{w}_{32}$ to further illustrate this.

2.1.2. The Two Components of a Multigrid Scheme

From the above observations, the two main components of a multigrid scheme can be defined. The first component is the *smoother*, which corresponds to a simple iterative method (e.g. Gauss-Seidel or Jacobi iteration). It effectively reduces highly-oscillatory components of the error in an approximate solution, but fails to eliminate smooth components. We denote by $\mathfrak{S}$ the operator associated to the smoother such that its action on a vector $\mathbf{v}^1 \in \Omega^h$ is given by of the update

$$\mathbf{v}^1 \leftarrow \mathfrak{S}\left(\mathbf{A}^1, \mathbf{v}^1, \mathbf{f}^1\right). \tag{2.12}$$

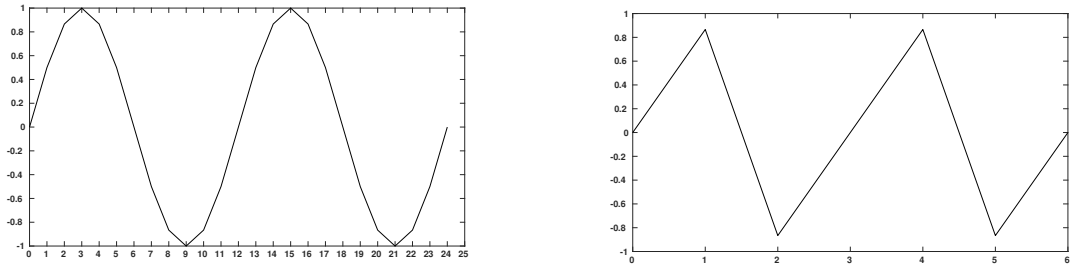
The second component is built to resolve the smooth components of the error that remain after smoothing. Its construction is guided by the coarse-grid approximation principle, which states that smooth modes on a certain grid can be well approximated on coarser grids. By considering the coarse-grid approximation principle, a coarse grid needs to be defined at level 2 where the smooth components of the error remaining after smoothing can be approximated and resolved. In geometric multigrid the definition of the coarse grid is based merely on the structure of the grid at level 1. Given the grid Ω^h at level 1, a coarse grid can be defined at level 2 by partitioning $\Omega = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$ into subintervals with grid points $x_i^H = iH$ where $H > h$. This coarse grid is denoted by Ω^H. A common choice for the coarse grid is by taking n as an even number and then setting $H = 2h$ (Figure 2.3).

Figure 2.3: Graphical representation of the one-dimensional grids Ω^h and Ω^H with $n = 8$, $H = 2h$.

Notice that, given the j-th mode on Ω^h with $1 \leq j \leq \frac{n}{2}$, its k-th component satisfies the identity

$$\mathbf{w}^1_{j,2k} = \sin\left(\frac{2kj\pi}{n}\right) = \sin\left(\frac{kj\pi}{n/2}\right) := \mathbf{w}^2_{j,k} \qquad \text{for } k = 0, \ldots, n/2. \tag{2.13}$$

Here the superscripts indicates the grids on which the modes are defined. Identity (2.13) implies that the j-th mode on Ω^h corresponds to the j-th mode on Ω^{2h}. Since $j \leq \frac{n}{2}$, this means that the smooth modes can be represented on Ω^{2h}. Because the number of modes in Ω^{2h} is half the number of modes in Ω^h, then $\mathbf{w}_j$ can be more oscillatory on Ω^{2h} (recall how the oscillatory modes are defined). This fact is illustrated in Figure 2.4.



(a) Mode $\mathbf{w}_4$ on Ω^h with $h = 1/24$. (b) Mode $\mathbf{w}_4$ on Ω^H with $H = 1/6$.

Figure 2.4: Mode $\mathbf{w}_4$, which is smooth on Ω^h with $h = 1/24$, is an oscillatory mode on Ω^{4h}.

Since the second component of a multigrid scheme involves a coarse-grid approximation of the smooth error, is it called the *coarse grid correction*. It proceeds as follows: Given an approximate solution $\mathbf{v}^1$ on Ω^h obtained by smoothing, the error $\mathbf{e}^1 = \mathbf{u}^1 - \mathbf{v}^1$ is smooth. The resulting equation for the low-oscillatory dominating error is

$$\mathbf{A}^1\mathbf{e}^1 = \mathbf{r}^1 = \mathbf{f}^1 - \mathbf{A}^1\mathbf{v}^1, \tag{2.14}$$

which is called residual equation. This equation is projected by means of a restriction operator R onto Ω^{2h} and solved there. Once the solution of the problem on Ω^{2h} is obtained, it is transferred back to Ω^h by means of an interpolation operator P and used to correct the

approximation $\mathbf{v}^1$ on Ω^h:

$$\mathbf{v}^1 \leftarrow \mathbf{v}^1 + P\mathbf{v}^2. \tag{2.15}$$

Note 2.1 *The problem on Ω^{2h}, which corresponds to a projection of equation (2.14) onto Ω^{2h} by means of R, has the form*

$$R\mathbf{A}^1\mathbf{e}^1 = R\mathbf{r}^1. \tag{2.16}$$

The interpolation operator P is commonly chosen to be piecewise linear from Ω^{2h} to Ω^h. In addition, it is assumed to have full rank (i.e. its columns are linearly independent). Notice that the range of the interpolation P consists of linear combinations of smooth modes on Ω^h. This is because only smooth modes can be represented on Ω^{2h}. Hence, given $\mathbf{v}^2 \in \Omega^{2h}$, $P\mathbf{v}^2$ can only be smooth on Ω^h. Thus, if $\mathbf{e}^2 \in \Omega^{2h}$ is a coarse-grid approximation to the smooth error $\mathbf{e}^1 \in \Omega^h$, one expects $P\mathbf{e}^2$ to accurately approximate $\mathbf{e}^1$ on Ω^h,i.e., $P\mathbf{e}^2 \approx \mathbf{e}^1$. Then, we can assume that the smooth error that remains after smoothing lies approximately in the range of P. This assumption implies that

$$\mathbf{e}^1 \approx P\mathbf{v}^2 \quad \text{for some } \mathbf{v}^2 \in \Omega^{2h}. \tag{2.17}$$

By replacing (2.17) into (2.16) one obtains

$$R\mathbf{A}^1P\mathbf{v}^2 = R\mathbf{r}^1, \tag{2.18}$$

which provides an explicit definition of $\mathbf{A}^2$: $\mathbf{A}^2 = R\mathbf{A}^1P$.

The idea used for the construction of the grids and the coarse-grid equations in the two-level setting can be generalized to a multilevel setting, i.e., given a grid Ω^{h_l} defined at level l, a coarse grid $\Omega^{h_{l+1}}$ at level $l+1$ is built doubling the step size h_l, i.e., $h_{l+1} = 2h_l$. Then, given the level l problem

$$\mathbf{A}^l\mathbf{v}^l = \mathbf{f}^l, \tag{2.19}$$

the coarse-grid problem on $\Omega^{h_{l+1}}$ is defined by the equation

$$\mathbf{A}^{l+1}\mathbf{v}^{l+1} = R_l^{l+1}\mathbf{r}^l \tag{2.20}$$

with $\mathbf{A}^{l+1} := R_l^{l+1}\mathbf{A}^l P_{l+1}^l$, which is commonly referred to as Galerkin approximation or RAP construction of the coarse-grid operator ([14]). Once $\mathbf{v}^{l+1}$ is determined, it is transferred back to level l to correct the approximation $\mathbf{v}^l$. The steps for multigrid are given in Algorithm (2.1).

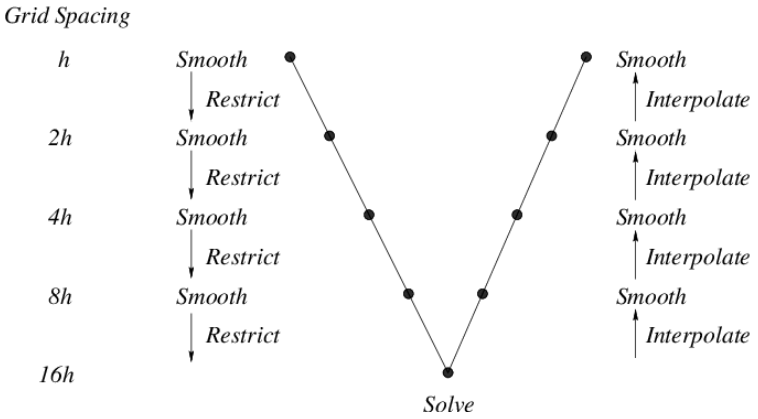


Figure 2.5: Pictorial representation of the multigrid V-Cycle.

2.1.3. Complementary Condition

One important principle to recognize in an efficient multigrid scheme is that the smoother and the coarse-grid correction work complementarily. In order to see this we first need to define the concepts of **A**-orthogonality, **A**-orthogonal projector, and two-grid error propagation operator.

Definition 2.1 *Given a symmetric positive definite matrix* $\mathbf{A} \in \mathbb{R}^{n\times n}$*, the* **A***-inner product of two vectors* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ *is defined as*

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{A}} := \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle \tag{2.21}$$

where $\langle \cdot, \cdot \rangle$ *denotes the usual Euclidean inner product on* $\mathbb{R}^n$*. The corresponding norm* $\|\cdot\|_{\mathbf{A}}$ *is defined as*

$$\|\mathbf{v}\| := \sqrt{\langle \mathbf{A}\mathbf{v}, \mathbf{v} \rangle} \tag{2.22}$$

Two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ *are said to be* $\mathbf{A}$*-orthogonal if* $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{A}} = 0$*.*

Definition 2.2 *Let* V *a vector space with an inner product* $\langle \cdot, \cdot \rangle$*, and let* W *be a nontrivial subspace of* V*. A linear operator* $\mathbf{P} : V \to W$ *is said to be a projection on* W *if* $\mathbf{P}^2\mathbf{v} = \mathbf{P}\mathbf{v}$ *for every* $\mathbf{v} \in V$*. In addition, given a symmetric positive definite matrix* $\mathbf{A}$*,* $\mathbf{P}$ *is said to be an* $\mathbf{A}$*-orthogonal projection if* $\langle \mathbf{P}\mathbf{v}, \mathbf{w} \rangle_{\mathbf{A}} = \langle \mathbf{v}, \mathbf{P}\mathbf{w} \rangle_{\mathbf{A}}$ *for all* $\mathbf{v}, \mathbf{w} \in V$*.*

Consider Algorithm 2.1 applied to problem (2.19) on a two-level scenario. Assuming no smoothing, the procedure described in Algorithm 2.1 can be written as a single operation on $\mathbf{v}^l$ as

$$\mathbf{v}^l \leftarrow \mathbf{v}^l + P_{l+1}^l \left(\mathbf{A}^{l+1}\right)^{-1} R_l^{l+1} \left(\mathbf{f}^l - \mathbf{A}^l \mathbf{v}^l\right). \tag{2.23}$$

Since the exact solution $\mathbf{u}^l$ satisfies $\mathbf{f}^l - \mathbf{A}^l\mathbf{u}^l = \mathbf{0}$, then we have

$$\mathbf{u}^l = \mathbf{u}^l + P_{l+1}^l \left(\mathbf{A}^{l+1}\right)^{-1} R_l^{l+1} \left(\mathbf{f}^l - \mathbf{A}^l \mathbf{u}^l\right). \tag{2.24}$$

After subtracting (2.23) from (2.24) we obtain

$$\mathbf{e}^l \leftarrow T^l \mathbf{e}^l \tag{2.25}$$

where $T^l = \mathbf{I} - P_{l+1}^l \left(\mathbf{A}^{l+1}\right)^{-1} R_l^{l+1} \mathbf{A}^l$, which is called the *two-grid error propagation operator.*

If $R_l^{l+1} := P_{l+1}^l$ and $\mathbf{A}^{l+1} := \left(P_{l+1}^l\right)^t \mathbf{A}^l P_{l+1}^l$, then it can be verified that the operator

$$\mathbf{I} - T^l = P_{l+1}^l \left[\left(P_{l+1}^l\right)^t \mathbf{A}^l P_{l+1}^l\right]^{-1} \left(P_{l+1}^l\right)^t \mathbf{A}^l$$

is an $\mathbf{A}^l$-orthogonal projection onto the range of P_{l+1}^l. In fact,

$$
\begin{aligned}
\left(\mathbf{I}-T^l\right)^2\mathbf{v}^l &= \left\{P_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}^2\mathbf{v}^l \quad \text{for all } \mathbf{v}^l \\
&= \left\{P_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{v}^l \\
&\quad \text{for all } \mathbf{v}^l \\
&= P_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\mathbf{v}^l \quad \text{for all } \mathbf{v}^l \\
&= \left(\mathbf{I}-T^l\right)\mathbf{v}^l \quad \text{for all } \mathbf{v}^l.
\end{aligned}
$$

This proves that $\mathbf{I}-T^l$ is a projection onto $\mathcal{R}\left(P_{l+1}^l\right)$. To see that $\mathbf{I}-T^l$ is an $\mathbf{A}^l$-orthogonal projection, note that if $\mathbf{A}^l$ is symmetric positive definite and P is full-rank, then $\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l$ is also symmetric positive definite. Thus, given $\mathbf{v}^l$ and $\mathbf{w}^l$, we have that

$$
\begin{aligned}
\left\langle\left(\mathbf{I}-T^l\right)\mathbf{v}^l,\mathbf{w}^l\right\rangle_{\mathbf{A}^l} &= \left\langle\left\{\mathbf{A}^l\left(\mathbf{I}-T^l\right)\right\}\mathbf{v}^l,\mathbf{w}^l\right\rangle \\
&= \left\langle\left\{\mathbf{A}^lP_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{v}^l,\mathbf{w}^l\right\rangle \\
&= \left\langle\left\{P_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{v}^l,\mathbf{A}^l\mathbf{w}^l\right\rangle \\
&= \left\langle\left\{\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{v}^l,\left\{\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{w}^l\right\rangle \\
&= \left\langle\left\{\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{v}^l,\left\{\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{w}^l\right\rangle \\
&= \left\langle\mathbf{A}^l\mathbf{v}^l,\left\{P_{l+1}^l\left[\left(P_{l+1}^l\right)^t\mathbf{A}^lP_{l+1}^l\right]^{-1}\left(P_{l+1}^l\right)^t\mathbf{A}^l\right\}\mathbf{w}^l\right\rangle \\
&= \left\langle\mathbf{A}^l\mathbf{v}^l,\left(\mathbf{I}-T^l\right)\mathbf{w}^l\right\rangle \\
&= \left\langle\mathbf{v}^l,\left(\mathbf{I}-T^l\right)\mathbf{w}^l\right\rangle_{\mathbf{A}^l}.
\end{aligned}
$$

The above conditions imply that the range of P_{l+1}^l lies in the nullspace of T^l ([14]). Indeed, if $\mathbf{v}^l$ is in the range of P_{l+1}^l, then $\mathbf{v}^l$ has the form $P_{l+1}^l\mathbf{v}^{l+1}$ for some vector $\mathbf{v}^{l+1}$. Thus,

$$
\begin{aligned}
T\mathbf{v}^l &= TP_{l+1}^l \mathbf{v}^{l+1} \\
&= \left(\mathbf{I} - P_{l+1}^l \left(\mathbf{A}^{l+1}\right)^{-1} R_l^{l+1} \mathbf{A}^l\right) \mathbf{v}^{l+1} \\
&= P\mathbf{v}^{l+1} - P\mathbf{v}^{l+1} \\
&= \mathbf{0}.
\end{aligned}
$$

Hence, if the smooth modes are approximated accurately by the interpolation, the smooth error will be effectively reduced by the coarse-grid correction. To this end, the interpolation must be built in such a way that its range, $\mathcal{R}\left(P_{l+1}^l\right)$, approximates the null-space of $\mathbf{A}^l$, $\mathcal{N}\left(\mathbf{A}^l\right)$; i.e., the relationship

$$
\mathcal{R}\left(P_{l+1}^l\right) \approx \mathcal{N}\left(\mathbf{A}^l\right) \tag{2.26}
$$

is required for an effective MG algorithm.

2.2. Algebraic Multigrid (AMG)

When the matrix $\mathbf{A}$ in equation (2.1) does not arise from a PDE discretization and/or the grid structure is irregular, geometric multigrid may no longer be applicable since there may not be a geometric grid available ([45]). A variation of multigrid that requires only the matrix equation is known as *algebraic multigrid* (AMG). While geometric multigrid approach is based on the structure of the discrete grid, an AMG approach is based only on the coefficients of $\mathbf{A}$. Thus, AMG is more appropriate for solving the power flow equations since these equations are defined on unstructured graphs. However, unlike geometric multigrid, AMG requires the selection of the coarse DOFs and intricate construction of the restriction and interpolation operators.

Assume a hierarchy of levels. At the level l the problem is

$$
\mathbf{A}^l \mathbf{v}^l = \mathbf{f}^l \tag{2.27}
$$

Algorithm 2.1 MG V(ν_1, ν_2)-Cycle

1: MG($\mathbf{A}^l, \mathbf{f}^l, \mathbf{v}^l, l, \nu_1, \nu_2$)
2: **if** $l = L$ **then**
3: Solve $\mathbf{A}^l\mathbf{v}^l = \mathbf{f}^l$.
4: **else**
5: $\mathbf{v}^l \leftarrow \mathfrak{S}\left(\mathbf{A}^l, \mathbf{v}^l, \mathbf{f}^l\right)$ ν_1 times.
6: $\mathbf{r}^l \leftarrow \mathbf{f}^l - \mathbf{A}^l\mathbf{v}^l$
7: $\mathbf{f}^{l+1} \leftarrow \left(R_l^{l+1}\right)\mathbf{r}^l,\ \mathbf{v}^{l+1} \leftarrow 0$
8: $\mathbf{v}^{l+1} \leftarrow$ MG($\mathbf{A}^{l+1}, \mathbf{f}^{l+1}, \mathbf{v}^{l+1}, l+1, \nu_1, \nu_2$)
9: $\mathbf{v}^l \leftarrow \mathbf{v}^l + \left(P_{l+1}^l\right)\mathbf{v}^{l+1}$
10: $\mathbf{v}^l \leftarrow \mathfrak{S}\left(\mathbf{A}^l, \mathbf{v}^l, \mathbf{f}^l\right)$ ν_2 times.
11: **end if**

Figure 2.6: Multigrid V-cycle algorithm for solving the discrete PDE-problem $\mathbf{Au} = \mathbf{f}$.

where $\mathbf{A}^l$ is a symmetric M-matrix; i.e., a symmetric positive definite matrix with positive entries along its main diagonal and nonpositive off-diagonal entries ([42]). As in geometric multigrid, we seek to define a coarse-grid problem at level $l+1$ and operators that transfer vectors between levels l and $l+1$. In geometric multigrid the coarse grid, and the interpolation and restriction operators are determined based only on the geometric information. Since AMG approach does not consider any geometric information from level l, defining a coarse grid at level $l+1$ and building the interpolation and restriction operators are not obvious.

2.2.1. Strong Influence

To determine the coarse DOFs that form the grid at level $l+1$, consider the concept of how strongly one variable at level l affects the other variables at the same level. This concept is illustrated by Example 2.1.

Example 2.1 *Consider the operator* $\mathbf{A}^l \in \mathbb{R}^{3\times 3}$ *defined by*

$$\mathbf{A}^l = \begin{bmatrix} 1 & -0.8 & -0.001 \\ -0.8 & 1 & -0.01 \\ -0.001 & -0.01 & 1 \end{bmatrix}.$$

By observing the components of $\mathbf{A}^l$*, it can be seen that changes in variable 2 will dramatically affect variable 1. On the other hand, changes in variable 3 will not strongly affect variable 1. Following the same reasoning, it can be also verified that variable 2 will be strongly affected by changes in variable 1; and variable 3 will be strongly affected by changes in variable 2, but not by changes in variable 1.*

Example 2.1 illustrates the concept of *variable influence*.

Definition 2.3 *Given a value* $\theta,\ 0 < \theta \leq 1$*, and nodes* i *and* j *at level* l*, node* j *is said to strongly influence node* i *if*

$$\left|\mathbf{A}^l_{ij}\right| \geq \theta\, max_{k\neq i}\left\{\left|\mathbf{A}^l_{ik}\right|\right\}. \tag{2.28}$$

The relevance of Definition 2.3 for the coarsening procedure can be seen by noticing that given the i-th equation (i.e., the equation associated to node i) in system (2.27), if node j strongly influences node i, then $\mathbf{A}^l_{ij}$ is large compared to other off-diagonal entries $\mathbf{A}^l_{ik}$ ($k \neq j$, $k \neq i$) in the i-th row of $\mathbf{A}^l$. Thus, any change in $\mathbf{v}^l_j$ will affect $\mathbf{v}^l_i$ more than changes in other component $\mathbf{v}^l_k$ of $\mathbf{v}^l$. Because $\mathbf{v}^l_j$ is significant in determining the value of $\mathbf{v}^l_i$, it makes sense to consider node $\mathbf{v}^l_j$ in the interpolation of $\mathbf{v}^l_i$. This makes node j a candidate for the set of coarse nodes at level $l+1$ ([14]). The nodes that strongly influence most nodes are good candidates for the set of coarse nodes.

2.2.2. Construction of the Coarse Grid and the Interpolation Operator

In addition to the definition of strong influence, a key concept for the construction of the coarse grid and the interpolation operator is *algebraic smoothness*. First, assume that

the smoother has been already defined. When no geometric information is available, it is not possible to determine the smooth components of the error $\mathbf{e}^l$ geometrically. Instead, we recall from Section 2.1.1 that smooth errors are hardly reduced by the smoother. Thus, the algebraically smooth error is defined as the error that does not decrease significantly in magnitude after applying successive smoothing iterations to (2.27).

With the help of the concepts of strong influence and algebraic smoothness, the construction of the coarse grid at level $l+1$ and the interpolation operator from level $l+1$ to level l can be described. First, the set of nodes at level l are partitioned into two sets C and F with $C \cap F = \emptyset$. The following definitions on the grid at level l are required ([41]):

- N_i, the neighborhood of node i, is the set of all nodes $j \neq i$ such that $\mathbf{A}^l_{ij} \neq 0$.
- C_i, the interpolatory set for i, is the neighboring nodes of i in C that strongly influence node i.
- D^s_i, the subset of neighboring nodes in F that strongly influence node i.
- D^w_i, the set of neighboring nodes in F that do not strongly influence node i.
- S_i, the set of nodes in $C \cup F$ that strongly influence node i.
- S^t_i, the set of nodes j strongly influenced by node i.

The elements of C, which are the nodes to form the grid at level $l+1$, are selected by enforcing the following two criteria ([41]):

- **Cr1**: For each node $i \in F$, every node that strongly influences node i either should be in C_i or should be influenced by at least one node in C_i.
- **Cr2**: C should be a maximal subset such that there is no node in C strongly influencing another node in C.

The selection of coarse-grid nodes (C-nodes) is done in a procedure known as coloring scheme ([14]), which is applied in two stages. A first set of C-nodes is formed such that it tends to

satisfy **Cr2**. Then, a second step is performed on the remaining F-nodes to enforce **Cr1**. In the first stage, every node i is initially assigned a weight that measures its potential quality as a C-point. A usual choice for this weight is the number of nodes that i strongly influences (i.e, the cardinality of S_i^t), which is denoted by λ_i. After all the weights have been assigned, the node with the largest λ_i is selected as a C-node. Since the nodes in S_i^t are already influenced by a C-node, they should become F-nodes in order to fulfill **Cr2** (otherwise, there would be C-nodes being strongly influenced by other C-nodes). All other nodes strongly influencing every node $j \in S_i^t$ can be useful for improving the accuracy of the interpolation of j. Thus, it makes sense to consider them as prospective C-nodes. Hence, for each node $j \in S_i^t$, the value λ_k is incremented for every $k \in S_j$ that has not been assigned as C- or F-node. The process is repeated for a new node that has not been assigned as C- or F-node until all the nodes are assigned to C or F. After this first step, it may occur that **Cr1** is not satisfied, i.e., there is a pair of F-nodes with one of them strongly influencing the other, but not being strongly connected to a common C-node. Therefore, a second step is necessary to ensure that **Cr1** is fulfilled. In this step, one checks for any nodes where **Cr1** is violated. If this occurs, then **Cr1** is enforced by converting some F-nodes into C-nodes. Example 2.2 illustrates how the coloring scheme works. In the following, we assume that the nodes of the grid at level l have been already partitioned into the sets C and F, and the C-nodes have been already determined.

Example 2.2 *Consider the two-dimensional Poisson equation*

$$
\begin{aligned}
-u_{xx} - u_{yy} &= f(x,y), \quad 0 < x < 1, \quad 0 < y < 1, && (2.29)\\
u(\tilde{x}, y) &= 0 \quad \textit{if } \tilde{x} = 0 \textit{ or } \tilde{x} = 1, && (2.30)\\
u(x, \tilde{y}) &= 0 \quad \textit{if } \tilde{y} = 0 \textit{ or } \tilde{y} = 1 && (2.31)
\end{aligned}
$$

defined on the unit square $[0,1] \times [0,1]$. *Given the positive integers* m *and* n, *a grid can be defined on this domain with grid points* $(x_i, y_j) := (ih_x, jh_y)$ *for every* $0 \leq i \leq m$ *and* $0 \leq j \leq n$ *with* $h_x = 1/m$ *and* $h_y = 1/n$. *Here we choose* $m = n = 6$. *By using second-order finite differences, problem (2.29)-(2.31) can be replaced by the linear system*

$$\frac{-v_{i-1,j} + 2v_{i,j} - v_{i+1,j}}{h_x^2} + \frac{-v_{i,j-1} + 2v_{i,j} - v_{i,j+1}}{h_y^2} = f_{i,y} \qquad (2.32)$$

$$1 < i < 5, \quad 1 < j < 5,$$

$$v_{i,j} = 0 \quad \text{if } i = 0 \text{ or } i = 6, \qquad (2.33)$$

$$v_{i,j} = 0 \quad \text{if } j = 0 \text{ or } j = 6 \qquad (2.34)$$

where $v_{i,j}$ *is an approximation to the exact solution* u *at* (x_i, y_j) *and* $f_{i,j} = f(x_i, y_j)$. *The number of unknowns is* $(m-1)(n-1) = 25$. *Thus, the system (2.32)-(2.34) can be rewritten in matrix form as*

$$\mathbf{A}\mathbf{v} = \mathbf{f}$$

where $\mathbf{v}, \mathbf{f} \in \mathbb{R}^{25}$ *are the vectors of unknowns and right-hand side values, respectively, which are organized lexicographically, i.e.* $\mathbf{v} = (v_{1,1}, v_{1,2}, \ldots, v_{1,5}, v_{2,1}, \ldots, v_{5,5})^t$ *and* $\mathbf{f} = (f_{1,1}, f_{1,2}, \ldots, f_{1,5}, f_{2,1}, \ldots, f_{5,5})^t$. *The matrix* $\mathbf{A}$ *is a block matrix with the form*

$$\mathbf{A} = \frac{1}{36} \begin{bmatrix} \mathbf{B} & -\mathbf{I} & & & \\ -\mathbf{I} & \mathbf{B} & -\mathbf{I} & & \\ & -\mathbf{I} & \mathbf{B} & -\mathbf{I} & \\ & & -\mathbf{I} & \mathbf{B} & -\mathbf{I} \\ & & & -\mathbf{I} & \mathbf{B} \end{bmatrix}$$

where

$$\mathbf{B} = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & -1 & 4 & -1 & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{bmatrix}$$

and $\mathbf{I}$ *is the* 5×5 *identity matrix.*

The grid corresponding to the graph connection is shown on the upper left corner of Figure 2.7. Initially, all the nodes are labeled with 'X', indicating that they have not been yet selected as C- or F-nodes. Nodes that have been selected as C-nodes (or F-nodes) are labeled with C (or F). New F-nodes and their undetermined neighbors are labeled in bold. For an arbitrary $0 < \theta < 1$*, every node is strongly influenced by and strongly influences its left, right, upper and lower neighbors. At the beginning, the nodes at the interior of the grid have weight* $\lambda = 4$*, the nodes along the sides have weight* $\lambda = 3$ *and the nodes at the corners have weight* $\lambda = 2$*. The steps of the coloring scheme algorithm are shown in Figure 2.7. Since* **Cr1** *is already satisfied in this case, there is no need of a second step.*

As in the geometric case, the interpolation operator P_{l+1}^l in AMG has to be built such that for an algebraically smooth error $\mathbf{e}^l$, $P_{l+1}^l \mathbf{e}^{l+1}$ must accurately approximates $\mathbf{e}^l$ for some $\mathbf{e}^{l+1}$. For an algebraically smooth error $\mathbf{e}^l$ we have

$$\mathbf{0} \approx \mathbf{A}^l \mathbf{e}^l = \mathbf{r}^l. \tag{2.35}$$

This shows that the residual is relatively small. The i-th component of equation (2.35) can be written as

$$\mathbf{A}_{ii}^l \mathbf{e}_i^l \approx -\sum_{j \in N_i} \mathbf{A}_{ij}^l \mathbf{e}_j^l. \tag{2.36}$$

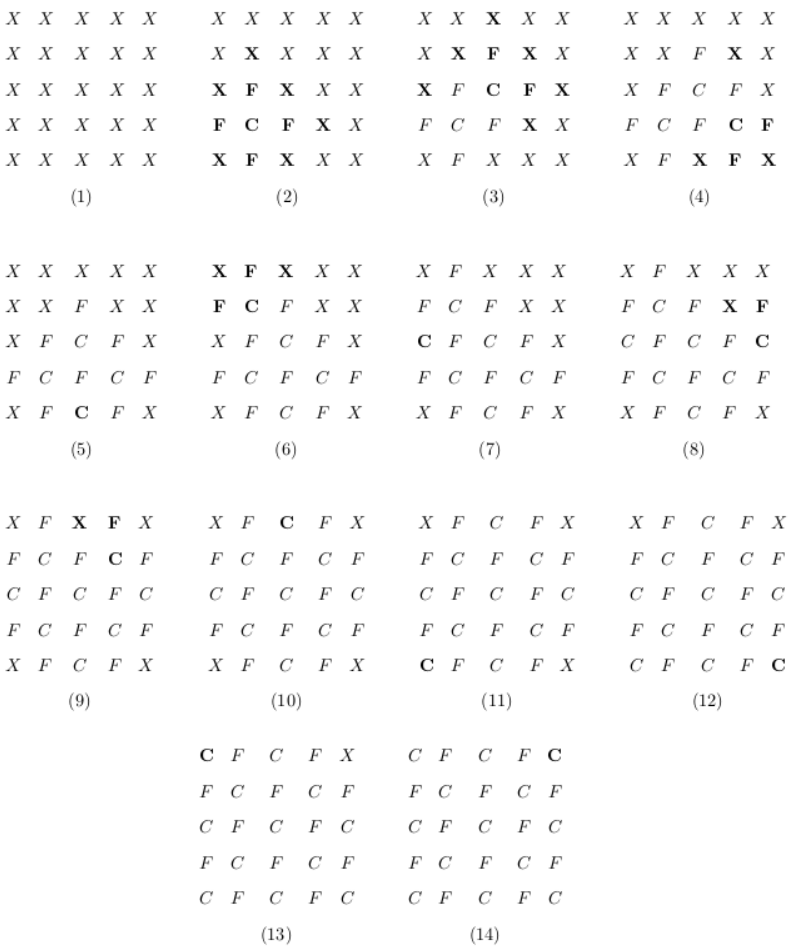


Figure 2.7: Coloring scheme steps for the graph corresponding to the discrete problem (2.32)-(2.34) with a five-point stencil.

Splitting the right-hand side of equation (2.36) into sums over the sets C_i, D_i^s and D_i^w yields

$$\mathbf{A}_{ii}^l \mathbf{e}_i^l \approx -\sum_{j \in C_i} \mathbf{A}_{ij}^l \mathbf{e}_j^l - \sum_{j \in D_i^s} \mathbf{A}_{ij}^l \mathbf{e}_j^l - \sum_{j \in D_i^w} \mathbf{A}_{ij}^l \mathbf{e}_j^l. \tag{2.37}$$

Now, assume that the interpolation operator from level $l+1$ to level l has the form

$$\left(P_{l+1}^l \mathbf{e}^l\right)_i = \begin{cases} \mathbf{e}_i^l & \text{if } i \in C \\ \sum_{j \in C_i} \omega_{ij} \mathbf{e}_j^l & \text{if } i \in F \end{cases}.$$

In order to determine ω_{ij} for every $j \in C_i$, the second and third summations on the right-

hand side in (2.37) need to be expressed in terms of $\mathbf{e}_i^l$ and $\mathbf{e}_j^l$, where $j \in C_i$. Since nodes $j \in D_i^w$ are weakly connected to node i, the coefficients $\mathbf{A}_{ij}^l$ in the third summation on the right-hand side of (2.37) are relatively small and can be lumped into the diagonal coefficient $\mathbf{A}_{ii}^l$ without causing any significant change. This gives us

$$\left(\mathbf{A}_{ii}^l + \sum_{j \in D_i^w} \mathbf{A}_{ij}^l\right) \mathbf{e}_i^l \approx -\sum_{j \in C_i} \mathbf{A}_{ij}^l \mathbf{e}_j^l - \sum_{j \in D_i^s} \mathbf{A}_{ij}^l \mathbf{e}_j^l. \tag{2.38}$$

Every error component $\mathbf{e}_j^l$ in the second summation on the right-hand side of (2.37) can be approximated by a weighted summation over C_i in the form

$$\mathbf{e}_j^l \approx \frac{\sum_{k \in C_i} \mathbf{A}_{jk}^l \mathbf{e}_k^l}{\sum_{k \in C_i} \mathbf{A}_{jk}^l}. \tag{2.39}$$

Since $\mathbf{A}^l$ is a symmetric M-matrix, equation (2.39) is well defined. Replacing equation (2.39) into equation (2.38) yields

$$\left(\mathbf{A}_{ii}^l + \sum_{j \in D_i^w} \mathbf{A}_{ij}^l\right) \mathbf{e}_i^l \approx \sum_{j \in C_i} \left[\mathbf{A}_{ij}^l + \sum_{m \in D_i^s} \left(\frac{\mathbf{A}_{im}^l \mathbf{A}_{mj}^l}{\sum_{k \in C_i} \mathbf{A}_{mk}^l}\right)\right] \mathbf{e}_j^l. \tag{2.40}$$

Therefore, the weights ω_{ij} are given by

$$\omega_{ij} = -\frac{\mathbf{A}_{ij}^l + \sum_{m \in D_i^s} \left(\frac{\mathbf{A}_{im}^l \mathbf{A}_{mj}^l}{\sum_{k \in C_i} \mathbf{A}_{mk}^l}\right)}{\mathbf{A}_{ii}^l + \sum_{n \in D_i^w} \mathbf{A}_{in}^l}. \tag{2.41}$$

With the weights w_{ij} determined, the interpolation P_{l+1}^l is obtained. The restriction operator is then defined as $R_l^{l+1} := \left(P_{l+1}^l\right)^t$ and the AMG V-cycle scheme can be described as in the geometric multigrid.

2.3. Full Approximation Scheme (FAS)

Since the power flow equations are nonlinear and AMG is designed for solving linear problems, a new multigrid approach is needed that is suitable for solving the nonlinear

problem

$$\mathfrak{N}(\mathbf{u}) = \mathbf{f} \tag{2.42}$$

where $\mathfrak{N}$ is a nonlinear operator acting on $\mathbf{u}$. Again, assume a multigrid hierarchy is available. With the problem

$$\mathfrak{N}^l\left(\mathbf{u}^l\right) = \mathbf{f}^l \tag{2.43}$$

defined on the grid at level l, we seek to derive an analogous problem at level $l+1$ based on equation (2.43). We start by noticing that the residual equation (2.14) no longer holds ([25]), i.e. if $\mathbf{v}^l$ is an approximation to the solution $\mathbf{u}^l$ of (2.43), then

$$\mathfrak{N}^l\left(\mathbf{u}^l\right) - \mathfrak{N}^l\left(\mathbf{v}^l\right) \neq \mathfrak{N}^l\left(\mathbf{u}^l - \mathbf{v}^l\right) = \mathfrak{N}^l\left(\mathbf{e}^l\right). \tag{2.44}$$

Thus, we consider the defect equation

$$\mathfrak{N}^l\left(\mathbf{u}^l\right) - \mathfrak{N}^l\left(\mathbf{v}^l\right) = \mathbf{d}^l. \tag{2.45}$$

If $\mathbf{u}^l$ is replaced by $\mathbf{v}^l + \mathbf{e}^l$ in (2.45), where $\mathbf{e}^l = \mathbf{u}^l - \mathbf{v}^l$ is the approximation error, then the defect equation (2.45) can be rewritten as

$$\mathfrak{N}^l\left(\mathbf{v}^l + \mathbf{e}^l\right) - \mathfrak{N}^l\left(\mathbf{v}^l\right) = \mathbf{d}^l. \tag{2.46}$$

Accordingly, equation (2.46) at level $l+1$ should have the form

$$\mathfrak{N}^{l+1}\left(\mathbf{v}^{l+1} + \mathbf{e}^{l+1}\right) - \mathfrak{N}^{l+1}\left(\mathbf{v}^{l+1}\right) = \mathbf{d}^{l+1} \tag{2.47}$$

where $\mathfrak{N}^{l+1}$ denotes the level $l+1$ coarse-grid operator, $\mathbf{d}^{l+1}$, $\mathbf{v}^{l+1}$ and $\mathbf{e}^{l+1}$ are coarse approximations to $\mathbf{d}^l$, $\mathbf{v}^l$ and $\mathbf{e}^l$, respectively. The coarse defect $\mathbf{d}^{l+1}$ is simply chosen as the projection of the fine-grid defect $\mathbf{d}^l$ onto the grid at level $l+1$ by means of a restriction operator R_l^{l+1}, i.e.

$$\begin{aligned}\mathbf{d}^{l+1} &= R_l^{l+1}\mathbf{d}^l = R_l^{l+1}\left[\mathfrak{N}^l\left(\mathbf{v}^l+\mathbf{e}^l\right)-\mathfrak{N}^l\left(\mathbf{v}^l\right)\right] \\ &= R_l^{l+1}\left[\mathfrak{N}^l\left(\mathbf{u}^l\right)-\mathfrak{N}^l\left(\mathbf{v}^l\right)\right] \\ &= R_l^{l+1}\left[\mathbf{f}^l-\mathfrak{N}^l\left(\mathbf{v}^l\right)\right]. \end{aligned} \tag{2.48}$$

As for the coarse approximation $\mathbf{v}^{l+1}$, it is often computed by restricting $\mathbf{v}^l$ to the coarse grid through a transfer operator $\widehat{R}_l^{l+1}$ which can be different from R_l^{l+1}. After substituting this restriction into (2.47), it gives us

$$\mathfrak{N}^{l+1}\left(\widehat{R}_l^{l+1}\mathbf{v}^l+\mathbf{e}^{l+1}\right)-\mathfrak{N}^{l+1}\left(\widehat{R}_l^{l+1}\mathbf{v}^l\right)=R_l^{l+1}\left[\mathbf{f}^l-\mathfrak{N}^l\left(\mathbf{v}^l\right)\right], \tag{2.49}$$

or more conveniently

$$\mathfrak{N}^{l+1}\left(\widehat{R}_l^{l+1}\mathbf{v}^l+\mathbf{e}^{l+1}\right)=\mathfrak{N}^{l+1}\left(\widehat{R}_l^{l+1}\mathbf{v}^l\right)+R_l^{l+1}\left[\mathbf{f}^l-\mathfrak{N}^l\left(\mathbf{v}^l\right)\right]. \tag{2.50}$$

Assuming that the right-hand side of equation (2.50) is already computed, the goal is to compute or approximate the solution $\mathbf{u}^{l+1}=\widehat{R}_l^{l+1}\mathbf{v}^l+\mathbf{e}^{l+1}$ to this system. The approximate error $\mathbf{e}^{l+1}=\mathbf{u}^{l+1}-\widehat{R}_l^{l+1}\mathbf{v}^l$ is then interpolated to level l and used to correct the approximate solution $\mathbf{v}^l$.

The process outlined above is summarized in Algorithm 2.2 and called the *Full Approximation Scheme* (FAS) because it resolves (2.50) for $\mathbf{u}^{l+1}$ rather than for the coarse error $\mathbf{e}^{l+1}$ ([5, 6]). It employs a nonlinear relaxation method (i.e., nonlinear Jacobi or Gauss-Seidel iteration) as the smoother. In contrast to Newton's method, which uses only the target problem in the solution process, FAS reduces the computational cost by using a hierarchy of grids. In addition, FAS can have a larger basin of attraction than Newton's method and handles the nonlinearities locally at every node ([38, 47, 51]).

Algorithm 2.2 FAS V(ν_1, ν_2)-cycle

FAS$(\mathfrak{N}^l, \mathbf{f}^l, \mathbf{v}^l, l, \nu_1, \nu_2)$

if $l = L$ **then**

    Solve $\mathfrak{N}^l\left(\mathbf{v}^l\right) = \mathbf{f}^l$ using Newton's method.

else

    $\mathbf{v}^l \leftarrow$ Smooth $(\mathfrak{N}^l, \mathbf{v}^l, \mathbf{f}^l)$ ν_1 times.

    $\mathbf{d}^l \leftarrow \mathbf{f}^l - \mathfrak{N}^l\left(\mathbf{v}^l\right)$.

    $\mathbf{d}^{l+1} \leftarrow R_l^{l+1}\mathbf{d}^l$.

    $\mathbf{v}^{l+1} \leftarrow \widehat{R}_l^{l+1}\mathbf{v}^l$

    $\mathbf{f}^{l+1} \leftarrow \mathbf{d}^{l+1} + \mathfrak{N}^{l+1}\left(\mathbf{v}^{l+1}\right)$.

    $\mathbf{w}^{l+1} \leftarrow$ FAS$(\mathfrak{N}^{l+1}, \mathbf{f}^{l+1}, \mathbf{v}^{l+1}, l+1, \nu_1, \nu_2)$.

    $\mathbf{e}^{l+1} \leftarrow \mathbf{w}^{l+1} - \mathbf{v}^{l+1}$

    $\mathbf{v}^l \leftarrow \mathbf{v}^l + P_{l+1}^l\mathbf{e}^{l+1}$

    $\mathbf{v}^l \leftarrow$ Smooth $(\mathfrak{N}^l, \mathbf{v}^l, \mathbf{f}^l)$ ν_2 times.

end if

Chapter 3

Affinity-Based Coarsening

Unfortunately, the FAS approach described in Section 2.3 by itself cannot be applied directly to the power flow equations since the coarse-grid problems are obviously not available. Furthermore, coarsening based on strong influence given in Section 2.2 is not effective for matrices that do not satisfy the hypothesis of standard AMG (i.e., satisfying a symmetric M-matrix property) and can lead to an incorrect coarse-node selection when applied to a power grid system [8, 40]. Thus, a different coarsening strategy is required for the power flow equations.

3.1. Coarsening of Simplified Real Power Flow Problems

As noted earlier, in standard AMG, the coarsening procedure is based on the concepts of *algebraically smooth error* and *strong influence*. These concepts are used to detect strong connections between nodes in a linear system. To "measure the strength of connection" between nodes in a power grid model, the coefficients of the graph Laplacian associated with the admittance matrix are considered ([31, 40]). A new coarsening procedure is described by considering first the simplified real power flow problem

$$\mathbf{P}_i = |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{B}_{ij} \sin\left(\delta_i - \delta_j\right) \right\} \tag{3.1}$$

$$\mathbf{Q}_i = -|\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{B}_{ij} \cos\left(\delta_i - \delta_j\right) \right\}. \tag{3.2}$$

which involves merely an admittance matrix with real-valued coefficients. For this problem the graph Laplacian $\mathbf{L}_f$ is defined as

$$\mathbf{L}_{f,\,ij} = \begin{cases} |\mathbf{V}_i|\,|\mathbf{V}_j|\mathbf{B}_{ij} & i \neq j \\ -|\mathbf{V}_i| \sum_{k \neq i} |\mathbf{V}_k|\mathbf{B}_{ik} & i = j \end{cases} . \tag{3.3}$$

In order to describe the coarsening procedure, note that the algebraically smooth error $\mathbf{e}$ needs to be accurately represented at the coarse level. As mentioned in Section 2.2.2, the algebraically smooth error $\mathbf{e}$ is poorly reduced after successive smoothing iterations, and satisfies the condition $\mathbf{L}_f\mathbf{e} \approx \mathbf{0}$. Hence, since this error is unknown, the smoother itself can be used to expose it by relaxing the problem $\mathbf{L}_f\mathbf{x} = \mathbf{0}$ ([8, 9, 40]). Here, we start with a random initial guess $\mathbf{x}^0$,and the right-hand side is set to zero to prevent any dependence of the solution on the right-hand side.

For simplicity, let us consider a two-level setting and assume that problem (3.1)-(3.2) is already defined on a grid (fine grid) at level 1. The set of neighboring nodes of node i in the fine grid is denoted by $N(i)$. The selection of the nodes that form the grid (coarse grid) at level 2 and the construction of the interpolation and coarse operators are carried out as follow:

1. Taking a set of K random test vectors $\left\{\mathbf{x}^{(k)}\right\}_{k=1}^{K}$ at the fine level and applying s relaxation sweeps to the linear system $\mathbf{L}_f\mathbf{x}^{(k)} = \mathbf{0}$, a set of smoothed test vectors is obtained that exposes the profile of the smooth error.

2. The *affinity measure* c_{ij} between nodes i and j is defined as ([31])

$$c_{ij} = \frac{|(\mathbf{x}_i, \mathbf{x}_j)|^2}{(\mathbf{x}_i, \mathbf{x}_i)(\mathbf{x}_j, \mathbf{x}_j)} \quad \text{with} \quad (\mathbf{x}_i, \mathbf{x}_j) := \sum_{k=1}^{K} \mathbf{x}_i^{(k)}\mathbf{x}_j^{(k)}. \tag{3.4}$$

   We have $0 \leq c_{ij} \leq 1$ and $c_{ij} = c_{ji}$.

3. The set of nodes in the fine grid is partitioned into two sets C and F, initially with F containing all the fine-grid nodes. Then

(i) For each node i in the fine grid, let v_i be its volume, which is initially set to 1. The projected volume for node i is defined as ([40])

$$\nu_i = v_i + \sum_{(i,j)\in E} v_j \frac{c_{ij}}{\sum_{(j,k)\in E} c_{jk}}. \tag{3.5}$$

All nodes having projected volume greater than some factor σ times the average projected volume are moved to C.

(ii) Given a threshold Q, if either of the following condition holds for an F-node i ([30])

$$\frac{\sum_{j\in C\cap N(i)} c_{ij}}{\sum_{j\in N(i)} c_{ij}} \le Q \quad \text{or} \quad \frac{\sum_{j\in C\cap N(i)} |\mathbf{L}_{f,ij}|}{\sum_{j\in N(i)} |\mathbf{L}_{f,ij}|} \le Q, \tag{3.6}$$

then node i is moved to C.

4. The caliber l interpolation operator P is computed as

$$P_{ij} = \begin{cases} \mathbf{L}_{f,ij} / \sum_{k\in N^{C_l}(i)} \mathbf{L}_{f,ik} & i \in F,\ j \in N^{C_l}(i) \\ 1 & j = i,\ i \in C \\ 0 & \text{elsewhere} \end{cases}, \tag{3.7}$$

where $N^{C_l}(i)$ is the set of (at most l) C-nodes that node $i \in F$ is associated with.

5. Once P has been computed, the coarse graph Laplacian $\mathbf{L}_c$ is defined as

$$\mathbf{L}_c = P^t \mathbf{L}_f P.$$

More generally, given a hierarchy of levels, if $\mathbf{L}^l$ is the graph Laplacian at level l, then

$$\mathbf{L}^{l+1} := \left(P_{l+1}^l\right)^t \mathbf{L}^l P_{l+1}^l \tag{3.8}$$

where P_{l+1}^l is the interpolation operator from level $l+1$ to level l.

COMMENTS:

1. The affinity measure is a correlation-related value between two variables. In step 1 of

the algorithm above, it is a common practice to select the initial random guesses to have uniformly distributed values from [-1,1] and zero mean ([31]).

2. The set C introduced in step 3 is the set of coarse-grid nodes (C-nodes). These nodes should be selected such that each node in F is strongly coupled to a set of nodes in C.

3. Step 3(ii) guarantees that F-nodes that are not strongly connected to any C-node become C-nodes themselves. This step ensures that there are no "near-isolated" F-nodes, i.e., nodes having none or a few strong affinity and/or weighted graph connections to C after Step 3(i).

4. In step 4, the caliber of interpolation is defined as the maximum number of C-nodes that a node $i \in F$ can interpolate from ([31]).

3.2. Coarsening of Complex Power Flow Problems

The procedures for determining the coarse nodes and the interpolation operator for a general power flow problem must be extended to the complex case. These procedures are similar to the ones described in Section 3.1, but with some modifications. Recall that the power flow problem is given in polar form by

$$\begin{aligned} \mathbf{P}_i &= |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{B}_{ij} \sin\left(\delta_i - \delta_j\right) + \mathbf{G}_{ij} \cos\left(\delta_i - \delta_j\right) \right\} \qquad (3.9) \\ &\quad \text{for } i = 1 \ldots, n-1, \\ \mathbf{Q}_i &= |\mathbf{V}_i| \sum_{j=1}^{n} |\mathbf{V}_j| \left\{ \mathbf{G}_{ij} \sin\left(\delta_i - \delta_j\right) - \mathbf{B}_{ij} \cos\left(\delta_i - \delta_j\right) \right\} \qquad (3.10) \\ &\quad \text{for every PQ bus } i, \\ \left(|\mathbf{V}_i^{sp}|\right)^2 &= |\mathbf{V}_i|^2 \qquad (3.11) \\ &\quad \text{for every PV bus } i, \end{aligned}$$

and in cartesian form by

$$\begin{aligned}
\mathbf{P}_i &= \mathbf{V}_i^R \sum_{j=1}^{n} \left(\mathbf{G}_{ij}\mathbf{V}_j^R - \mathbf{B}_{ij}\mathbf{V}_j^I\right) + \mathbf{V}_i^I \sum_{j=1}^{n} \left(\mathbf{B}_{ij}\mathbf{V}_j^R + \mathbf{G}_{ij}\mathbf{V}_j^I\right) \qquad (3.12)\\
&\quad \text{for } i = 1, \ldots, n-1\\
\mathbf{Q}_i &= -\mathbf{V}_i^R \sum_{j=1}^{n} \left(\mathbf{B}_{ij}\mathbf{V}_j^R + \mathbf{G}_{ij}\mathbf{V}_j^I\right) + \mathbf{V}_i^I \sum_{j=1}^{n} \left(\mathbf{G}_{ij}\mathbf{V}_j^R - \mathbf{B}_{ij}\mathbf{V}_j^I\right) \qquad (3.13)\\
&\quad \text{for every PQ bus } i\\
\left(|\mathbf{V}^{sp}|\right)^2 &= \left(\mathbf{V}_i^R\right)^2 + \left(\mathbf{V}_i^I\right)^2 \qquad (3.14)\\
&\quad \text{for every PV bus } i
\end{aligned}$$

where $\mathbf{V}_i^R$ and $\mathbf{V}_i^I$ are the real and imaginary components of the voltage $\mathbf{V}_i$. This formulation may improve the accuracy of the calculations because polynomial expressions of the real and imaginary components of voltages, admittances and power injections are involved rather than transcendental functions.

3.2.1. Smoothing

Since PV and PQ buses are of different nature, and the injected power $\mathbf{S}_i$ is fully known on PQ buses, equations (3.10) and (3.13) are consistent only if i is a PQ bus. Therefore, the smoother can be applied exclusively to equations corresponding to PQ buses at every level, except for the coarsest level where a different method will be used. Applying the smoother only to the equations corresponding to PQ buses reduces the effectiveness of the smoother since only voltages at the PQ buses are resolved, while voltages at PV buses are left untouched until the coarsest level is reached. However, not resolving the PV bus voltages in the intermediate levels does not significantly affect the effectiveness of the smoother because the number of PV buses is considerably less than the number of PQ buses.

3.2.2. Coarsening Procedure for Complex Admittance Matrices

Because of the physical differences between the PV and PQ buses (PV buses have a constraint condition), PQ nodes are permitted to interpolate only from PQ buses, so the PV buses and the slack bus need to be removed from the interpolatory set of every PQ bus. On the other hand, because the undetermined quantities for the PV buses are left unresolved at every level until the coarsest level, it is necessary to include all the PV buses into every coarse grid during the coarsening procedure. Other modifications to the coarsening procedure and the construction of the interpolation operator are:

1. By assuming that the admittance matrix is nearly a graph Laplacian and all the voltage magnitudes (in p.u.) are nearly constant, the graph Laplacian in (3.3) is replaced with the complex admittance matrix $\mathbf{Y}$. Thus,

$$\mathbf{L}_f = \mathbf{Y}. \tag{3.15}$$

2. The inner product $(\mathbf{x}_i, \mathbf{x}_j)$ in (3.4) is extended to complex values, giving

$$(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{K} \overline{\mathbf{x}_i^k} \mathbf{x}_j^k. \tag{3.16}$$

3. The inequalities in (3.6) are now expressed in terms of the moduli of the coefficients of the graph Laplacian defined in (3.15), giving

$$\frac{\sum_{j \in C \cap \tilde{N}(i)} c_{ij}}{\sum_{j \in N(i)} c_{ij}} \leq Q \text{ or } \frac{\sum_{j \in C \cap \tilde{N}(i)} |\mathbf{L}_{f,ij}|}{\sum_{j \in N(i)} |\mathbf{L}_{f,ij}|} \leq Q. \tag{3.17}$$

   Here, $\tilde{N}(i) = N(i) = i$ if i is a PV bus. If i is a PQ bus, then $\tilde{N}(i)$ denotes the set of neighboring nodes to i that remains after removing all the PV buses (and the slack bus) from $N(i)$.

4. With interpolation operator P, the coarse graph Laplacian $\mathbf{L}_c$ in a two-level setting is

$$\mathbf{L}_c = R\mathbf{L}_f P \tag{3.18}$$

where the restriction operator R is given by $R = P^t$ if $\mathbf{L}_f$ is symmetric, and $R = P^* = \overline{P}^t$ otherwise ([35]). The same applies for a multilevel setting; i.e., given $\mathbf{L}^l$ defined on a grid a level l, the level $l+1$ graph Laplacian is computed as

$$\mathbf{L}^{l+1} = \left(R_{l+1}^l\right)^t \mathbf{L}^l P_{l+1}^l \tag{3.19}$$

where $R_{l+1}^l = \left(P_{l+1}^l\right)^t$ if $\mathbf{L}^l$ is symmetric and $R_{l+1}^l = \overline{\left(P_{l+1}^l\right)}^t$ otherwise.

Algorithm 3.1 summarizes the steps of the coarsening technique for the power flow problem.

Algorithm 3.1 Coarsening procedure for the power flow equations

1: Construct a set of orthogonal test vector $\left\{\mathbf{x}^{(k)}\right\}_{k=1}^K$ by relaxing on the problem $\mathbf{Lx} = \mathbf{0}$ where $\mathbf{L}$ is given by equation (3.15).
2: Determine the affinity measure between two nodes i and j by (3.4), and using the inner product (5.41).
3: Perform the first pass in the selection of coarse nodes by following step 3(i) of the algorithm described in Section 3.1.
4: Perform the second pass in the selection of coarse nodes by following step 3(ii) in the algorithm described in Section 3.1, but with the inequalities given in (3.17).
5: Add all PV buses and the slack bus to C.
6: Construct the interpolation operator P according to (3.7).

3.2.3. Coarse-Grid Power Flow Problem for the AMG-FAS Scheme

With the coarsening procedure and the smoother already defined, there is only one aspect that remains to be attended to describe the AMG-FAS scheme for the power flow problem. Specifically, we need to construct the coarse power flow equations.

Assume that an L-level hierarchy of grids has been already formed. According to the FAS algorithm, the power flow equations at level 1 are given (including the constraint on

the PV buses) as

$$\mathfrak{N}^1\left(\mathbf{V}^1\right) = \mathbf{f}^1$$

where

$$\begin{cases} \mathbf{f}_i^1 = \mathbf{S}_i & \text{if } i \text{ is a PQ bus} \\ Re\left(\mathbf{f}_i^1\right) = Re\left(\mathbf{S}_i\right) & \text{if } i \text{ is a PV bus} \\ \left(|\mathbf{V}_i^{sp}|\right)^2 = \left(|\mathbf{V}_i|\right)^2 & \text{if } i \text{ is a PV bus} \end{cases} \tag{3.20}$$

and

$$\begin{cases} \left[\mathfrak{N}^1\left(\mathbf{V}^1\right)\right]_i = \mathbf{V}_i^1\left[\overline{\mathbf{YV}}\right]_i & \text{if } i \text{ is a PQ bus} \\ Re\left(\left[\mathfrak{N}^1\left(\mathbf{V}^1\right)\right]_i\right) = Re\left(\mathbf{V}_i^1\left[\overline{\mathbf{YV}}\right]_i\right) & \text{if } i \text{ is a PV bus} \end{cases} . \tag{3.21}$$

Assuming that the nonlinear problem at level $l-1$ $(2 \leq l \leq L)$ is defined by the equation

$$\mathfrak{N}^{l-1}\left(\mathbf{V}^{l-1}\right) = \mathbf{f}^{l-1}, \tag{3.22}$$

the problem at level l is given by

$$\mathfrak{N}^{l}\left(\mathbf{V}^{l}\right) = \mathbf{f}^{l} \tag{3.23}$$

where $\mathfrak{N}^l\left(\mathbf{V}^l\right)$ is constructed using the coarse graph Laplacian $\mathbf{L}^l$, $\mathbf{V}^l = \widehat{R}_{l-1}^l \mathbf{V}^{l-1} + \mathbf{e}^l$, where $\widehat{R}_{l-1}^l$ is the injection from level $l-1$ to level l, $\mathbf{e}^l$ is a coarse approximation to the error $\mathbf{e}^{l-1}$, and $\mathbf{f}^l = \mathfrak{N}^l\left(\widehat{R}_{l-1}^l \mathbf{V}^{l-1}\right) + R_{l-1}^l\left[\mathbf{f}^{l-1} - \mathfrak{N}^{l-1}\left(\mathbf{V}^{l-1}\right)\right]$. Thus, for the polar form the equations at level L are

$$\begin{aligned} \tilde{\mathbf{P}}_i^L &= |\tilde{\mathbf{V}}_i^L| \sum_{j=1}^{n_L} |\tilde{\mathbf{V}}_j^L| \left\{\tilde{\mathbf{B}}_{ij}^L \sin\left(\tilde{\delta}_i^L - \tilde{\delta}_j^L\right) + \tilde{\mathbf{G}}_{ij}^L \cos\left(\tilde{\delta}_i^L - \tilde{\delta}_j^L\right)\right\} \qquad (3.24) \\ &\quad \text{for } i = 1, \ldots, n-1, \\ \tilde{\mathbf{Q}}_i^L &= |\tilde{\mathbf{V}}_i^L| \sum_{j=1}^{n_L} |\tilde{\mathbf{V}}_j^L| \left\{\tilde{\mathbf{G}}_{ij}^L \sin\left(\tilde{\delta}_i^L - \tilde{\delta}_j^L\right) - \tilde{\mathbf{B}}_{ij}^L \cos\left(\tilde{\delta}_i^L - \tilde{\delta}_j^L\right)\right\} \qquad (3.25) \\ &\quad \text{for every PQ bus } i, \\ \left(|\mathbf{V}_i^{sp}|\right)^2 &= |\tilde{\mathbf{V}}_i^L|^2 \quad \text{for every P bus } i \qquad (3.26) \end{aligned}$$

where $\tilde{\mathbf{P}}_i^L = Re\left(\mathbf{f}_i^L\right)$. $\tilde{\mathbf{Q}}_i^L = Im\left(\mathbf{f}_i^L\right)$, $\tilde{\mathbf{G}}_{ij}^L$ and $\tilde{\mathbf{B}}_{ij}^L$ denote the real and imaginary components of the coarse admittance matrix at level L, n_L is the number of nodes in the grid at level L, and $|\tilde{\mathbf{V}}_j^L|$ and $\tilde{\delta}_j^L$ are to the magnitude and phase angle of the unknown $\tilde{\mathbf{V}}_i^L$.

The equations at level L in cartesian coordinates are

$$\begin{aligned}
\tilde{\mathbf{P}}_i^L &= \left(\tilde{\mathbf{V}}_i^R\right)^L \sum_{j=1}^{n_L} \left\{ \tilde{\mathbf{G}}_{ij}^L \left(\tilde{\mathbf{V}}_j^R\right)^L - \tilde{\mathbf{B}}_{ij}^L \left(\tilde{\mathbf{V}}_j^I\right)^L \right\} \\
&\quad + \left(\tilde{\mathbf{V}}_i^I\right)^L \sum_{j=1}^{n_L} \left\{ \tilde{\mathbf{B}}_{ij}^L \left(\tilde{\mathbf{V}}_j^R\right)^L + \tilde{\mathbf{G}}_{ij}^L \left(\tilde{\mathbf{V}}_j^I\right)^L \right\} \qquad (3.27) \\
&\quad \text{for } i = 1, \ldots, n-1, \\
\tilde{\mathbf{Q}}_i^L &= -\left(\tilde{\mathbf{V}}_i^R\right)^L \sum_{j=1}^{n_L} \left\{ \tilde{\mathbf{B}}_{ij}^L \left(\tilde{\mathbf{V}}_j^R\right)^L - \tilde{\mathbf{G}}_{ij}^L \left(\tilde{\mathbf{V}}_j^I\right)^L \right\} \\
&\quad + \left(\tilde{\mathbf{V}}_i^I\right)^L \sum_{j=1}^{n_L} \left\{ \tilde{\mathbf{G}}_{ij}^L \left(\tilde{\mathbf{V}}_j^R\right)^L - \tilde{\mathbf{B}}_{ij}^L \left(\tilde{\mathbf{V}}_j^I\right)^L \right\} \qquad (3.28) \\
&\quad \text{for every PQ bus } i, \\
\left(|\mathbf{V}_i^{sp}|\right)^2 &= \left[\left(\tilde{\mathbf{V}}_i^R\right)^L\right]^2 + \left[\left(\tilde{\mathbf{V}}_i^I\right)^L\right]^2 \quad \text{for every P bus } i. \qquad (3.29)
\end{aligned}$$

The solver at the coarsest level is Newton-Raphson. The resulting AMG-FAS scheme for the solution of the power flow equations $\mathfrak{N}^1\left(\mathbf{V}^1\right) = \mathbf{f}^1$ is described in Algorithm 3.2.

Algorithm 3.2 AMG-FAS V(ν_1, ν_2)-cycle

AMG-FAS($\mathfrak{N}^l, \mathbf{f}^l, \mathbf{V}^l, l, \nu_1, \nu_2$)

if $l = L$ **then**

  Solve $\mathfrak{N}^l\left(\mathbf{V}^l\right) = \mathbf{f}^l$ using Newton's method.

else

  Relax $\mathfrak{N}^l\left(\mathbf{V}^l\right) = \mathbf{f}^l$ ν_1 times using the iterate (1.14) for each PQ bus i.

  $\mathbf{d}^l \leftarrow \mathbf{f}^l - \mathfrak{N}^l\left(\mathbf{V}^l\right)$.

  $\mathbf{d}^{l+1} \leftarrow R_l^{l+1}\mathbf{d}^l$.

  $\hat{\mathbf{V}}^{l+1} \leftarrow \hat{R}_l^{l+1}\mathbf{V}^l$

  $\mathbf{f}^{l+1} \leftarrow \mathbf{d}^{l+1} + \mathfrak{N}^{l+1}\left(\hat{\mathbf{V}}^{l+1}\right)$.

  $\mathbf{W}^{l+1} \leftarrow$ AMG-FAS($\mathfrak{N}^{l+1}, \mathbf{f}^{l+1}, \hat{\mathbf{V}}^{l+1}, l+1, \nu_1, \nu_2$).

  $\mathbf{e}^{l+1} \leftarrow \mathbf{W}^{l+1} - \hat{\mathbf{V}}^{l+1}$

  $\mathbf{V}^l \leftarrow \mathbf{V}^l + P_{l+1}^l\mathbf{e}^{l+1}$

  Relax $\mathfrak{N}^l\left(\mathbf{V}^l\right) = \mathbf{f}^l$ ν_2 times using the iterate (1.14) for each PQ bus i.

end if

Chapter 4

Multigrid with Multiplicative Correction for the Power Flow Equation

A different strategy can be developed for solving the power flow equations by constructing a hierarchy of power flow problems, all having the form

$$\mathbf{S} = D_{\mathbf{V}} \left(\overline{\mathbf{Y}\mathbf{V}} \right), \tag{4.1}$$

where $D_{\mathbf{V}}$ is a diagonal matrix with the nodal voltages along the diagonal. This idea was proposed by Ponce *et al.* in ([39]). In order to describe this new approach, consider a two-grid scenario. The coarse-grid problem has to have the form of (4.1). This can be attained by using a multiplicative update of the form

$$\mathbf{V} \leftarrow D_{\mathbf{V}} \left(P\mathbf{W} \right) \tag{4.2}$$

for the fine grid element $\mathbf{V}$, rather than using the traditional additive update

$$\mathbf{V} \leftarrow \mathbf{V} + \left(P\hat{\mathbf{W}} \right), \tag{4.3}$$

in the correction phase. Here, P is an interpolation operator, $\mathbf{W}$ and $\hat{\mathbf{W}}$ are multiplicative and additive coarse-grid corrections to $\mathbf{V}$, respectively. In addition, in the method decribed in ([39]), the admittance matrix is assumed symmetric. Hence, only the nonzero elements of the upper triangular part are considered in the coarsening procedure. These elements are first ordered by their magnitudes (i.e., ordering the quantities $|\mathbf{Y}_{ij}| \neq 0$ with $j > i$). Then, nodes i and j are aggregated together into one coarse node if $|\mathbf{Y}_{ij}|$ is large compared to the magnitude of other admittance matrix entries.

4.1. Description of the Multilevel Scheme

The following lemma shows how the multiplicative correction leads to a coarse-grid problem having the same form as the fine-grid problem.

Lemma 4.1 ([39]) *Let $P \in \{0,1\}^{n\times l}$, $l \leq n$, denote a matrix with exactly one 1 in each row and let $\mathbf{x} \in \mathbb{C}^l$. Then*

$$P^t D_{P\mathbf{x}} = D_{\mathbf{x}} P^t.$$

Using the above lemma and (4.2), the coarse-grid problem can be derived. Suppose that $\mathbf{U}$ is an exact solution to (4.1). Thus, $\mathbf{U}$ satisfies

$$\mathbf{S} = D_{\mathbf{U}} \left(\overline{\mathbf{Y}\mathbf{U}}\right). \tag{4.4}$$

Let $\mathbf{V}$ be an approximation to $\mathbf{U}$ and $\mathbf{W}$ a coarse-grid correction to $\mathbf{V}$ such that

$$\mathbf{U} = D_{\mathbf{V}} \left(P\mathbf{W}\right). \tag{4.5}$$

By replacing (4.5) into (4.4), we have that

$$D_{\mathbf{V}} D_{P\mathbf{W}} \overline{\mathbf{Y} D_{\mathbf{V}}} P\overline{\mathbf{W}} = \mathbf{S}. \tag{4.6}$$

With the restriction operator chosen as $R = P^t$, then projection of equation (4.6) onto the coarse grid gives us

$$\begin{aligned}
P^t \left(D_{\mathbf{V}} D_{P\mathbf{W}} \overline{\mathbf{Y} D_{\mathbf{V}}} P\overline{\mathbf{W}}\right) &= P^t\mathbf{S} \\
P^t D_{P\mathbf{W}} D_{\mathbf{V}} \left(\overline{\mathbf{Y} D_{\mathbf{V}}} P\overline{\mathbf{W}}\right) &= P^t\mathbf{S} \\
D_{\mathbf{W}} P^t D_{\mathbf{V}} \left(\overline{\mathbf{Y} D_{\mathbf{V}}} P\overline{\mathbf{W}}\right) &= P^t\mathbf{S} \\
D_{\mathbf{W}} \left(P^t D_{\mathbf{V}} \overline{\mathbf{Y} D_{\mathbf{V}}} P\right) \overline{\mathbf{W}} &= P^t\mathbf{S} \\
D_{\mathbf{W}} \overline{\mathbf{Y}}_c \overline{\mathbf{W}} &= \mathbf{S}_c
\end{aligned} \tag{4.7}$$

where $\overline{\mathbf{Y}}_c = P^t D_{\mathbf{V}} \overline{\mathbf{Y} D_{\mathbf{V}}} P$ and $\mathbf{S}_c = P^t \mathbf{S}$.

Assuming that a hierarchy of L grids has been already built, the problem at level 1 is

$$\mathbf{S}^1 = D_{\mathbf{V}^1} \left(\overline{\mathbf{Y}^1 \mathbf{V}^1} \right) \tag{4.8}$$

where $\mathbf{Y}^1 = \mathbf{Y}$, $\mathbf{S}^1 = \mathbf{S}$ and $\mathbf{V}^1 = \mathbf{V}$. Thus, given the problem

$$\mathbf{S}^k = D_{\mathbf{V}^k} \left(\overline{\mathbf{Y}^k \mathbf{V}^k} \right) \tag{4.9}$$

defined at level $k \geq 1$, the problem at level $k+1$ is given by

$$\mathbf{S}^{k+1} = D_{\mathbf{V}^{k+1}} \left(\overline{\mathbf{Y}^{k+1} \mathbf{V}^{k+1}} \right) \tag{4.10}$$

where $\mathbf{Y}^{k+1} = \left(P^k\right)^t \overline{D_{\mathbf{V}^k}} \mathbf{Y}^k D_{\mathbf{V}^k} P^k$, $\mathbf{S}_{k+1} = \left(P^k\right)^t \mathbf{S}^k$ and P^k is an interpolation operator from level $k+1$ to level k.

At every level, the error is smoothed by applying a few sweeps of Gauss-Seidel iteration. If $\hat{\mathbf{V}}^1$ is solution to (4.8), then $\hat{\mathbf{W}}^k = \mathbf{1} = [1, \ldots, 1]^t$ is solution to (4.9) for every $k > 1$. This fact suggests that $\mathbf{V}^k = \mathbf{1}$ is a good initial guess for the process at every level $k = 2, \ldots, L$ ([39]).

Once the problem at level L has been solved, the solution $\mathbf{V}^L$ is interpolated up to level $L-1$ via the operator P_L^{L-1} and the result corrects $\mathbf{V}^{L-1}$ in a multiplicative fashion:

$$\mathbf{V}^{L-1} \leftarrow D_{\mathbf{V}^{L-1}} \left(P_L^{L-1} \mathbf{V}^L \right). \tag{4.11}$$

Then $\mathbf{V}^{L-1}$ is smoothed before being used to correct the approximation $\mathbf{V}^{L-2}$ a level $L-2$. In general, given the solution $\mathbf{V}^k$ at level k, the solution at level $k-1$ is obtained by updating the approximation $\mathbf{V}^{k-1}$ using a multiplicative correction

$$\mathbf{V}^{k-1} \leftarrow D_{\mathbf{V}^{k-1}} \left(P_k^{k-1} \mathbf{V}^k \right) \tag{4.12}$$

and smoothing $\mathbf{V}^{k-1}$.

4.2. Modified Scheme

Here, a similar scheme is presented with some modifications. First, because of the different attributes between PV and PQ buses, the voltages at these buses will be resolved in different fashions. The strategy used here is the same as in Section 3.2: at every level, some Gauss-Seidel sweeps are applied to the equations corresponding to PQ buses to smooth out the error; and the equations associated to the PV buses are left untouched until the coarsest level is reached, where the PV components of the solution are resolved.

At level L, the problem is solved using Newton's iteration for all the equations. The equations expressed in cartesian coordinates have the same form as (3.27)-(3.28) with the constraint

$$(|\mathbf{V}_i^{sp}|)^2 = \left[\left(|\hat{\mathbf{V}}_i^1|\right)\right]^2 \left\{\left[\left(\mathbf{V}_i^{L,R}\right)\right]^2 + \left[\left(\mathbf{V}_i^{L,I}\right)\right]^2\right\} \quad \text{for every PV bus } i. \tag{4.13}$$

Here $\left(|\hat{\mathbf{V}}_i^1|\right)$ is the approximated voltage magnitude at the PV bus i on level 1. As in the finer levels, the initial guess at level L is $\mathbf{V}^L = \left(\mathbf{V}_i^{L,R}\right) + \hat{\imath}\left(\mathbf{V}_i^{L,I}\right) = \mathbf{1}$. Equation (4.13) results from considering (3.11) at level 1 for every PV bus. If $\hat{\mathbf{V}}^1$ is an approximation to the exact solution $\mathbf{V}^{sp}$ of (3.11) and $\mathbf{W}$ is such that $\mathbf{V}^{sp} = D_{\hat{\mathbf{V}}^1}\mathbf{W}$, then for every PV bus i

$$|\mathbf{V}_i^{sp}|^2 = \left|\hat{\mathbf{V}}_i^1\mathbf{W}_i\right|^2 = \left|\hat{\mathbf{V}}_i^1\right|^2 |\mathbf{W}_i|^2 = \left|\hat{\mathbf{V}}_i^1\right|^2 \left\{\left(\mathbf{W}_i^R\right)^2 + \left(\mathbf{W}_i^I\right)^2\right\} \tag{4.14}$$

where $\mathbf{W}_i^R$ and $\mathbf{W}_i^I$ are the real and imaginary components of $\mathbf{W}_i$, respectively.

For the AMG-FAS scheme described in Chapter 3 the hierarchy of grids, and the coarse-grid and transfer operators are computed prior to the solution process. For Ponce's method requires these elements to be recomputed after each multigrid cycle, which makes this method computationally more expensive.

The procedures of a multigrid V-cycle scheme for solving the power flow equations using a multiplicative correction, including the modifications presented above, are shown in Algorithm 4.1.

Algorithm 4.1 MG-V(ν_1, ν_2)-Cycle-MULT

MG-MULT($\mathbf{Y}^k, \mathbf{V}^k, \mathbf{S}^k, k, \nu_1, \nu_2$)

if $k = L$ **then**

    Solve (3.27)-(3.28) and (4.13) using Newton's method.

else

    Relax (4.9) ν_1 times using the iterate (1.14) for each PQ bus i.

    $R_k^{k+1} \leftarrow\leftarrow \left(P_{k+1}^k\right)^t$.

    $\mathbf{Y}^{k+1} \leftarrow \left(P_{k+1}^k\right)^t \overline{D_{\mathbf{V}^k}} \mathbf{Y}^k D_{\mathbf{V}^k} P_{k+1}^k$.

    $\mathbf{V}^{k+1} \leftarrow \mathbf{1}$.

    $\mathbf{S}^{k+1} \leftarrow \left(P_{k+1}^k\right)^t \mathbf{S}^k$

    $\mathbf{V}^{k+1} \leftarrow$ MG-MULT $\left(\mathbf{Y}^{k+1}, \mathbf{V}^{k+1}, \mathbf{S}^{k+1}, k+1, \nu_1, \nu_2\right)$.

    $\mathbf{V}^k \leftarrow D_{\mathbf{V}^k}\left(P_{k+1}^k \mathbf{V}^{k+1}\right)$

    Relax (4.9) ν_2 times using the iterate (1.14) for each PQ bus i.

end if

Chapter 5

Convergence Analysis of Multigrid with Affinity-Based Coarsening

The aim of this chapter is to discuss the convergence of the two-level AMG-FAS scheme for the power flow equations using the affinity-based coarsening procedure introduced in Chapter 3. Convergence is analyzed for a two-level rather than a multilevel setting because, in general, the theory for multilevel methods involves more restricted conditions ([34, 37]). Although a convergence factor can be approximated only on the smoother, we are interested in convergence estimates based on the norm of the complete two-grid scheme operator, which involves both the smoother and the coarse-grid correction operator. We start by discussing the convergence of a two-grid AMG scheme for a linear problem with a real symmetric positive definite matrix using the theory presented in ([41]) and ([34]). Then, we consider problems involving complex-valued matrices, establishing conditions for the extension of the theory from the real case. Finally, we review the theory of convergence of a two-level FAS scheme for nonlinear PDE-based problems given in ([24]), which provides a starting point for discussing on the convergence of the AMG-FAS algorithm for the power flow equations.

5.1. Linear Problem with a Real Symmetric Positive Definite Operator

Consider the linear system

$$\mathbf{A}\mathbf{u} = \mathbf{f} \tag{5.1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a symmetric positive definite (SPD) linear operator. The set of unknowns $\mathbf{u}$ is associated to a set of indices (nodes), $\Omega = \{1, \ldots, n\}$. We seek conditions that guarantee the convergence of AMG for problem (5.1) and then progressively move to the power flow

problem establishing suitable conditions for the convergence of the proposed AMG-FAS scheme.

Consider a two-level AMG setting. Let C denote the set of coarse-grid nodes and P the prolongation operator. The restriction R is defined by $R = P^t$, and the coarse-grid operator $\mathbf{A}_c$ is defined by $\mathbf{A}_c = R\mathbf{A}P$. In addition, the pre- and post-smoothing operators $\hat{S}$ and S are expressed as

$$\hat{S} = \mathbf{I} - \hat{\mathbf{Q}}^{-1}\mathbf{A} \quad \text{and} \quad S = \mathbf{I} - \mathbf{Q}^{-1}\mathbf{A} \tag{5.2}$$

for some matrices $\hat{\mathbf{Q}}$ and $\mathbf{Q}$, respectively. Here, $\mathbf{I} \in \mathbb{R}^{n\times n}$ denotes the identity operator. The matrices $\hat{\mathbf{Q}}$ and $\mathbf{Q}$ are determined by the smoother. For instance, if $\mathbf{A} = \mathbf{D} - \mathbf{U} - \mathbf{U}^t$, where $\mathbf{D} = diag\,(\mathbf{A})$ and $\mathbf{U}$ is the negative strictly upper triangular part of $\mathbf{A}$, then $\mathbf{Q} = \mathbf{D}$ for the Jacobi iteration, and $\mathbf{Q} = \mathbf{D} - \mathbf{U}^t$ for the Gauss-Seidel iteration. The approximate solution is updated through the operators $\hat{\mathfrak{S}}$ and $\mathfrak{S}$ associated to $\hat{S}$ and S, respectively:

$$\mathbf{v} \leftarrow \hat{\mathfrak{S}}\,(\mathbf{v}, \mathbf{f}) \quad \text{and} \quad \mathbf{v} \leftarrow \mathfrak{S}\,(\mathbf{v}, \mathbf{f})\,. \tag{5.3}$$

Using C to denote the elements in the coarse level, the two-level AMG V(1,1)-cycle is described in Algorithm 5.1.

Algorithm 5.1 AMG V(1, 1)-Cycle

AMG($\mathbf{A}, \mathbf{f}, \mathbf{v}$)

1: $\mathbf{v} \leftarrow \hat{\mathfrak{S}}\,(\mathbf{v}, \mathbf{f})$

2: $\mathbf{r} \leftarrow (\mathbf{f} - \mathbf{A}\mathbf{v})$

3: $\mathbf{f}_c \leftarrow R\mathbf{r}$

4: $\mathbf{v}_c \leftarrow \mathbf{A}_c^{-1}\mathbf{f}_c$

5: $\mathbf{v} \leftarrow \mathbf{v} + P\mathbf{v}_c$

6: $\mathbf{v} \leftarrow \mathfrak{S}\,(\mathbf{v}, \mathbf{f})$

The $F-C$ partition and the prolongation operator have to be determined in such a way that the interaction between the smoother and the coarse-grid correction leads to an efficient AMG method.

We start the discussion of the convergence for the two-level setting by noticing that, since $\mathbf{A}$ is SPD, it leads to the inner product

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{A}} := \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle \tag{5.4}$$

with the corresponding norm

$$\|\mathbf{v}\|_{\mathbf{A}} := \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle_{\mathbf{A}}}. \tag{5.5}$$

$\mathbf{A}$ being SPD also implies that $\mathbf{A}_{ii} > 0 \ \forall i = 1, \ldots, n$. Thus, denoting the diagonal of $\mathbf{A}$ by $\mathbf{D}$, two additional inner products are

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{D}} := \langle \mathbf{D}\mathbf{u}, \mathbf{v} \rangle \quad \text{and} \quad \langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}} := \left\langle \mathbf{D}^{-1}\mathbf{A}\mathbf{u}, \mathbf{A}\mathbf{v} \right\rangle \tag{5.6}$$

together with their corresponding norms

$$\|\mathbf{v}\|_{\mathbf{D}} := \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle_{\mathbf{D}}} \quad \text{and} \quad \|\mathbf{v}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}} := \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}}. \tag{5.7}$$

Recall from Section 2.1.3 that the two-grid error propagator is defined as

$$T = \mathbf{I} - P \left(\mathbf{A}_C\right)^{-1} R\mathbf{A}. \tag{5.8}$$

After replacing the explicit form of $\mathbf{A}_C$ into (5.8) and using the fact that $R = P^t$, the two-grid error propagation operator is given by

$$T = \mathbf{I} - P \left(P^t\mathbf{A}P\right)^{-1} P^t\mathbf{A}. \tag{5.9}$$

It was proven in Section 2.1.3 that the operator $\mathbf{I}-T = P \left(P^t\mathbf{A}P\right)^{-1} P^t\mathbf{A}$ is the $\mathbf{A}$-orthogonal projector onto the range $\mathcal{R}\left(P\right)$ of P, and $\mathcal{R}\left(P\right)$ is $\mathbf{A}$-orthogonal to the range $\mathcal{R}\left(T\right)$ of T.

In addition to the above conditions, the smoothing operators S and $\hat{S}$ are assumed to be convergent in the $\mathbf{A}$-norm, i.e., $\|S\|_{\mathbf{A}} < 1$ and $\|\hat{S}\|_{\mathbf{A}} < 1$. Lemma 5.1 below will show how the norms of the two-level V(0,1)-, V(1,0)- and V(1,1)-cycle operators are related to each other. To understand the lemma, we need the concept of the M-adjoint of a linear operator.

Definition 5.1 *Given a linear operator $\mathbf{A} \in \mathbb{R}^{m\times m}$, its adjoint with respect to the Euclidean inner product is a linear operator $\mathbf{B} \in \mathbb{R}^{m\times m}$ such that $\langle \mathbf{Av}, \mathbf{w}\rangle = \langle \mathbf{v}, \mathbf{Bw}\rangle$ for all $\mathbf{v}, \mathbf{w} \in \mathbb{R}^m$. Given an SPD matrix $\mathbf{M}$, $\mathbf{B}$ is said to be the $\mathbf{M}$-adjoint of $\mathbf{A}$ if $\langle \mathbf{Av}, \mathbf{w}\rangle_{\mathbf{M}} = \langle \mathbf{v}, \mathbf{Bw}\rangle_{\mathbf{M}}$ for all $\mathbf{v}, \mathbf{w} \in \mathbb{R}^m$.*

Lemma 5.1 ([36]) *Assume that $\mathbf{A}$ is SPD, the prolongation operator P is full rank and the two-grid error propagation operator is $T = \mathbf{I} - P\left(P^t\mathbf{A}P\right)^{-1}P^t\mathbf{A}$. In addition, assume that S and $\hat{S}$ are convergent under the $\mathbf{A}$-norm. Then the $\mathbf{A}$-adjoint of the two-level V(0,1)-cycle operator ST is the operator of the two-level V(1,0)-cycle $TS^\dagger$, with $S^\dagger = \mathbf{I} - \mathbf{Q}^{-t}\mathbf{A}$. Hence, the operators ST and $TS^\dagger$ have the same $\mathbf{A}$-norm; i.e., $\|ST\|_{\mathbf{A}} = \|TS^\dagger\|_{\mathbf{A}}$. Furthermore, the $\mathbf{A}$-norm of the two-level V(1,1)-cycle operator $STS^\dagger$ is $\|STS^\dagger\|_{\mathbf{A}} = \|ST\|_{\mathbf{A}}^2$.*

Therefore, according to Lemma 5.1, it suffices to discuss the convergence of the two-level V(0,1)-cycle operator. In ([41]) and ([34]) sufficient conditions on S and T are provided to guarantee the convergence of the two-level AMG scheme. The results are summarized in the next theorem.

Theorem 5.2 ([41]) *Assume that $\mathbf{A}$ is SPD, P is a full rank prolongation operator, the the two-level coarse-grid correction operator has the form $T = \mathbf{I} - P\left(P^t\mathbf{A}P\right)^{-1}P^t\mathbf{A}$, and the smoothing operator S is convergent under the $\mathbf{A}$-norm. If there exists $\alpha_g > 0$ such that*

$$\|S\mathbf{e}\|_{\mathbf{A}}^2 \leq \|\mathbf{e}\|_{\mathbf{A}}^2 - \alpha_g g\left(\mathbf{e}\right) \quad \textit{for all } \mathbf{e} \quad \textit{(smoothing assumption)} \tag{5.10}$$

and there exists $\beta_g > 0$ such that

$$\|T\mathbf{e}\|_{\mathbf{A}}^2 \leq \beta_g g\left(T\mathbf{e}\right) \quad \textit{for all } \mathbf{e} \quad \textit{(weak approximation assumption)}, \tag{5.11}$$

then

$$\|ST\|_{\mathbf{A}} \leq \sqrt{1 - \frac{\alpha_g}{\beta_g}} \tag{5.12}$$

where $g(\mathbf{e})$ *is a given nonnegative function.*

A common choice for $g(\mathbf{e})$ is $g(\mathbf{e}) = \|\mathbf{e}\|^2_{\mathbf{AD}^{-1}\mathbf{A}}$ ([41]). The approximation assumption (5.11) can be rewritten in terms of the action of the prolongation operator P by considering the $\mathbf{A}$-orthogonality between the range of P and the range of T. Then

$$\|T\mathbf{e}\|_{\mathbf{A}} = \inf_{\mathbf{e}_C} \|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{A}}. \tag{5.13}$$

Relationship (5.13) is illustrated in Figure 5.1. Hence, again using this $\mathbf{A}$-orthogonality and the Cauchy-Schwarz inequality, for an arbitrary $\mathbf{e}_C$ we have

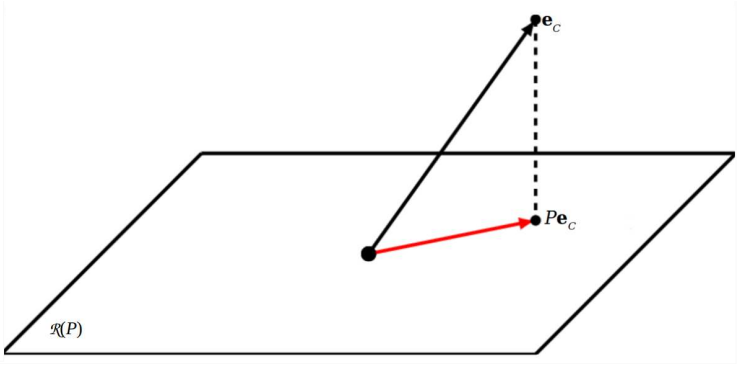


Figure 5.1: Graphical representation of relationship (5.13). The length of the dotted line corresponds to $\|T\mathbf{e}\|_{\mathbf{A}}$.

$$
\begin{aligned}
\|T\mathbf{e}\|_{\mathbf{A}}^2 &= \langle \mathbf{A}T\mathbf{e}, T\mathbf{e}\rangle \\
&= \langle \mathbf{A}T\mathbf{e}, T\mathbf{e} - P\mathbf{e}_C\rangle \\
&= \left\langle \mathbf{D}^{1/2}\mathbf{D}^{-1/2}\mathbf{A}T\mathbf{e}, T\mathbf{e} - P\mathbf{e}_C\right\rangle && (5.14)\\
&= \left\langle \mathbf{D}^{-1/2}\mathbf{A}T\mathbf{e}, \mathbf{D}^{1/2}\left(T\mathbf{e} - P\mathbf{e}_C\right)\right\rangle && (5.15)\\
&\leq \|\mathbf{D}^{-1/2}\mathbf{A}T\mathbf{e}\|\ \|\mathbf{D}^{1/2}\left(T\mathbf{e} - P\mathbf{e}_C\right)\| && (5.16)\\
&= \sqrt{\langle \mathbf{D}^{-1/2}\mathbf{A}T\mathbf{e}, \mathbf{D}^{-1/2}\mathbf{A}T\mathbf{e}\rangle}\ \sqrt{\langle \mathbf{D}^{1/2}\left(T\mathbf{e} - P\mathbf{e}_C\right), \mathbf{D}^{1/2}\left(T\mathbf{e} - P\mathbf{e}_C\right)\rangle} && (5.17)\\
&= \sqrt{\langle \mathbf{A}\mathbf{D}^{-1}\mathbf{A}T\mathbf{e}, T\mathbf{e}\rangle}\ \sqrt{\langle \mathbf{D}\left(T\mathbf{e} - P\mathbf{e}_C\right), T\mathbf{e} - P\mathbf{e}_C\rangle} && (5.18)\\
&= \|T\mathbf{e}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}\|T\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}. && (5.19)
\end{aligned}
$$

Since Ω is finite dimensional,

$$\inf_{\mathbf{e}_C}\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}} = \min_{\mathbf{e}_C}\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}} \quad \text{for all } \mathbf{e}.$$

Thus, for every $\mathbf{e}$ there exists $\hat{\mathbf{e}}_C$ such that

$$\|\mathbf{e} - P\hat{\mathbf{e}}_C\|_{\mathbf{D}} = \min_{\mathbf{e}_C}\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}.$$

In particular, since $\mathbf{e}_C$ in (5.19) is arbitrary, $\|T\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}$ can be replaced with $\|T\mathbf{e} - P\hat{\mathbf{e}}_C\|_{\mathbf{D}}$. Hence, if for some β_w we have that

$$\min_{\mathbf{e}_C}\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}^2 \leq \beta_w\|\mathbf{e}\|_{\mathbf{A}}^2 \tag{5.20}$$

for all $\mathbf{e}$, and in particular $T\mathbf{e}$, i.e., $\|T\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}^2 \leq \beta_w\|T\mathbf{e}\|_{\mathbf{A}}^2$, then, from (5.19), we have

$$
\begin{aligned}
\|T\mathbf{e}\|_{\mathbf{A}}^2 &\leq \|T\mathbf{e}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}\|T\mathbf{e} - P\hat{\mathbf{e}}_C\|_{\mathbf{D}} \\
&= \|T\mathbf{e}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}\min_{\mathbf{e}_C}\|T\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}} \\
&\leq \|T\mathbf{e}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}\sqrt{\beta_w}\|T\mathbf{e}\|_{\mathbf{A}}. && (5.21)
\end{aligned}
$$

Inequality (5.21) implies that

$$\|T\mathbf{e}\|_{\mathbf{A}} \leq \sqrt{\beta_w}\|T\mathbf{e}\|_{\mathbf{AD}^{-1}\mathbf{A}}$$

which leads to (5.11) with $g(T\mathbf{e}) = \|T\mathbf{e}\|_{\mathbf{AD}^{-1}\mathbf{A}}$. Thus, condition (5.20) will be used rather than (5.11).

5.1.1. Smoothing Assumption for Gauss-Seidel

In order to determine the conditions for the Gauss-Seidel iteration to satisfy the smoothing assumption (5.10), we start by stating Lemma 5.3 and Lemma 5.4. They are used later for the proof of the main result in Theorem 5.5.

Lemma 5.3, whose proof is given in ([44]), states an equivalent statement for the smoothing assumption given in (5.10).

Lemma 5.3 ([44]) *Let* $\mathbf{A}$ *be SPD and the smoothing operator be of the form* $S = \mathbf{I} - \mathbf{Q}^{-1}\mathbf{A}$ *for some nonsingular matrix* $\mathbf{Q}$*. Then the smoothing assumption (5.10) is equivalent to*

$$\alpha_g \left\langle \mathbf{Q}^t\mathbf{D}^{-1}\mathbf{Q}\mathbf{e}, \mathbf{e} \right\rangle \leq \left\langle \left(\mathbf{Q} + \mathbf{Q}^t - \mathbf{A}\right)\mathbf{e}, \mathbf{e} \right\rangle \quad \text{for all } \mathbf{e}. \tag{5.22}$$

The next lemma proves that the matrix norm $\|\mathbf{A}\|_{\mathbf{W}}$ defined for symmetric matrices as

$$\|\mathbf{A}\|_{\mathbf{W}} := \max_{1\leq i\leq n}\left\{ \frac{1}{\mathbf{W}_{ii}} \sum_{j=1}^{n} \mathbf{W}_{jj} \left|\mathbf{A}_{ij}\right| \right\} \tag{5.23}$$

where $\mathbf{W}$ is a $n \times n$ diagonal matrix with $\mathbf{W}_{ii} > 0$ for $i = 1, \ldots, n$ is induced by a vector norm.

Lemma 5.4 *The matrix norm defined by (5.23) is induced by the weighted vector 1-norm* $\|\mathbf{x}\|_{\mathbf{W}} := \|\mathbf{W}\mathbf{x}\|_1 = \sum_{i=1}^{n} |\mathbf{W}_{ii}\mathbf{x}_i|$.

Proof: By definition, the matrix norm induced by $\|\mathbf{x}\|_{\mathbf{W}}$ is given by

$$\|\mathbf{A}\|_{\mathbf{x},\mathbf{W}} = \sup_{\|\mathbf{x}\|_{\mathbf{W}}=1} \|\mathbf{A}\mathbf{x}\|_{\mathbf{W}}.$$

Then we need to prove that $\|\mathbf{A}\|_{\mathbf{x},\mathbf{W}}$ agrees with $\|\mathbf{A}\|_{\mathbf{W}}$. Consider $\mathbf{y} = \mathbf{W}\mathbf{x}$. Since $\mathbf{W}$ is SPD, $\mathbf{y}$ is uniquely determined by $\mathbf{x}$. We have

$$\begin{aligned}
\|\mathbf{A}\|_{\mathbf{x},\mathbf{W}} &= \sup_{\|\mathbf{x}\|_{\mathbf{W}}=1} \|\mathbf{A}\mathbf{x}\|_{\mathbf{W}} \\
&= \sup_{\|\mathbf{W}\mathbf{x}\|_1=1} \|\mathbf{W}\mathbf{A}\mathbf{x}\|_1 \\
&= \sup_{\|\mathbf{y}\|_1} \|\mathbf{W}\mathbf{A}\mathbf{W}^{-1}\mathbf{y}\|_1 \\
&= \|\mathbf{W}\mathbf{A}\mathbf{W}^{-1}\|_1.
\end{aligned}$$

Now notice that

$$\left(\mathbf{W}\mathbf{A}\mathbf{W}^{-1}\right)_{ij} = \frac{\mathbf{W}_{ii}}{\mathbf{W}_{jj}}\mathbf{A}_{ij}.$$

Thus, by definition, the induced 1-norm of $\mathbf{W}\mathbf{A}\mathbf{W}^{-1}$ is

$$\begin{aligned}
\|\mathbf{W}\mathbf{A}\mathbf{W}^{-1}\|_1 &= \max_{1\le j\le n}\left\{\sum_{i=1}^{n}\left|\frac{\mathbf{W}_{ii}}{\mathbf{W}_{jj}}\mathbf{A}_{ij}\right|\right\} \\
&= \max_{1\le i\le n}\left\{\frac{1}{\mathbf{W}_{ii}}\sum_{j=1}^{n}\mathbf{W}_{jj}\left|\mathbf{A}_{ij}\right|\right\} \quad \text{(by symmetry of } \mathbf{A}\text{)}.
\end{aligned}$$

Therefore, the induced matrix norm $\|\mathbf{A}\|_{\mathbf{x},\mathbf{W}}$ agrees with $\|\mathbf{A}\|_{\mathbf{W}}$. □

With the help of the above lemmas, the smoothing operator associated with Gauss-Seidel iteration is proven to satisfy the smoothing assumption (5.10). This result is stated in Theorem 5.5, whose proof is given in ([44]).

Theorem 5.5 ([44]) *Let* $\mathbf{A}$ *be SPD and define, for any vector* $\mathbf{w}$ *such that* $\mathbf{w}_i > 0$,

$$\gamma_- = max_{1\le i\le n}\left\{\frac{1}{\mathbf{w}_i\mathbf{A}_{ii}}\sum_{j<i}\mathbf{w}_j\left|\mathbf{A}_{ij}\right|\right\} \quad \textit{and} \quad \gamma_+ = max_{1\le i\le n}\left\{\frac{1}{\mathbf{w}_i\mathbf{A}_{ii}}\sum_{j>i}\mathbf{w}_j\left|\mathbf{A}_{ij}\right|\right\}. \tag{5.24}$$

Then Gauss-Seidel relaxation satisfies the smoothing assumption (5.10) with

$$\alpha_g = \frac{1}{(1+\gamma_-)(1+\gamma_+)}. \tag{5.25}$$

If $\mathbf{w}$ in (5.24) is chosen to be $\mathbf{w} = \mathbf{1}$, then the norm defined by (5.23) becomes simply the usual matrix 1-norm. Furthermore, γ_- and γ_+ are given by

$$\gamma_- = \max_{1\le i\le n}\left\{\frac{1}{\mathbf{A}_{ii}}\sum_{j<i}|\mathbf{A}_{ij}|\right\} \quad \text{and} \quad \gamma_+ = \max_{1\le i\le n}\left\{\frac{1}{\mathbf{A}_{ii}}\sum_{j>i}|\mathbf{A}_{ij}|\right\}.$$

If, in addition, $\mathbf{A}$ is assumed to be diagonally dominant, then

$$\begin{aligned}
\frac{1}{\mathbf{A}_{ii}}\sum_{j>i}|\mathbf{A}_{ij}| &\le \frac{1}{\mathbf{A}_{ii}}\sum_{\substack{j=1\\ j\neq i}}^{n}|\mathbf{A}_{ij}| \\
&\le \frac{1}{\mathbf{A}_{ii}}|\mathbf{A}_{ii}| \\
&= \frac{\mathbf{A}_{ii}}{\mathbf{A}_{ii}} \\
&= 1
\end{aligned}$$

for all $i = 1, \ldots, n$, and similarly

$$\frac{1}{\mathbf{A}_{ii}}\sum_{j<i}|\mathbf{A}_{ij}| \le 1$$

for all $i = 1, \ldots, n$. This implies that $\gamma_- \le 1$ and $\gamma_+ \le 1$. Thus

$$\frac{1}{4} \le \frac{1}{(1+\gamma_+)(1+\gamma_-)}.$$

Therefore, if $\mathbf{A}$ is SPD and diagonally dominant, the hypothesis of Theorem 5.5 is satisfied

with

$$\alpha_g \leq \frac{1}{4}. \tag{5.26}$$

5.1.2. Approximation Assumption for the Coarse-Grid Correction

Conditions are now determined on the coarse-node selection and the prolongation operator in order to find an appropriate value for the constant β_g in the approximation assumption (5.20):

$$\inf_{\mathbf{e}_C} \|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}{}^2 \leq \beta_g \|\mathbf{e}\|_{\mathbf{A}}^2 \quad \text{for all } \mathbf{e}.$$

First, assume that a subset C has been already selected from Ω as the set of the coarse nodes, and let $F = \Omega \setminus C$. In addition, as in Section 3.1, given the caliber of interpolation l, denote by $N^{C_l}(i)$ the maximal set of C-nodes to which node $i \in F$ can be interpolated from. In order to find a value of β_g that satisfies inequality (5.20), we can compare both sides of (5.20) by expressing it as summations involving entries of $\mathbf{e}$. The following lemma provides summation expansion for $\|\mathbf{e}\|_{\mathbf{A}}^2$.

Lemma 5.6 ([44]) *For any SPD M-matrix* $\mathbf{A}$ *such that* $\mathbf{A}$ *is weakly diagonally dominant the following identity holds:*

$$\|\mathbf{e}\|_{\mathbf{A}}^2 = \frac{1}{2}\sum_{i=1}^{n}\left[\sum_{j=1}^{n}\left(-\mathbf{A}_{ij}\right)\left(\mathbf{e}_i - \mathbf{e}_j\right)^2\right] + \sum_{i=1}^{n}\left(\sum_{j=1}^{n}\mathbf{A}_{ij}\right)\mathbf{e}^2 \quad \textit{for all } \mathbf{e}. \tag{5.27}$$

Since $\mathbf{A}$ is a weakly diagonally dominant M-matrix, $\sum_{j=1}^{n}\mathbf{A}_{ij} \geq 0$, and $\mathbf{A}_{ij} \leq 0$ if $j \neq i$ for all $i = 1, \ldots, n$. Thus, $\sum_{i=1}^{n}\left(\sum_{j=1}^{n}\mathbf{A}_{ij}\right)\mathbf{e}_i^2 \geq 0$. Then

$$
\begin{aligned}
\|\mathbf{e}\|_{\mathbf{A}}^2 &= \langle \mathbf{A}\mathbf{e}, \mathbf{e} \rangle \\
&= \sum_{i=1}^{n} \left[\sum_{j=1}^{n} \mathbf{e}_i \mathbf{A}_{ij} \mathbf{e}_j \right] \\
&= \frac{1}{2} \sum_{i=1}^{n} \left[\sum_{j=1}^{n} \left(-\mathbf{A}_{ij}\right) \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 \right] + \sum_{i=1}^{n} \left(\sum_{j=1}^{n} \mathbf{A}_{ij} \right) \mathbf{e}_i^2 \\
&= \frac{1}{2} \sum_{i=1}^{n} \left[\sum_{\substack{j=1 \\ j \neq i}}^{n} \left(-\mathbf{A}_{ij}\right) \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 \right] + \sum_{i=1}^{n} \left(\sum_{j=1}^{n} \mathbf{A}_{ij} \right) \mathbf{e}_i^2 \\
&\geq \sum_{i \in F} \left[\sum_{j \in N^{C_l}(i)} \left(-\mathbf{A}_{ij}\right) \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 \right] + \sum_{i \in F} \left(\sum_{j=1}^{n} \mathbf{A}_{ij} \right) \mathbf{e}_i^2 \\
&= \sum_{i \in F} \left[\sum_{j \in N^{C_l}(i)} \left(-\mathbf{A}_{ij}\right) \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 + \left(\sum_{j=1}^{n} \mathbf{A}_{ij} \right) \mathbf{e}_i^2 \right]. \qquad (5.28)
\end{aligned}
$$

Consider now a reordering of the indices in Ω such that the F-nodes appear first. Then, the vector $\mathbf{e}$ can be written blockwise as

$$
\mathbf{e} = \begin{bmatrix} \mathbf{e}_F \\ \mathbf{e}_C \end{bmatrix}
$$

where $\mathbf{e}_F$ and $\mathbf{e}_C$ are the blocks corresponding to the components of $\mathbf{e}$ in F and C, respectively. The prolongation operator can be also rewritten, according to the proposed reordering, as

$$
P = \begin{bmatrix} \mathbf{I}_{FC} \\ \mathbf{I} \end{bmatrix} \qquad (5.29)
$$

where $\mathbf{I}_{FC} \in \mathbb{R}^{n_F \times n_C}$ is a block of coarse-to-fine prolongation weights and $\mathbf{I} \in \mathbb{R}^{n_C \times n_C}$ is the identity on the coarse level. Hence,

$$\begin{aligned}
\mathbf{e} - P\mathbf{e}_C &= \begin{bmatrix} \mathbf{e}_F \\ \mathbf{e}_C \end{bmatrix} - \begin{bmatrix} \mathbf{I}_{FC} \\ \mathbf{I} \end{bmatrix} [\mathbf{e}_C] \\
&= \begin{bmatrix} \mathbf{e}_F - \mathbf{I}_{FC}\mathbf{e}_C \\ \mathbf{e}_C - \mathbf{e}_C \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{e}_F - \mathbf{I}_{FC}\mathbf{e}_C \\ \mathbf{0} \end{bmatrix}. \qquad (5.30)
\end{aligned}$$

With the help of (5.30), the expression $\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}^2$ can now be bounded by a summation in terms of the entries of $\mathbf{e}$ similar to that in (5.27). This result is given in the following lemma.

Lemma 5.7 ([44]) *Let* $\mathbf{A}$ *be a weakly diagonally dominant SPD matrix. Suppose that* P *is defined as in (5.29) with* $P_{ij} \geq 0$ *for* $1 \leq i \leq n$ *and* $1 \leq j \leq n_C$ *and* $\sum_{j \in N^{C_l}(i)} P_{ij} \leq 1$ *for* $1 \leq i \leq n$. *Then*

$$\|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}^2 \leq \sum_{i \in F} \mathbf{A}_{ii} \left[\sum_{j \in N^{C_l}(i)} P_{ij} \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 + \left(1 - \sum_{j \in N^{C_l}(i)} P_{ij}\right) \mathbf{e}_i^2 \right]. \qquad (5.31)$$

By considering (5.27) and (5.31), the weak approximation assumption (5.20) can be satisfied if

$$\begin{aligned}
&\sum_{i \in F} \sum_{j \in N^{C_l}(i)} \mathbf{A}_{ii} P_{ij} \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 + \sum_{i \in F} \mathbf{A}_{ii} \left(1 - \sum_{j \in N^{C_l}(i)} P_{ij}\right) \mathbf{e}_i^2 \;\leq \\
&\qquad \beta_g \left[\sum_{i \in F} \sum_{j \in N^{C_l}(i)} \left(-\mathbf{A}_{ij}\right) \left(\mathbf{e}_i - \mathbf{e}_j\right)^2 + \sum_{i \in F} \left(\sum_{j=1}^{n} \mathbf{A}_{ij} \right) \mathbf{e}_i^2 \right] \qquad (5.32)
\end{aligned}$$

for certain β_g. The existence of $\beta_g \geq 0$ such that inequality (5.32) is satisfied can be shown by first noticing that each term of the summations in both sides of (5.32) is nonnegative. In

fact, since $\mathbf{A}$ is a diagonally dominant M-matrix, we have that

$$\mathbf{A}_{ii} > 0,\ \mathbf{A}_{ij} \leq 0 \quad \text{for all } i,j = 1,\ldots,n\ (j \neq i) \quad \text{and} \quad \sum_{j=1}^{n} \mathbf{A}_{ij} \geq 0 \quad \text{for all } i = 1,\ldots,n.$$

In addition, because of the way P is constructed, we have that

$$P_{ij} \geq 0 \quad \text{for all } i = 1,\ldots,n,\ j \in N^{C_l}(i) \quad \text{and} \quad 1 - \sum_{j \in N^{C_l}(i)} P_{ij} = 0 \quad \text{for all } i = 1,\ldots,n.$$

Thus, the second summation on the left-hand side of (5.32) is always zero. This implies that

$$\mathbf{A}_{ii}\left(1 - \sum_{j \in N^{C_l}(i)} P_{ij}\right) \leq \beta_g \sum_{j=1}^{n} \mathbf{A}_{ij} \qquad \text{for all } i \in F \text{ and for any } \beta_g \geq 0. \tag{5.33}$$

Moreover, the value of the left-hand side of (5.32) is only determined by the first summation. It only remains to prove that there exists a $\beta_g \geq 0$ such that

$$\sum_{i \in F} \sum_{j \in N^{C_l}(i)} \mathbf{A}_{ii} P_{ij} \leq \beta_g \sum_{i \in F} \sum_{j \in N^{C_l}(i)} \left(-\mathbf{A}_{ij}\right). \tag{5.34}$$

If $\mathbf{A}_{ii}P_{ij} = 0$ for all $i = 1,\ldots,n$ and $j \in N^{C_l}(i)$, then (5.34) holds for any $\beta_g \geq 0$. Hence, (5.32) is satisfied. Otherwise, if $\mathbf{A}_{ii}P_{ij} > 0$ for some i and $j \in N^{C_l}(i)$, then $P_{ij} > 0$ ($\mathbf{A}_{ii} > 0$ because $\mathbf{A}$ is an M-matrix). However, $P_{ij} > 0$ implies that $(-\mathbf{A}_{ij}) > 0$. Thus, one can always find $\hat{\beta}_{ij} > 0$ such that

$$\mathbf{A}_{ii}P_{ij} \leq \hat{\beta}_{ij}\left(-\mathbf{A}_{ij}\right).$$

If we define

$$\beta_g := max_{i \in F}\left[\max_{j \in N_l^C(i)}\left\{\hat{\beta}_{ij}\right\}\right],$$

then (5.34) is satisfied and so (5.32) is satisfied. This proves the existence of $\beta_g \geq 0$ such that (5.32) is fulfilled.

From (5.34), and recalling how the entries of P are computed, we also have that

$$\beta_g\left(-\mathbf{A}_{ij}\right) \geq \mathbf{A}_{ii}P_{ij} = \mathbf{A}_{ii}\frac{\mathbf{A}_{ij}}{\sum_{k \in N^{C_l}(i)} \mathbf{A}_{ik}} \qquad \text{for all } i \in F,\ j \in N^{C_l}(i). \tag{5.35}$$

Since $\mathbf{A}_{ij} < 0$ when $j \in N^{C_l}(i)$, then dividing (5.35) by $-\mathbf{A}_{ij}$ yields

$$\beta_g \geq -\frac{\mathbf{A}_{ii}}{\sum_{k \in N^{C_l}(i)} \mathbf{A}_{ik}} = \frac{\mathbf{A}_{ii}}{\sum_{k \in N^{C_l}(i)} (-\mathbf{A}_{ik})} = \frac{\mathbf{A}_{ii}}{\sum_{k \in N^{C_l}(i)} |\mathbf{A}_{ik}|} \qquad \text{for all } i \in F. \tag{5.36}$$

Now, $\mathbf{A}$ being diagonally dominant implies, together with (5.36), that

$$\beta_g \geq \frac{\mathbf{A}_{ii}}{\sum_{k \in N^{C_l}(i)} |\mathbf{A}_{ik}|} \geq \frac{\sum_{\substack{k=1 \\ k \neq i}}^{n} |\mathbf{A}_{ik}|}{\sum_{k \in N^{C_l}(i)} |\mathbf{A}_{ik}|} \geq 1 \qquad \text{for all } i \in F. \tag{5.37}$$

5.1.3. Convergence Bound for the Two-Grid AMG Operator

Combining (5.26) and (5.37) for the smoothing and approximation assumptions, we have the following:

Proposition 5.8 *Suppose* $\mathbf{A}$ *is a SPD and weakly diagonally dominant matrix with* $\mathbf{A}_{ij} \leq 0$ *for all* $i = 1, \ldots, n,\ j \neq i$. *Consider a two-level V(0,1)-cycle setting constructed with a relaxation-based coarsening. Suppose that the prolongation operator* P *satisfies*

$$P_{ij} \geq 0 \quad \text{for all } i = 1, \ldots, n, \quad j \in N^{C_l}(i) \qquad \text{and} \qquad \sum_{j \in N^{C_l}(i)} P_{ij} = 1 \quad \text{for all } i = 1, \ldots, n,$$

and the post-smoothing operator S *is Gauss-Seidel iteration and satisfies the smoothing assumption*

$$\|S\mathbf{e}\|_{\mathbf{A}}^2 \leq \|\mathbf{e}\|_{\mathbf{A}}^2 - \alpha_g \|\mathbf{e}\|_{\mathbf{AD}^{-1}\mathbf{A}}^2$$

with $\alpha_g \leq \frac{1}{4}$. *Then, the coarse-grid operator* $T = \mathbf{I} - P\left(P^t\mathbf{A}P\right)^{-1}P^t\mathbf{A}$ *satisfies*

$$\inf_{\mathbf{e}_C} \|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{D}}^2 \leq \beta_g \|\mathbf{e}\|_{\mathbf{A}}^2$$

with $\beta_g \geq 1$ *and the two-level V(0,1)-cycle operator* ST *satisfies*

$$\|ST\|_{\mathbf{A}} \leq \sqrt{1 - \frac{\alpha_g}{\beta_g}}. \tag{5.38}$$

5.2. Linear Problem with a Complex Operator

We now consider the complex-valued problem

$$\mathbf{A}\mathbf{u} = \mathbf{f} \qquad \mathbf{A} \in \mathbb{C}^{n\times n}, \quad \mathbf{u}, \mathbf{f} \in \mathbb{C}^n. \tag{5.39}$$

As in Section 5.1, $\mathbf{u}$ is associated to the set of indices $\Omega = \{1, \ldots, n\}$ and a two-level scheme is constructed by partitioning Ω into subsets F and C. The complex prolongation operator $P : C \to \Omega$ is assumed to be full-rank. To describe how the restriction operator R is selected, we recall that a matrix $\mathbf{M} \in \mathbb{C}^{n\times n}$ is said to be Hermitian positive definite if $\mathbf{M}^\star = \mathbf{M}$ and $\langle \mathbf{M}\mathbf{v}, \mathbf{v}\rangle > 0$ for all $\mathbf{v} \neq \mathbf{0}$. Here, $\mathbf{M}^\star$ denotes the adjoint (i.e., the complex conjugate transpose) of $\mathbf{M}$. If $\mathbf{A}$ in (5.39) is Hermitian positive definite, then R is selected as $R = P^\star$. For more general matrices, the restriction operator R is chosen to be the adjoint of $\hat{P}$, the interpolation operator associated with $\mathbf{A}^\star$; i.e., $R = \hat{P}^\star$ ([35]). Note that this choice of R coincides with the case when $\mathbf{A}$ is Hermitian positive definite since $\mathbf{A} = \mathbf{A}^\star$.

The complex two-grid error propagation operator is

$$T = \mathbf{I} - P\mathbf{A}_C^{-1}R\mathbf{A} \tag{5.40}$$

where $\mathbf{A}_C = R\mathbf{A}P$ is the coarse-grid approximation to the operator $\mathbf{A}$. The post- and pre-smoothing operators S and $\hat{S}$ are also expressed as in (5.2).

Given the way R is chosen, the convergence analysis of the two-grid V-cycle is discussed considering two cases. We consider only the case when $\mathbf{A}$ is Hermitian positive definite. Then R is chosen to be $R = P^\star$ and the inner products and norms given in (5.4)-(5.7) can be redefined on $\mathbb{C}^n$ by means of the Euclidean inner product

$$\langle \mathbf{u}, \mathbf{v}\rangle = \sum_{i=1}^{n} \overline{\mathbf{u}}_i \mathbf{v}_i \tag{5.41}$$

where the bar denotes complex conjugation.

5.2.1. Hermitian Positive Definite Case

For the two-grid scheme this implies that the $\mathbf{A}$-adjoint of ST is $TS^{\dagger}$ where $S^{\dagger} = \mathbf{I} - \mathbf{Q}^{-\star}\mathbf{A}$, and the Hermitian two-grid V(1,1)-cycle operator can be written as $STS^{\dagger}$. Thus, as in the real case, the convergence of the two-grid V(0,1)-cycle operator ST leads to similar results for the two-grid V(1,0)- and V(1,1)-cycle operators. In addition, since the smoothing assumption (5.10) and the weak approximation assumption (5.11) are derived using only properties of the $\mathbf{A}$-norm, they can be extended to their analogous complex forms

$$\|S\mathbf{e}\|_{\mathbf{A}}^2 \leq \|\mathbf{e}\|_{\mathbf{A}}^2 - \alpha_g g(\mathbf{e}) \quad \text{for all } \mathbf{e} \tag{5.42}$$

and

$$\|T\mathbf{e}\|_{\mathbf{A}}^2 \leq \beta_g g(T\mathbf{e}) \quad \text{for all } \mathbf{e} \tag{5.43}$$

with $\alpha_g, \beta_g \in \mathbb{R}$, $\alpha_g, \beta_g > 0$ and $g(\mathbf{e})$ a real-valued nonnegative function. As a result, Theorem 5.2 can be also extended to the complex Hermitian case; i.e.,

$$\|ST\|_{\mathbf{A}} \leq \sqrt{1 - \frac{\alpha_g}{\beta_g}}. \tag{5.44}$$

Following the reasoning in Section 5.1, and using the $\mathbf{A}$-orthogonality between $\mathcal{R}(T)$ and $\mathcal{R}(P)$, the quantity $\|T\mathbf{e}\|_{\mathbf{A}}$ can be expressed in terms of P as

$$\|T\mathbf{e}\|_{\mathbf{A}} = \inf_{\mathbf{e}_C} \|\mathbf{e} - P\mathbf{e}_C\|_{\mathbf{A}}. \tag{5.45}$$

If $\mathbf{A}_{ii} > 0$ for all $i = 1, \ldots, n$, then by choosing $g(\mathbf{e}) = \|\mathbf{e}\|_{\mathbf{A}\mathbf{D}^{-1}\mathbf{A}}^2$ one obtains the alternative form of the weak approximation assumption (5.20).

Unfortunately, we have not yet determined an extension of the convergence from the real case to complex-valued problems.

5.3. Nonlinear Problem for a Real PDE Operator

Establishing convergence for FAS is nontrivial because of the complicated form of this algorithm. To get an idea on how convergence of FAS can be established, we consider its application to nonlinear elliptic PDEs. Let

$$\mathfrak{N}_{\Omega_h}(\mathbf{u}) = \mathbf{f}_{\Omega_h} \tag{5.46}$$

$$\mathfrak{N}_{\Gamma_h}(\mathbf{u}) = \mathbf{f}_{\Gamma_h} \tag{5.47}$$

be a system of nonlinear equations arising from a discretization of a boundary value problem. Here $\mathbf{u}, \mathbf{f}_{\Omega_h}, \mathbf{f}_{\Gamma_h} \in \mathbb{R}^n$, Ω_h is a resolution of the domain Ω, Γ_h is the resolution of the boundary of Ω, $\mathfrak{N}_{\Omega_h}$ is a discretization of a nonlinear PDE operator defined on Ω and $\mathfrak{N}_{\Gamma_h}$ is the discretization of the boundary conditions. Problem (5.46)-(5.47) leads to a system of nonlinear equations

$$\mathfrak{N}_h(\mathbf{u}) = \mathbf{0}. \tag{5.48}$$

Assume a two-level setting. At level l, problem (5.48) is written as

$$\mathfrak{N}_l(\mathbf{u}_l) = \mathbf{f}_l \qquad (l = 1, 2) \tag{5.49}$$

where $\mathbf{f}_1 = \mathbf{f}_2 = \mathbf{0}$. The number of unknowns at level l is denoted by n_l. Let the Jacobian of $\mathfrak{N}_l$ evaluated at $\mathbf{v}_l$ be denoted by $\mathbf{J}_l(\mathbf{v}_l)$. The solution of (5.49), if there exists one, is denoted as $\mathfrak{N}_l^{-1}(\mathbf{f}_l)$, and it is approximately computed using a two-grid FAS V(ν_1, ν_2)-cycle.

Algorithm 5.2 Two-Grid FAS V-cycle

$FAS^{(\nu_1,\nu_2)}(\mathbf{u}_1, \mathbf{f}_1)$

1: $\mathbf{u}_1 \leftarrow \mathfrak{S}_1^{(\nu_1)}(\mathbf{u}_1, \mathbf{f}_1)$

2: $d_2 \leftarrow R * [\mathbf{f}_1 - \mathfrak{N}_1(\mathbf{u}_1)]$

3: $\tilde{\mathbf{u}}_2 \leftarrow \hat{R}\mathbf{u}_1$

4: $\tilde{\mathbf{f}}_2 \leftarrow \mathfrak{N}_2(\tilde{\mathbf{u}}_2)$

5: $d_2 \leftarrow \tilde{\mathbf{f}}_2 + d_2$

6: $\mathbf{u}_2 \leftarrow \mathfrak{N}_2^{-1}(d_2)$

7: $\mathbf{u}_1 \leftarrow \mathbf{u}_1 + P * (\mathbf{u}_2 - \tilde{\mathbf{u}}_2)$

8: $\mathbf{u}_1 \leftarrow \mathfrak{S}_1^{(\nu_2)}(\mathbf{u}_1, \mathbf{f}_1)$

$\mathfrak{S}_l^{(\nu_1)}(\mathbf{u}_l, \mathbf{f}_l)$ and $\mathfrak{S}_l^{(\nu_2)}(\mathbf{u}_l, \mathbf{f}_l)$ correspond to smoothing steps at level l. This smoothing procedure consists of a nonlinear relaxation method (e.g., nonlinear Jacobi or nonlinear Gauss-Seidel iteration). Notice that Algorithm 5.2 is well defined if the following conditions are satisfied:

1. **CND1**: $\mathbf{u}_1$ in the right-hand side of Steps 1 and 8 lie in the region of attraction of $\mathfrak{S}_l^{(\nu_1)}$ and $\mathfrak{S}_l^{(\nu_2)}$,

2. **CND2**: $\mathbf{u}_1$ in the right-hand side of Step 2 is within the domain where Step 2 is well defined and

3. **CND3**: d_2 in Step 6 leads to a unique solution of $\mathfrak{N}_2(\mathbf{v}_2) = \mathbf{d}_2$.

Condition **CND3** is held if the hypothesis of the following proposition is fulfilled.

Proposition 5.9 (Remark 9.1.1 in [24]) *Let $\mathfrak{N}_l(\cdot)$ be continuous in a neighborhood of a solution $\mathbf{u}_l^*$ of (5.49). Suppose that $\mathbf{f}_l = \mathbf{0}$ and $\mathbf{J}_l(\mathbf{u}_l^*)$ is nonsingular. Then the Implicit Function Theorem implies that there are neighborhoods $N_{\mathbf{u}}(\mathbf{u}_l^*)$ of $\mathbf{u}_l^*$ and $N_{\mathbf{f}}(\mathbf{0})$ of $\mathbf{0}$ such that*

$$\mathfrak{N}_l|_{N_{\mathbf{u}}(\mathbf{u}_l^*)} : N_{\mathbf{u}}(\mathbf{u}_l^*) \to N_{\mathbf{f}}(\mathbf{0})$$

is a homeomorphism.

Thus, according to Proposition 5.9, Algorithm 5.2 is well-defined if $\mathbf{u}_1$ in the right-hand sides of Steps 1,2 and 8 of Algorithm 5.2 lie in $N_{\mathbf{u}}(\mathbf{u}_1^*)$, and d_2 from Step 6 to lie in $N_{\mathbf{f}}(\mathbf{0})$. In order to provide conditions for the latter, we need to specify the neighborhoods $N_{\mathbf{u}}(\mathbf{u}_l^*)$ and $N_{\mathbf{f}}(\mathbf{0})$. For an appropriate norm $\|\cdot\|$ defined on $N_{\mathbf{u}}(\mathbf{u}_l^*)$ there exists $\varepsilon_l > 0$ such that

$$N_{\mathbf{u}}(\mathbf{u}_l^*, \varrho_l) := \{\mathbf{u}_l : \|\mathbf{u}_l - \mathbf{u}_l^*\| \leq \varrho\} \tag{5.50}$$

satisfies

$$N_{\mathbf{u}}(\mathbf{u}_l^*, \varrho_l) \subset N_{\mathbf{u}}(\mathbf{u}_l^*) \quad \text{for all } 0 < \varrho \leq \varepsilon_l. \tag{5.51}$$

Similarly, $N_{\mathbf{f}}(\mathbf{0}, \varepsilon_l)$ is defined as

$$N_{\mathbf{f}}(\mathbf{0}, \varrho) := \mathfrak{N}_l(N_{\mathbf{u}}(\mathbf{u}_l^*, \varrho)) \subset N_{\mathbf{f}}(\mathbf{0}) \quad \text{for all } 0 < \varrho \leq \varepsilon_l. \tag{5.52}$$

To analyze the convergence of the two-grid FAS algorithm, we aim to show that the two-grid FAS iteration applied to problem (5.48) behaves asymptotically as the linear two-grid iteration applied to the linearized problem. In order to do so, divided differences $D\mathfrak{N}_l$ and $D\mathfrak{S}_l$ must be assigned to $\mathfrak{N}_l$ and $\mathfrak{S}_l^{(\nu)}$:

$$\mathfrak{N}_l(\mathbf{u}_l) - \mathfrak{N}_l(\mathbf{u}_l') = D\mathfrak{N}_l(\mathbf{u}_l, \mathbf{u}_l')(\mathbf{u}_l - \mathbf{u}_l'), \tag{5.53}$$

$$\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l, \mathbf{f}_l) - \mathfrak{S}_l^{(\nu)}(\mathbf{u}_l', \mathbf{f}_l) = D\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l, \mathbf{u}_l', \mathbf{f}_l)(\mathbf{u}_l - \mathbf{u}_l') \tag{5.54}$$

for all $\mathbf{u}_l, \mathbf{u}_l' \in N_{\mathbf{u}}(\mathbf{u}_l^*)$ and $\mathbf{f}_l \in N_{\mathbf{f}}(\mathbf{0})$.

We recall smoothing and approximation properties that the linear operators $D\mathfrak{N}_l(\mathbf{u}_l', \mathbf{u}_l)$ and $D\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l'', \mathbf{u}_l, \mathbf{f}_l)$ must satisfy.

Definition 5.2 *Suppose there exist functions $\eta(\nu)$ and $\tilde{\nu}(h)$, and a number α such that*

$$\|D\mathfrak{N}_l(\mathbf{u}_l', \mathbf{u}_l) D\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l'', \mathbf{u}_l, \mathbf{f}_l)\| \leq \eta(\nu) h_l^{-\alpha} \quad \textit{for all } 1 \leq \nu < \tilde{\nu}(h_l), l \geq 1, \tag{5.55}$$

$$\eta(\nu) \to 0 \text{ as } \nu \to \infty, \tag{5.56}$$

$$\tilde{\nu}(h) = \infty \quad \text{or} \quad \tilde{\nu}(h) \to \infty \text{ as } h \to \infty. \tag{5.57}$$

Then $D\mathfrak{S}_l^{(\nu)}$ is said to possess the smoothing property.

$D\mathfrak{N}_{l+1}\left(\mathbf{u}'_{l+1}, \mathbf{u}_{l+1}\right)$ is said to hold the approximation property if there is some constant C_A such that

$$\|D\mathfrak{N}_l^{-1}(\mathbf{u}'_l, \mathbf{u}_l) - P * D\mathfrak{N}_{l+1}^{-1}\left(\mathbf{u}'_{l+1}, \mathbf{u}_{l+1}\right) * R\| \leq C_A h_l^{\alpha} \quad \text{for all } l \geq 1. \tag{5.58}$$

The following result provides sufficient conditions for d_2 to lie in $N_{\mathbf{f}}(\mathbf{0})$:

Note 5.10 (Remark 9.5.3 in [24]) *Suppose for every $\mathbf{u}_l \in N_{\mathbf{u}}(\mathbf{u}_l^*)$ we have*

$$\left\| D\mathfrak{N}_2^{-1}(\mathbf{u}_2, \mathbf{u}_2^*) \left[\mathfrak{N}_2\left(\tilde{R} * \mathbf{u}_1\right) - R * \mathfrak{N}_1(\mathbf{u}_1) \right] \right\| \leq C h_2^{\alpha} \tag{5.59}$$

for all $\mathbf{u}_2 \in N_{\mathbf{u}}(\mathbf{u}_2^*, \varepsilon_2)$. Furthermore, suppose for $\tilde{\mathbf{u}}_2 \in N(\varepsilon_2/2)$, the condition

$$\left\| D\mathfrak{N}_2^{-1}(\mathbf{u}_2, \mathbf{u}'_2) \right\| \leq C_{DL} \qquad \text{for all } \mathbf{u}_2, \mathbf{u}'_2 \in N_{\mathbf{u}}(\mathbf{u}_2^*, \varepsilon_2)$$

holds for a constant C_{DL}, and $\|R\mathbf{f}_1\| * C_{DL} + C h_2^{\alpha} \leq \varepsilon_2$. Then $d_2 \in N_{\mathbf{f}}(\mathbf{0})$ for d_2 in Step 6 of Algorithm 5.2.

Assuming that these conditions hold for the two-grid FAS algorithm to be well-defined, we review its convergence. The convergence analysis for the two-grid FAS algorithm is based on its contraction number. Denote the approximate solution of the k-th iteration by $\mathbf{u}_l^k$. The iterative solver is said to be a contraction if there exists $\zeta_l < 1$ such that

$$\|\mathbf{u}_l^{k+1} - \mathfrak{N}_l^{-1}(\mathbf{f}_l)\| \leq \zeta_l \|\mathbf{u}_l^k - \mathfrak{N}_l^{-1}(\mathbf{f}_l)\| \quad \text{for } k = 0, 1, \ldots \tag{5.60}$$

Lemma 5.11 (Lemma 9.5.5 in [24]) *Let $\mathfrak{N}_1^{-1}(\mathbf{f}_1)$ be a fixed point of $\mathfrak{S}_1^{(\nu_i)}$, $(i = 1, 2)$. If the two-grid iteration described in Algorithm 5.2 is well-defined, then the error $\delta\mathbf{u}_1^k :=$*

$\mathbf{u}_1^k - \mathfrak{N}_1^{-1}(\mathbf{f}_1)$ *satisfies*

$$\delta\mathbf{u}_1^{k+1} = D\mathfrak{S}_1^{(\nu_2)}(\mathbf{u}_1'', \mathbf{u}_1, \mathbf{f}_1)\left[I - P * D\mathfrak{N}_2^{-1}(\mathbf{v}_2, \tilde{\mathbf{u}}_2) R * D\mathfrak{N}_1(\mathbf{u}_1', \mathbf{u}_1)\right] D\mathfrak{S}_1^{(\nu_1)}\left(\mathbf{u}_1^k, \mathbf{u}_1, \mathbf{f}_1\right) * \delta\mathbf{u}_1^k \tag{5.61}$$

where $\mathbf{u}_1 = \mathfrak{N}_1^{-1}(\mathbf{f}_1)$, $\mathbf{u}_1' = \mathfrak{S}_1^{(\nu_1)}\left(\mathbf{u}_1^k, \mathbf{f}_1\right)$, $\mathbf{v}_2 = \mathfrak{N}_2^{-1}(d_2)$, $d_2 = \tilde{\mathbf{f}}_2 + R * (\mathbf{f}_1 - \mathfrak{N}_1(\mathbf{u}_1'))$, $\mathbf{u}_1'' = \mathbf{u}_1' + P * (\mathbf{v}_2 - \tilde{\mathbf{u}}_2)$.

In order to prove the convergence of the two-grid FAS algorithm, it is more convenient to have conditions (5.55)-(5.57) and (5.58) satisfied for fixed arguments $\mathbf{u}_l$, $\mathbf{u}_l'$, $\mathbf{u}_l''$, $\mathbf{u}_{l+1}$, and $\mathbf{u}_l'$, instead of having these conditions defined on the sets $N_{\mathbf{u}}(\mathbf{u}_l^*)$ and $N_{\mathbf{u}}\left(\mathbf{u}_{l+1}^*\right)$. In order to determine these fixed arguments notice that, if $\mathfrak{N}_l$ and $\mathfrak{S}_l^{(\nu)}$ are continuously differentiable at $\mathbf{u}_l^*$, then the derivatives

$$\frac{\partial}{\partial\mathbf{u}_l}\mathfrak{N}_l(\mathbf{u}_l^*) \quad \text{and} \quad \frac{\partial}{\partial\mathbf{u}_l}\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l^*, \mathbf{0})$$

are approximated by $D\mathfrak{N}_l(\mathbf{u}_l, \mathbf{u}_l')$ and $D\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l, \mathbf{u}_l', \mathbf{f}_l)$ respectively as $\mathbf{u}_l, \mathbf{u}_l' \to \mathbf{u}_l^*$ and $\mathbf{f}_l \to \mathbf{0}$. This is formally stated by requiring

$$\left\| D\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l, \mathbf{u}_l', \mathbf{f}_l) - \frac{\partial}{\partial\mathbf{u}_l}\mathfrak{S}_l^{(\nu)}(\mathbf{u}_l^*, \mathbf{0}) \right\| \leq \Theta(\varrho_l/\varepsilon_l) \quad \text{for all } \mathbf{u}_l', \mathbf{u}_l \in N_{\mathbf{u}}(\mathbf{u}_l^*, \varrho_l),\ \mathbf{f}_l \in N_{\mathbf{f}}(\mathbf{0}, \varrho_l), \tag{5.62}$$

and

$$\left\| \left[\frac{\partial}{\partial\mathbf{u}_l}\mathfrak{N}_l(\mathbf{u}_l^*)\right]^{-1} D\mathfrak{N}_l(\mathbf{u}_l, \mathbf{u}_l') - \mathbf{I} \right\| \leq \Theta(\varrho_l/\varepsilon_l) \quad \text{for all } \mathbf{u}_l', \mathbf{u}_l \in N_{\mathbf{u}}(\mathbf{u}_l^*, \varrho_l) \tag{5.63}$$

where

$$\Theta(\varepsilon) \searrow 0 \ \text{ as } \ \varepsilon \searrow 0. \tag{5.64}$$

Finally, Proposition 5.12 states conditions for the convergence of the two-grid FAS algorithm.

Proposition 5.12 (Proposition 9.5.7 in [24]: convergence of $FAS^{(\nu,0)}$) *Suppose*

(i) for $0 < \nu < \tilde{\nu}(h_1)$ *we have*

$$\| D\mathfrak{N}_1(\mathbf{u}_1', \mathbf{u}_1) D\mathfrak{S}_1^{(\nu)}(\mathbf{u}_1'', \mathbf{u}_1, \mathbf{f}_1) \| \leq C_S, \tag{5.65}$$

(ii) the smoothing property (5.55)-(5.57) and the approximation property (5.58) are satisfied by $\frac{\partial}{\partial \mathbf{u}_1}\mathfrak{N}_1(\mathbf{u}_1^*)$, $\frac{\partial}{\partial \mathbf{u}_2}\mathfrak{N}_2(\mathbf{u}_2^*)$ *and* $\frac{\partial}{\partial \mathbf{u}_1}\mathfrak{S}_1^{(\nu)}(\mathbf{u}_1^*, \mathbf{0})$,

(iii) conditions (5.62)-(5.64) are satisfied and

$$\underline{C}_P^{-1}\|\mathbf{u}_2\| \leq \|P\mathbf{u}_2\| \leq \overline{C}_P\|\mathbf{u}_2\| \qquad \text{for all } \mathbf{u}_2 \in N_{\mathbf{u}}(\mathbf{u}_2^*) \tag{5.66}$$

holds where

$$\varrho_1 \leq \varepsilon_1,\ \varrho_2 \leq \varepsilon_2,\ \tilde{\mathbf{u}}_2 \in N_{\mathbf{u}}(\mathbf{u}_2^*, \varrho_2/2),\ C_S\,\varrho_1 \leq \varepsilon_1, \tag{5.67}$$

and

(iv) we have

$$\mathbf{f}_1 = \mathbf{0} \text{ and } \mathbf{u}_1^0 \in N_{\mathbf{u}}(\mathbf{u}_1^*, \varrho_1) \tag{5.68}$$

or

$$\mathbf{f}_1 \in N_{\mathbf{f}}(\mathbf{0}, \varrho_1/3) \text{ and } \mathbf{u}_1^0 \in N_{\mathbf{u}}(\mathbf{u}_1^*, \varrho_1/3). \tag{5.69}$$

Then, for ϱ_1, ϱ_2 *sufficiently small the two-grid FAS algorithm satisfies (5.60) with*

$$\hat{\zeta} = \zeta(\varrho_1/\varepsilon_1, \varrho_2/\varepsilon_2, \nu) < 1$$

where

$$\zeta(\alpha, \beta, \nu) \to \left\| D\mathfrak{S}_1^{(\nu_2)}(\mathbf{u}_1'', \mathbf{u}_1, \mathbf{f}_1)\left[I - P * D\mathfrak{N}_2^{-1}(\mathbf{v}_2, \tilde{\mathbf{u}}_2) R * D\mathfrak{N}_1(\mathbf{u}_1', \mathbf{u}_1)\right] D\mathfrak{S}_1^{(\nu_1)}(\mathbf{u}_1^0, \mathbf{u}_1, \mathbf{f}_1)\right\|$$

as $\alpha, \beta \to 0$. *Here* $\mathbf{u}_1$, $\mathbf{u}_1'$, $\mathbf{u}_1''$ *and* $\tilde{\mathbf{u}}_2$ *are defined as in Lemma 5.11 and* $\mathbf{u}_1^0 \in N_{\mathbf{u}}(\mathbf{u}^*, \varrho_1)$.

5.3.1. Convergence of FAS Applied to More General Nonlinear Problems

If the nonlinear problem (5.48) is not associated to a PDE, then the convergence theory above needs to be modified to remove the dependencies on the grid structure. In particular, bounds for (5.55)-(5.57), (5.58), (5.59) need to be extended for h-independent problems. By observing the approximation property (5.58), we note that this can be achieved by finding a parameter ϵ_l such that

$$\|D\mathfrak{N}_l^{-1}(\mathbf{u}_l', \mathbf{u}_l) - P * D\mathfrak{N}_{l+1}^{-1}\left(\mathbf{u}_{l+1}', \mathbf{u}_{l+1}\right) * R\| \leq C_A \epsilon_l \tag{5.70}$$

for all $\mathbf{u}_l, \mathbf{u}_l' \in N_{\mathbf{u}}(\mathbf{u}_l^*)$; $\mathbf{u}_{l+1}, \mathbf{u}_{l+1}' \in N_{\mathbf{u}}\left(\mathbf{u}_{l+1}^*\right)$ and $l \geq 1$.

Chapter 6

Numerical Results

In this chapter the effectiveness of the affinity-based coarsening introduced in Section 3.2 is evaluated by comparing its performance with the classical Ruge-Stüben coarsening [41] on a linear complex-valued problem. Additionally, the performance of the AMG-FAS solver and the multiplicative correction scheme are examined by applying both solvers to some real world power systems. These systems considered here are widely used by the power grid community for testing purposes.

The data used in these tests has been obtained from several sources. The cases IEEE 57, 68, 118, 145 and 300 were taken from the Power System Test Case Archive of the University of Washington's College of Engineering [48]. Case IEEE 68 was extracted from [18]. The Illinois, South Carolina, Texas and Wisconsin cases are entirely synthetic. They were designed in algorithms described in [3] using the PowerWorld simulator. They are statistically similar to the actual transmission systems on the corresponding geographical regions. All these systems are symmetric (i.e., the involved admittance matrix is symmetric). In addition, several test cases that accurately represent the European high voltage transmission network and the Polish transmission network are considered. These European and Polish test suites are taken from MatPower [52]. The first European case describes a network with 1354 buses and operating at 380 and 220kV. The second European case corresponds to a network with 9241 buses operating at 750, 400, 380, 330, 220, 154, 150, 120, and 110 kV. The Polish cases represent networks with 2383, 2746 and 3375 buses, respectively, all operating at 400, 220 and 110 kV. These cases describe the operation of the Polish system under peak conditions during the winters of 1999-2000, 2003-2004 and 2007-2008, respectively. The 3375-bus Polish

system actually involves 3374 buses since one isolated bus has been removed.

6.1. Comparison Between Classical and Affinity-Based Coarsening Procedures

In this section the performance of AMG is tested with the classical Ruge-Stüben coarsening method [41] and the affinity-based method on a linear problem, where the operator corresponds to the admittance matrix of the synthetic power grid simulating the Texas network with 2007 buses and 3043 transmission lines. This linear system has the form

$$\mathbf{A}\mathbf{u} = \mathbf{f},$$

where $\mathbf{A}$ is not diagonally dominant (which affects the effectiveness of the classical coarsening scheme), and $f = \mathbf{0}$ to avoid any dependence of the solution on the right-hand side as we are interested in the convergence of the method. The solution is iteratively computed starting from a random initial guess $\mathbf{u}^0$.

A 5-level grid hierarchy was constructed using the two coarsenings, and the system is solved by applying 20 V(1,1)-cycles. For the classical case, the parameter θ were taken from the interval $[0.5, 1)$ so that, in average, at most a half of the nodes in the fine grid become candidates for the formation of the coarse grid. For the affinity-based coarsening, 10 test vectors and 10 smoothing sweeps were used in the coarsening procedure, and the parameters Q and σ were set initially to 0.75 and 1.5, respectively. In addition a caliber-3 interpolation was used. Each parameter were changed to observe how they affect the performance of the resulting AMG schemes.

Table 6.1 shows that the coarse grids for classical AMG get smaller as θ approaches 1. The number of nonzero entries of $\mathbf{A}$ and the interpolation P reduced significantly for values of $\theta > 0.5$. However, as θ approaches 1, the grid size decreases too rapidly. This rapid coarsening leads to a loss of information from one level to the next coarser level. Table 6.2 illustrates the poor convergence of AMG with the classical coarsening using a V(1,1)-cycles.

$\theta = 0.5$			$\theta = 0.75$			$\theta = 0.875$		
Level	Avg value		Level	Avg value		Level	Avg value	
1	size=	2007.0	1	size=	2007.0	1	size=	2007.0
	nnz(A)=	7221.0		nnz(A)=	7221.0		nnz(A)=	7221.0
	nnz(P)=	–		nnz(P)=	–		nnz(P)=	–
2	size=	412.0	2	size=	214.0	2	size=	89.0
	nnz(A)=	1776.0		nnz(A)=	630.0		nnz(A)=	215.0
	nnz(P)=	1313.0		nnz(P)=	701.0		nnz(P)=	299.0
3	size=	119.0	3	size=	24.0	3	size=	37.0
	nnz(A)=	601.0		nnz(A)=	50.0		nnz(A)=	59.0
	nnz(P)=	383.0		nnz(P)=	91.0		nnz(P)=	88.0
4	size=	49.0	4	size=	13.0	4	size=	28.0
	nnz(A)=	245.0		nnz(A)=	17.0		nnz(A)=	34.0
	nnz(P)=	149.0		nnz(P)=	24.0		nnz(P)=	36.0
5	size=	24.0	5	size=	12.0	5	size=	25.0
	nnz(A)=	100.0		nnz(A)=	16.0		nnz(A)=	29.0
	nnz(P)=	58.0		nnz(P)=	12.0		nnz(P)=	26.0

Table 6.1: Grid size at every level and number of nonzero entries in operators $\mathbf{A}$ and P for different values of θ using the classical coarsening procedure.

Turning to the affinity-based procedure, Table 6.3 illustrates the average grid size at the levels and the average number of nonzero entries of $\mathbf{A}$ and P. Table 6.6 gives the average residual norms after 20 V(1,1)-cycles. Table 6.3 shows the grid sizes and the number of nonzeros of $\mathbf{A}$ and P for different values of σ. Even though there is no considerable effect on the grid size and the number of nonzero entries in the coarse operators, Table 6.4 shows that the AMG solver performs better when σ is around 1.25. Note also how effectively the residual norm is reduced when the affinity-based coarsening is applied, with a rate of convergence close to $1/2$ when $\sigma = 1.25$.

$\theta = 0.5$		$\theta = 0.75$		$\theta = 0.875$	
V-cycles	Res. norm	V-cycles	Res. norm	V-cycles	Res. norm
0	1.914e+03	0	1.914e+03	0	1.914e+03
1	7.865e+02	1	1.041e+03	1	1.151e+03
2	2.804e+02	2	5.137e+02	2	6.124e+02
3	1.509e+02	3	3.565e+02	3	4.306e+02
4	9.613e+01	4	2.740e+02	4	3.247e+02
5	6.985e+01	5	2.212e+02	5	2.554e+02
6	5.515e+01	6	1.836e+02	6	2.071e+02
7	4.566e+01	7	1.553e+02	7	1.721e+02
8	3.885e+01	8	1.332e+02	8	1.456e+02
9	3.364e+01	9	1.154e+02	9	1.251e+02
10	2.948e+01	10	1.009e+02	10	1.088e+02
11	2.608e+01	11	8.887e+01	11	9.560e+01
12	2.325e+01	12	7.874e+01	12	8.471e+01
13	2.084e+01	13	7.013e+01	13	7.562e+01
14	1.879e+01	14	6.275e+01	14	6.793e+01
15	1.702e+01	15	5.637e+01	15	6.136e+01
16	1.548e+01	16	5.083e+01	16	5.570e+01
17	1.413e+01	17	4.598e+01	17	5.077e+01
18	1.296e+01	18	4.172e+01	18	4.646e+01
19	1.192e+01	19	3.795e+01	19	4.266e+01
20	1.100e+01	20	3.461e+01	20	3.929e+01

Table 6.2: Residual norm for different values of θ after 20 consecutive V(1,1)-cycles using the classical coarsening procedure.

$\sigma = 1.25$			$\sigma = 1.5$			$\sigma = 1.75$		
Level	Avg value		Level	Avg value		Level	Avg value	
1	size=	2007.0	1	size=	2007.0	1	size=	2007.0
	nnz(A)=	7221.0		nnz(A)=	7221.0		nnz(A)=	7221.0
	nnz(P)=	–		nnz(P)=	–		nnz(P)=	–
2	size=	1038.8	2	size=	1037.5	2	size=	1037.9
	nnz(A)=	4551.6		nnz(A)=	4553.5		nnz(A)=	4552.9
	nnz(P)=	2719.1		nnz(P)=	2710.7		nnz(P)=	2711.7
3	size=	723.3	3	size=	717.9	3	size=	718.4
	nnz(A)=	3795.3		nnz(A)=	3794.5		nnz(A)=	3785.8
	nnz(P)=	1419.1		nnz(P)=	1427.4		nnz(P)=	1421.6
4	size=	531.0	4	size=	529.7	4	size=	530.4
	nnz(A)=	3424.0		nnz(A)=	3442.7		nnz(A)=	3430.4
	nnz(P)=	1012.7		nnz(P)=	1010.0		nnz(P)=	1003.3
5	size=	398.8	5	size=	398.3	5	size=	397.6
	nnz(A)=	3197.8		nnz(A)=	3199.5		nnz(A)=	3194.6
	nnz(P)=	753.6		nnz(P)=	756.7		nnz(P)=	756.9

Table 6.3: Grid size at every level and number of nonzero entries in operators $\mathbf{A}$ and P for different values of σ using the affinity-based coarsening procedure.

$\sigma = 1.25$		$\sigma = 1.5$		$\sigma = 1.75$	
V-cycles	Res. norm	V-cycles	Res. norm	V-cycles	Res. norm
0	2.101e+03	0	2.101e+03	0	2.101e+03
1	3.520e+02	1	3.557e+02	1	3.553e+02
2	8.860e+00	2	1.088e+01	2	1.036e+01
3	9.235e-01	3	1.659e+00	3	1.539e+00
4	2.191e-01	4	4.496e-01	4	4.747e-01
5	6.488e-02	5	1.370e-01	5	1.811e-01
6	2.144e-02	6	4.417e-02	6	7.848e-02
7	7.502e-03	7	1.485e-02	7	3.746e-02
8	2.709e-03	8	5.205e-03	8	1.940e-02
9	9.978e-04	9	1.901e-03	9	1.076e-02
10	3.723e-04	10	7.249e-04	10	6.295e-03
11	1.402e-04	11	2.889e-04	11	3.831e-03
12	5.321e-05	12	1.206e-04	12	2.396e-03
13	2.032e-05	13	5.268e-05	13	1.526e-03
14	7.803e-06	14	2.405e-05	14	9.828e-04
15	3.011e-06	15	1.142e-05	15	6.376e-04
16	1.167e-06	16	5.603e-06	16	4.156e-04
17	4.542e-07	17	2.823e-06	17	2.716e-04
18	1.774e-07	18	1.452e-06	18	1.779e-04
19	6.956e-08	19	7.582e-07	19	1.166e-04
20	2.736e-08	20	4.005e-07	20	7.647e-05

Table 6.4: Residual norm for different values of σ after 20 consecutive V(1,1)-cycles using affinity-based coarsening procedure.

Changes in Q more strongly affect the size of the coarse grids and the number of nonzero entries in $\mathbf{A}$ and P, as can be observed in Table 6.5. As the value of Q increases, the grid becomes larger by allowing more F-nodes to turn into C-nodes during the second pass of the coarsening algorithm. Table 6.6 shows that larger values of Q lead to an improvement in the rate of convergence since the coarse grid captures more features of the grid in the previous level.

$Q = 0.25$			$Q = 0.5$			$Q = 0.75$		
Level	Avg value		Level	Avg value		Level	Avg value	
1	size=	2007.0	1	size=	2007.0	1	size=	2007.0
	nnz(A)=	7221.0		nnz(A)=	7221.0		nnz(A)=	7221.0
	nnz(P)=	–		nnz(P)=	–		nnz(P)=	–
2	size=	735.0	2	size=	913.5	2	size=	1039.0
	nnz(A)=	4149.8		nnz(A)=	4508.1		nnz(A)=	4550.8
	nnz(P)=	2734.2		nnz(P)=	2809.9		nnz(P)=	2720.3
3	size=	285.2	3	size=	517.6	3	size=	721.0
	nnz(A)=	2807.8		nnz(A)=	3627.8		nnz(A)=	3787.4
	nnz(P)=	1259.4		nnz(P)=	1443.6		nnz(P)=	1421.8
4	size=	107.4	4	size=	301.6	4	size=	531.4
	nnz(A)=	1525.8		nnz(A)=	3080.0		nnz(A)=	3431.8
	nnz(P)=	563.9		nnz(P)=	876.4		nnz(P)=	1012.0
5	size=	36.1	5	size=	172.8	5	size=	397.3
	nnz(A)=	546.3		nnz(A)=	2418.6		nnz(A)=	3191.7
	nnz(P)=	242.1		nnz(P)=	537.0		nnz(P)=	759.9

Table 6.5: Grid size at every level and number of nonzero entries in operators $\mathbf{A}$ and P for different values of Q using the affinity-based coarsening procedure.

$Q = 0.25$		$Q = 0.5$		$Q = 0.75$	
V-cycles	Res. norm	V-cycles	Res. norm	V-cycles	Res. norm
0	1.895e+03	0	1.895e+03	0	1.895e+03
1	4.891e+02	1	4.017e+02	1	3.012e+02
2	5.692e+01	2	2.736e+01	2	1.047e+01
3	1.662e+01	3	3.202e+00	3	1.661e+00
4	6.645e+00	4	7.656e-01	4	4.946e-01
5	3.122e+00	5	3.003e-01	5	1.728e-01
6	1.673e+00	6	1.412e-01	6	6.819e-02
7	1.009e+00	7	7.183e-02	7	3.028e-02
8	6.752e-01	8	3.797e-02	8	1.502e-02
9	4.882e-01	9	2.049e-02	9	8.192e-03
10	3.711e-01	10	1.118e-02	10	4.792e-03
11	2.899e-01	11	6.149e-03	11	2.938e-03
12	2.295e-01	12	3.399e-03	12	1.855e-03
13	1.830e-01	13	1.888e-03	13	1.191e-03
14	1.469e-01	14	1.053e-03	14	7.726e-04
15	1.186e-01	15	5.890e-04	15	5.040e-04
16	9.610e-02	16	3.308e-04	16	3.298e-04
17	7.784e-02	17	1.863e-04	17	2.162e-04
18	6.300e-02	18	1.053e-04	18	1.419e-04
19	5.104e-02	19	5.968e-05	19	9.321e-05
20	4.149e-02	20	3.392e-05	20	6.123e-05

Table 6.6: Residual norm for different values of Q after 20 consecutive V(1,1)-cycles using affinity-based coarsening procedure.

The previous experiment demonstrates that, when the operator $\mathbf{A}$ is not an M-matrix, using the classical coarsening may lead to slow convergence of AMG. As noted in Table 6.1, the coarsening is too aggressive. Hence, the overall performance of classical AMG is poor when the matrix fails to be an M-matrix. With the affinity-based coarsening, AMG shows good convergence. The coarsening is less aggressive for the affinity-based procedure, which may help improving the quality of the coarse-grid approximation for carefully chosen values of σ and Q.

6.1.1. Comparing AMG-FAS and Multiplicative Correction Schemes

In this section each power grid case is tested using the AMG-FAS and the multiplicative correction schemes with a five-level hierarchy and with starting guess $\mathbf{V}^0 = \mathbf{1}$. V(1,1) multigrid cycles are applied until the residual reduces by six orders of magnitude. Two sweeps of Newton's iteration are performed at the coarsest level. Relaxation-based coarsening with the parameters set to $\sigma = 1.5$, $Q = 0.75$, and a caliber 3 interpolation are used.

Table 6.7 shows some important features of the systems and the average number of iterations required to reach the stopping criterion using 10 simulations. Table 6.7 shows that the AMG-FAS performs better in most of the cases. Furthermore, since the coarse grid operators have to be recalculated after each cycle in the multiplicative correction scheme, the AMG-FAS scheme is computationally cheaper.

Case	Buses	Lines	Gens.	Transf./Ph. Shifters	FAS its.	Mult. MG its.
IEEE 57	57	80	7	15/0	11	10
IEEE 68	68	86	16	16/0	9	11
IEEE 118	118	186	54	9/0	9	12
IEEE 145	145	453	50	52/0	10	11
IEEE 300	300	411	69	62/0	32	48
Illinois 200	200	245	49	0/0	16	17
S. Carolina 500	500	597	90	0/0	17	19
Wisconsin 1664	1664	2462	78	0/0	15	22
Texas 2007	2007	3043	282	0/0	12	16
PEGASE 1354	1354	1991	260	234/6	20	20
Polish 2383	2383	2896	327	170/6	12	16
Polish 2746	2746	3514	382	171/1	16	20
Polish 3375*	3374	4161	441	383/2	21	24
PEGASE 9241*	9241	16049	1445	1319/66	24	28

Table 6.7: AMG-FAS and multigrid with multiplicative correction performances on real world power systems. The number of V(1,1) cycles is averaged over 10 simulations.
($*$) The original Polish 3375 and PEGASE 9241 systems have negative line resistances and/or reactances. These values arose from the π-representation of three-winding transformers. Both systems were modified to have positive resistances and reactances.
($*$) The 3375-bus Polish system actually involves 3374 buses since one isolated bus has been removed.

Chapter 7

General Form of the Admittance Matrix

Because of the important role the admittance matrix plays in the proposed nonlinear multigrid solvers, the derivation of this matrix is discussed in this chapter. This matrix provides the (complex) coupling coefficients in the power flow equations (3.9)-(3.11) and (3.12)-(3.14).

In order to derive the bus admittance matrix and formulate the power flow equations, note that every line $(i, j) \in E$ in the system has an impedance $\mathbf{z}_{ij}$ given by

$$\mathbf{z}_{ij} = \mathbf{r}_{ij} + \hat{\imath}\mathbf{x}_{ij}, \tag{7.1}$$

where $\mathbf{r}_{ij}$ and $\mathbf{x}_{ij}$ are real numbers called the series resistance and series reactance, respectively. $\mathbf{r}_{ij}$ and $\mathbf{x}_{ij}$ are generally non-negative for actual transmission lines [1]. However, representation of two-winding tap-changing transformers and three-winding transformers may lead to (small) negative resistances or reactances [28]. This is an unusual network condition that often leads to difficulties for Gauss-Seidel in solving the power flow equations.

Consider the voltage and the current injection ($\mathbf{V}_i$ and $\mathbf{I}_i$) at node i. At every node, the current injection may be either positive (into the node) or negative (out of the node). Voltages, currents and impedances are generally complex quantities. In order to simplify the computations, the magnitudes (moduli) of these quantities are expressed in per-unit or percent of specified base values. For instance, if 20kV is specified as base voltage, then 19kV corresponds to 19/20=0.95 per unit (p.u.). Calculations are made using per-unit quantities rather than dimensional quantities [20]. Use of the per-unit system can be thought as a normalization or rescaling of the quantities involved in a power flow model.

According to Ohm's law, the component of the current injection $\mathbf{I}_{ij}$ from node j to node i and the voltage difference $(\mathbf{V}_i - \mathbf{V}_j)$ are linked through the relationship

$$\mathbf{I}_{ij} = \frac{(\mathbf{V}_i - \mathbf{V}_j)}{\mathbf{z}_{ij}}, \tag{7.2}$$

which involves the impedance $\mathbf{z}_{ij}$. On the other hand, Kirchoff's current law states that the total current flow through node i to be equal to the sum of the currents flowing out and into node i. Therefore, using these two laws, the total current injected into node i can be written as

$$\mathbf{I}_i = \sum_{(i,j)\in E,\, j\neq i} \mathbf{I}_{ij} = \sum_{(i,j)\in E,\, j\neq i} \frac{(\mathbf{V}_i - \mathbf{V}_j)}{\mathbf{z}_{ij}}. \tag{7.3}$$

Rather than using the system of equations (7.3), one often rewrites this system by setting

$$\mathbf{y}_{ij} = \frac{1}{\mathbf{z}_{ij}} = \frac{1}{\mathbf{r}_{ij} + \hat{\imath}\mathbf{x}_{ij}} = \frac{\mathbf{r}_{ij}}{\mathbf{r}_{ij}^2 + \mathbf{x}_{ij}^2} - \hat{\imath}\frac{\mathbf{x}_{ij}}{\mathbf{r}_{ij}^2 + \mathbf{x}_{ij}^2} = \mathbf{G}_{ij} + \hat{\imath}\mathbf{B}_{ij}. \tag{7.4}$$

$\mathbf{G}_{ij}$ and $\mathbf{B}_{ij}$ are known as the conductance and the susceptance of the line $(i,j) \in E$, respectively. After replacing (7.4) into (7.3), it results in the system of *nodal network equations* (as $i = 1, \ldots, n$), introduced in Section 1.1 and given by

$$\mathbf{I}_i = \sum_{(i,j)\in E,\, j\neq i} (\mathbf{V}_i - \mathbf{V}_j)\, \mathbf{y}_{ij}, \quad i = 1, \ldots, n. \tag{7.5}$$

The quantity $\mathbf{y}_{ij}$ is called the admittance of the line $(i,j) \in E$. The system (7.5) may be rewritten as

$$\mathbf{I}_i = \left(\sum_{(i,j)\in E,\, j\neq i} \mathbf{y}_{ij} \right) \mathbf{V}_i + \sum_{(i,j)\in E,\, j\neq i} (-\mathbf{y}_{ij})\, \mathbf{V}_j, \quad i = 1, \ldots, n, \tag{7.6}$$

or more conveniently

$$\mathbf{I}_i = \sum_{j=1}^{n} \mathbf{Y}_{ij}^0 \mathbf{V}_j, \quad i = 1, \ldots, n, \tag{7.7}$$

where

$$\mathbf{Y}_{ij}^0 = \begin{cases} \sum_{(i,j)\in E,\, j\neq i} \mathbf{y}_{ij} & i = j \\ -\mathbf{y}_{ij} & (i,j) \in E, i \neq j \\ 0 & (i,j) \notin E \end{cases} . \tag{7.8}$$

By using (7.8), the system of nodal network equations (7.7) can be written also in matrix form as

$$\mathbf{I} = \mathbf{Y}^0 \mathbf{V}, \tag{7.9}$$

where $\mathbf{V} = (\mathbf{V}_1, \ldots, \mathbf{V}_n)^T$, $\mathbf{I} = (\mathbf{I}_1, \ldots, \mathbf{I}_n)^T$ and

$$\mathbf{Y}^0 = \begin{bmatrix} \mathbf{Y}_{11}^0 & \mathbf{Y}_{12}^0 & \ldots & \mathbf{Y}_{1n}^0 \\ \mathbf{Y}_{21}^0 & \mathbf{Y}_{22}^0 & \ldots & \mathbf{Y}_{2n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}_{n1}^0 & \mathbf{Y}_{n2}^0 & \ldots & \mathbf{Y}_{nn}^0 \end{bmatrix} . \tag{7.10}$$

Matrix $\mathbf{Y}^0$ is called the bus admittance matrix of the system, and (7.10) is its most basic form, which arises when only line admittances are involved.

OBSERVATIONS:

1. Usually the resistance of a transmission line is significantly less than the reactance. Hence, by looking at equation (7.4) it can be observed this implies that $\mathbf{G}_{ij}$ is very small compared to $\mathbf{B}_{ij}$. It is common practice to approximate the terms $\mathbf{y}_{ij}$ by neglecting its real part $\mathbf{G}_{ij}$. This leads to the simplified power flow problem (1.10)-(1.12).

2. From the way that the elements of the bus admittance matrix are defined in (7.8), it can be seen that $\mathbf{Y}^0$ is symmetric.

There are other factors in a power system that might affect the computation of the bus admittance matrix such as the length of the transmission lines, the presence of transformers and shunt components. Shunt elements cause current diversion through them and prevent current from flowing through other higher resistance components in a power system. These

elements are connected to the network at the nodes and are used primarily for compensation and stability purposes [1, 33].

Note that any transmission line is composed of one or more sections that are equivalent to a standard π model circuit (Figure 7.1) [1, 21], which is a simple circuit consisting of a series impedance $\mathbf{z}_{ij} = \frac{1}{\mathbf{y}_{ij}}$ between the two ends i and j and a shunt admittance $\hat{\imath}\mathbf{b}_{ij}$ that is halved and placed at each end of the circuit. Here, $\mathbf{b}_{ij}$ is a real non-negative number since this term corresponds to a capacitive reactance and is caused by the insulation of the transmission line from the ground. The difference in voltage between the line and the ground causes them to form a capacitor. This effect of capacitance between the wire and the ground is negligible for short transmission lines, but for medium-length and long transmission lines, the capacitance increases with the length of the line and its effect has to be considered for these types of lines [21].

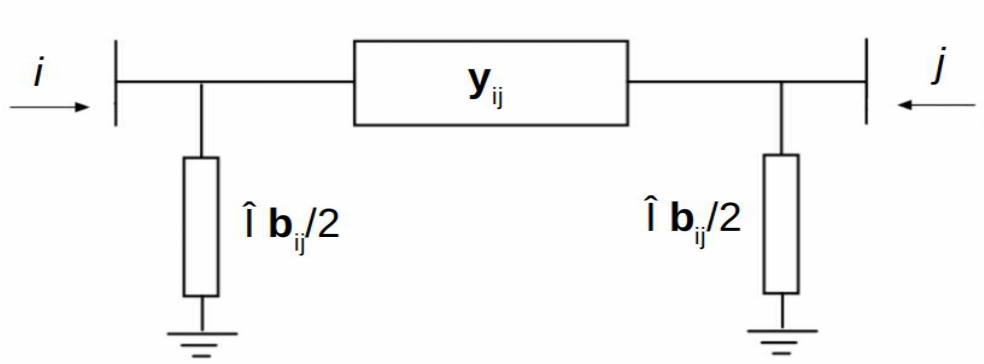


Figure 7.1: Standard π circuit model.

When nodes i and j are connected through a line with charging capacitance $\hat{\imath}\mathbf{b}_{ij}$, then the admittance matrix is modified by adding $\hat{\imath}\frac{\mathbf{b}_{ij}}{2}$ to the diagonal entries corresponding to

nodes i and j. The resulting form of the admittance matrix is

$$\mathbf{Y}^1 = \begin{bmatrix} \mathbf{Y}^0_{11} & \dots & \mathbf{Y}^0_{1i} & \dots & \mathbf{Y}^0_{1j} & \dots & \mathbf{Y}^0_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{Y}^0_{i1} & \dots & \mathbf{Y}^0_{ii} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2} & \dots & \mathbf{Y}^0_{ij} & \dots & \mathbf{Y}^0_{in} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{Y}^0_{j1} & \dots & \mathbf{Y}^0_{ji} & \dots & \mathbf{Y}^0_{jj} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2} & \dots & \mathbf{Y}^0_{jn} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}^0_{n1} & \dots & \mathbf{Y}^0_{ni} & \dots & \mathbf{Y}^0_{nj} & \dots & \mathbf{Y}^0_{nn} \end{bmatrix}. \tag{7.11}$$

On the other hand, tap setting transformers modify the actual magnitudes of voltages, currents and impedances from one end to the other of the lines [20]. When nodes i and j are connected through a single-phase two-winding ideal transformer, the corresponding p.u. quantities keep the same values on both sides of the transformer. However, in the case of a phase-shifting transformer, the phase angle of these quantities might differ from one side to the other. In order to model the action of an ideal transformer connecting nodes i and j with tap ratio τ and phase shift ϕ, the bus admittance matrix $\mathbf{Y}^1$ must be modified. The modification leads to [20]

$$\mathbf{Y}^2 = \begin{bmatrix} \mathbf{Y}^1_{11} & \dots & \mathbf{Y}^1_{1i} & \dots & \mathbf{Y}^1_{1j} & \dots & \mathbf{Y}^1_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{Y}^1_{i1} & \dots & \sum_{(i,k)\in E,\, k\neq j} \mathbf{y}_{ik} + \left(\mathbf{y}_{ij} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2}\right)\frac{1}{\tau^2} & \dots & -\mathbf{y}_{ij}\frac{1}{\tau\, e^{-\hat{\imath}\phi}} & \dots & \mathbf{Y}^1_{in} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{Y}^1_{j1} & \dots & -\mathbf{y}_{ji}\frac{1}{\tau\, e^{\hat{\imath}\phi}} & \dots & \sum_{(j,k)\in E} \mathbf{y}_{jk} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2} & \dots & \mathbf{Y}^1_{jn} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}^1_{n1} & \dots & \mathbf{Y}^1_{ni} & \dots & \mathbf{Y}^1_{nj} & \dots & \mathbf{Y}^1_{nn} \end{bmatrix}. \tag{7.12}$$

Finally, shunt elements such as capacitors and inductors might be connected to the network at the position of the nodes. A shunt admittance at node i is denoted by $\alpha_i + \hat{\imath}\beta_i$ and contributes to the corresponding diagonal term in the computation of the bus admittance

matrix [20]. By adding shunt admittances at nodes i and j, the bus admittance matrix becomes

$$\mathbf{Y}^3 = \begin{bmatrix} \mathbf{Y}^2_{11} & \dots & \mathbf{Y}^2_{1i} & \dots & \mathbf{Y}^2_{1j} & \dots & \mathbf{Y}^2_{1n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{Y}^2_{i1} & \dots & \sum_{(i,k)\in E, k\neq j} \mathbf{y}_{ik} + \left(\mathbf{y}_{ij} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2}\right)\frac{1}{\tau^2} + \alpha_i + \hat{\imath}\beta_i & \dots & -\mathbf{y}_{ij}\frac{1}{\tau e^{-\hat{\imath}\phi}} & \dots & \mathbf{Y}^2_{in} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{Y}^2_{j1} & \dots & -\mathbf{y}_{ji}\frac{1}{\tau e^{\hat{\imath}\phi}} & \dots & \sum_{(j,k)\in E} \mathbf{y}_{ik} + \hat{\imath}\frac{\mathbf{b}_{ij}}{2} + \alpha_j + \hat{\imath}\beta_j & \dots & \mathbf{Y}^2_{jn} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}^2_{n1} & \dots & \mathbf{Y}^2_{ni} & \dots & \mathbf{Y}^2_{nj} & \dots & \mathbf{Y}^2_{nn} \end{bmatrix}, \quad (7.13)$$

where α_i is a real non-negative number, while β_i can be either positive (when corresponding to a capacitive shunt, i.e. capacitors) or negative (when corresponding to an inductive shunt, i.e. reactors) [1]. In this document, $\mathbf{Y}$ denotes any one of the forms (7.10),(7.11), (7.12) or (7.13).

By expressing $\mathbf{Y}_{ij}$ in terms of its complex components as $\mathbf{Y}_{ij} = \mathbf{G}_{ij} + \hat{\imath}\mathbf{B}_{ij}$ and the voltage $\mathbf{V}_i$ in polar coordinates as $\mathbf{V}_i = |\mathbf{V}_i| e^{\hat{\imath}\delta_i}$, and replacing these expressions into (1.2) for each $i = 1, \dots, n$ the resulting system is the power flow equations given in (1.7)-(1.9).

REFERENCES

[1]Anderson, G. *Modelling and Analysis of Electric Power Systems*. ETH, 2008.

[2]Bakhvalov, N. On the convergence of a relaxation method with natural constraints on the elliptic operator. *USSR Computational Mathematics and Mathematical Physics 6*, 5 (1966), 101 – 135.

[3]Birchfield, A. B., Xu, T., Gegner, K. M., Shetye, K. S., and Overbye, T. J. Grid Structural Characteristics as validation criteria for synthetic networks. *IEEE Transactions on Power Systems 32*, 4 (July 2017), 3258–3265.

[4]Brandt, A. Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems. In *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics* (Berlin, Heidelberg, 1973), H. Cabannes and R. Temam, Eds., Springer Berlin Heidelberg, pp. 82–89.

[5]Brandt, A. Multi-level Adaptive Solutions to Boundary-Value Problems. *Mathematics of Computation 31*, 138 (1977), 333–390.

[6]Brandt, A. Guide to multigrid development. In *Multigrid Methods* (Berlin, Heidelberg, 1982), W. Hackbusch and U. Trottenberg, Eds., Springer Berlin Heidelberg, pp. 220–312.

[7]Brandt, A. The Gauss Center Research in Multiscale Scientific Computation, 1997.

[8]Brandt, A., Brannick, J., Kahl, K., and Livshits, A. An algebraic distances measure of AMG strength of connection. *arXiv preprint arXiv:1106.5990* (06 2011).

[9]Brandt, A., Brannick, J., Kahl, K., and Livshits, I. Bootstrap AMG. *SIAM Journal on Scientific Computing 33*, 2 (2011), 612–632.

[10]Brandt, A., McCormick, S., and Ruge, J. Algebraic Multigrid (AMG) for Automatic Multigrid Solution with Application to Geodetic Computations, Report. *Institute for Computational Studies, Fort Collins, CO* (1982).

[11]Brandt, A., McCormick, S., and Ruge, J. Algebraic multigrid (AMG) for sparse matrix equations. *Sparsity and its Applications 257* (1985).

[12]Brezina, M., Cleary, A. J., Falgout, R. D., Henson, V. E., Jones, J. E., Manteuffel, T. A., McCormick, S. F., and Ruge, J. W. Algebraic Multigrid Based on Element Interpolation (AMGe). *SIAM Journal on Scientific Computing 22*, 5 (2001), 1570–1592.

[13]Brezina, M., Falgout, R., MacLachlan, S., Manteuffel, T., McCormick, S., and Ruge, J. Adaptive Algebraic Multigrid. *SIAM Journal on Scientific Computing 27*, 4 (2006), 1261–1286.

[14]Briggs, W., Henson, V., and McCormick, S. *A Multigrid Tutorial*. SIAM, 2000.

[15]Cleary, A. J., Falgout, R. D., Henson, V. E., and Jones, J. E. Coarse-grid selection for parallel algebraic multigrid. In *Solving Irregularly Structured Problems in Parallel* (Berlin, Heidelberg, 1998), A. Ferreira, J. Rolim, H. Simon, and S.-H. Teng, Eds., Springer Berlin Heidelberg, pp. 104–115.

[16]Fedorenko, R. A relaxation method for solving elliptic difference equations. *USSR Computational Mathematics and Mathematical Physics 1*, 4 (1962), 1092 – 1096.

[17]Fedorenko, R. The speed of convergence of one iterative process. *USSR Computational Mathematics and Mathematical Physics 4*, 3 (1964), 227 – 235.

[18]General Electric Company, and United States Department of Energy. Office of Energy Systems Research. Singular Perturbations, Coherency and Aggregation of Dynamic Systems. *Report, Electric Utility Systems Engineering Department, General Electric Company, Schenectady* (1981).

[19]Glimn, A. F., and Stagg, G. W. Automatic Calculation of Load Flows. *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems 76*, 3 (1957), 817–825.

[20]Glover, J. D., Overbye, T. J., and Sarma, M. S. *Power System Analysis and Design*. Cengage Learning, 2017.

[21]Grainger, J., and Stevenson, W. *Power system analysis*. McGraw-Hill series in electrical and computer engineering: Power and energy. McGraw-Hill, 1994.

[22]Hackbusch, W. Convergence of Multi-Grid Iterations Applied to Difference Equations. *Mathematics of Computation 34* (05 1980), 425–440.

[23]Hackbusch, W. On the convergence of multi-grid iterations. *Beiträge zur Numerischen Mathematik 9* (01 1981), 213–239.

[24]Hackbusch, W. *Multigrid Methods and Applications*. Springer, 2003.

[25]Henson, V. E. Multigrid methods nonlinear problems: an overview. In *Computational Imaging* (06 2003), vol. 5016 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*.

[26]Idema, R., Lahaye, D., Vuik, C., and Sluis, L. Fast Newton load flow. In *2010 IEEE PES Transmission and Distribution Conference and Exposition: Smart Solutions for a Changing World* (May 2010), IEEE, pp. 1–7.

[27]Jameson, A. Multigrid Algorithms for Compressible Flow Calculations. In *Multigrid Methods II*. Springer, 1986, pp. 166–201.

[28]Josz, C., Fliscounakis, S., Maeght, J., and Panciatici, P. AC Power Flow Data in MATPOWER and QCQP Format: iTesla, RTE snapshots, and PEGASE, 2016.

[29]Köstler, H., Popa, C., Prümmer, M., and Rüde, U. Algebraic Full Multigrid in Image Reconstruction. In *Mathematical Modelling of Environmental and Life Sciences Problems* (01 2008), pp. 123–130.

[30]Lee, B. Multigrid for Model Reduction of Power Grid Networks. *Numer. Linear Algebra Appl.* (2018), e2201.

[31]Livne, O., and Brandt, A. Lean Algebraic Multigrid (LAMG): Fast Graph Laplacian Linear Solver. *SIAM J. Sci. Comput. 34*, 4 (August 2012), 407–423.

[32]Livne, O. E. Coarsening by compatible relaxation. *Numerical Linear Algebra with Applications 11*, 2-3 (2004), 205–227.

[33]Machowski, J., Bialek, J., and Bumby, J. *Power System Dynamics*. John Wiley and Sons, 2008.

[34]MacLachlan, S., and Olson, L. Theoretical bounds for algebraic multigrid performance: Review and analysis. *Numerical Linear Algebra with Applications 21* (03 2014).

[35]MacLachlan, S., and Oosterlee, C. Algebraic Multigrid Solvers for Complex-Valued Matrices. *SIAM J. Sci. Comput. 30* (January 2008), 1548–1571.

[36]McCormick, S. Multigrid Methods for Variational Problems: Further Results. *Siam Journal on Numerical Analysis - SIAM J NUMER ANAL 21* (04 1984), 255–263.

[37]Napov, A., and Notay, Y. Comparison of bounds for v-cycle multigrid. *Appl. Numer. Math 60*, 3 (2010), 176–192.

[38]Ortega, J. M., and Rheinboldt, W. C. *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.

[39]Ponce, C., Bindel, D., and Vassilevski, P. A Nonlinear Algebraic Multigrid Framework for the Power Flow Equations. *SIAM Journal on Scientific Computing 40* (01 2018), B812–B833.

[40]Ron, D., Safro, Y., and Brandt, A. Relaxation-based Coarsening and Multiscale Graph Organization. *SIAM J. Sci. Comput. 9* (2011), B499–B522.

[41]Ruge, J. W., and Stüben, K. Algebraic multigrid. In *Multigrid methods*. SIAM, 1987, pp. 73–130.

[42]Saad, Y. *Iterative methods for sparse linear systems*. SIAM, 2003.

[43]Stott, B. Review of Load-Flow Calculation Methods. *Proceedings of the IEEE 62*, 7 (1974), 916–929.

[44]Stüben, K. Algebraic Multigrid (AMG): An Introduction with Applications, March 1999.

[45]Stüben, K. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics 128*, 1 (2001), 281 – 309. Numerical Analysis 2000. Vol. VII: Partial Differential Equations.

[46]Tinney, W. F., and Hart, C. E. Power Flow Solution by Newton's Method. *IEEE Transactions on Power Apparatus and Systems PAS-86*, 11 (1967), 1449–1460.

[47]Trottenberg, U., and Schuller, A. *Multigrid*. Academic Press, Inc., USA, 2000.

[48]University of Washington. Power systems test case archive, 1999.

[49]Vaněk, P. Acceleration of convergence of a two-level algorithm by smoothing transfer operators. *Applications of Mathematics 37*, 4 (1992), 265–274.

[50]Ward, J., and Hale, H. W. Digital computer solution of power-flow problems [includes discussion]. *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems 75* (1956), 398–404.

[51]Yavneh, I., and Dardyk, G. A multilevel nonlinear method. *SIAM journal on scientific computing 28*, 1 (2006), 24–46.

[52]Zimmerman, R. D., Murillo-Sánchez, C. E., and Thomas, R. J. MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education. *IEEE Transactions on Power Systems 26*, 1 (Feb 2011), 12–19.