國立臺灣師範大學理學院數學系
碩士論文

Department of Mathematics
College of Science
National Taiwan Normal University
Master’s Thesis

計算最大特徵空間之結構可調控二次方法
An Adaptive Structure Partitioning Doubling Algorithm for Dominant Eigenspace

林威儒
Wei-Ju Lin

指導教授：謝世峰 博士
Advisor: Shih-Feng Shieh, Ph.D.

中華民國 114 年 6 月
June 2025

誌　謝

在此，我要向許多人獻上誠摯的感謝，許多師長與親友的幫助對本篇論文的完成至關重要。

首先，我要特別感謝我的指導教授謝世峰教授。謝教授自研究初期便提供我一份珍貴的筆記，作爲研究的起點，使整個研究工作得以建立在清晰而完整的架構之上。在研究過程中，每當我遇到困難與瓶頸時，謝教授總能適時地給予啓發性的建議。雖然這些建議往往不是直接的解答，卻爲我開啓了思考的方向，讓我有機會透過自己的努力一步步探索與完成。謝教授的專業知識與豐富經驗，讓我的研究內容更加充實，並對我在學術上的成長產生深遠的影響。

此外，我也誠摯感謝口試委員郭岳承教授與黃聰明教授撥冗審閱本論文，並提供寶貴的建議與指導。郭教授所提出的實驗面向建議極具實用性，能夠一針見血地指出實驗設計中尚待改進之處，使我能回頭檢視並補足不足，進而使實驗結果更加完整而具說服力；而黃教授則對論文的整體一致性與結構嚴謹度提出高度要求，強調各章節應有清楚的連貫與邏輯架構，並指出部分篇章在承接與過渡上仍有加強空間，讓我更加重視內容的整體性與知識脈絡的連結。兩位教授的建議皆令我獲益良多，對論文的品質提升有極大助益。

我也要感謝在求學過程中一路陪伴與支持我的朋友們。王品傑同學自大學以來便與我一同學習、成長，進入師大後仍繼續爲同學，並在我進行程式撰寫時盡力協助，甚至在我過於投入研究時，會適時提醒我注意休息，讓我能維持良好的學習狀態。吳宗螢同學是在師大課堂上認識的夥伴，無論我有任何研究或課業上的疑問，他總能耐心提供解答與建議，令我受益良多。此外，我也感謝室友們的陪伴與照顧，讓我能夠快速適應師大的環境，並共同維持一個良好舒適的生活空間；還有研究室的同學們，當我在研究上遇到困難時，總能彼此討論、互相幫助、互相打氣，是我能堅持下去的重要支撐。

最後，我要向我的父母表達最深的感謝之意。謝謝你們一直以來在背後默默支持我，給予我無條件的信任與包容，讓我能夠無後顧之憂地投入研究與論文撰寫，順利完成碩士課程與學位。這份支持是我一路走來最堅實的力量，衷心感謝你們一直以來的陪伴與鼓勵。

林威儒　於

國立臺灣師範大學數學系碩士班

中華民國 114 年 6 月

摘　要

本研究結合冪次法（power method）與源自結構保持倍增演算法（Structure-Preserving Doubling Algorithm, SDA）之倍增策略，以加速矩陣運算。儘管SDA 原爲求解離散時間代數Riccati 方程（DARE）所設計，其迭代框架在本研究中被改造應用於主特徵空間之提取。透過變形之倍增程序，可近似求得對應於主特徵值之特徵向量。然而，對任意矩陣而言，其收斂性未必得以保證。爲改善此一問題，本文引入一種受冪次法啓發之正規化策略：於每次迭代中以矩陣之範數進行縮放，以增強收斂行爲。因此，我們提出一種混合方法，稱爲倍增演算法（doubling algorithm）。另方面，爲保留有利於收斂之矩陣結構，本文進一步提出自適應區塊分割演算法（adaptive partition algorithm），透過適當區塊分割策略，在無需正規化情況下，仍可保證迭代過程之收斂性。上述兩種方法提供彈性且高效率之主特徵空間計算策略，並有助於提升迭代型矩陣演算法之整體穩定性。

關鍵字： 倍增演算法、標準形式、主特徵空間、自適應分割演算法、收斂分析

Abstract

In this study, we aim to accelerate matrix computations by integrating the power method with a doubling strategy originally developed in the structure-preserving doubling algorithm (SDA). While SDA was initially proposed for solving discrete-time algebraic Riccati equations (DAREs), we adapt its iterative framework to efficiently extract the dominant eigenspace of a given matrix. By applying a variant of the doubling process, we obtain approximations to the eigenvector corresponding to the dominant eigenvalue. However, convergence is not guaranteed for arbitrary matrices. To address this issue, we incorporate a normalization strategy inspired by the power method—scaling the matrix at each iteration by its norm—to enhance convergence behavior. As a result, we propose a hybrid approach, referred to as the *doubling algorithm*. Alternatively, by appropriately partitioning the matrix to preserve the structure responsible for convergence, we introduce a second method: the *adaptive partition algorithm*. In this approach, we apply the doubling algorithm without normalization, and show that, under a suitable block-partitioning strategy, the convergence of the iterative process can still be guaranteed. These two methods provide flexible and efficient strategies for computing dominant eigenspaces and improving the overall stability of iterative matrix algorithms.

Keywords: Doubling algorithm, standard form, dominant eigenspace, adaptive partitioning algorithm, convergence analysis

Contents

List of Tables

List of Figures

1 Introduction

The computation of dominant eigenspaces is a fundamental problem with wide-ranging applications in control theory, machine learning, and data science. Among various numerical methods, the structure-preserving doubling algorithm (SDA) has emerged as a powerful iterative framework—originally designed for solving discrete-time algebraic Riccati equations (DAREs)—but also offering valuable insights for eigenvalue-related computations due to its fast convergence and structural stability.

Initially developed for matrix equations possessing specific algebraic structures, SDA has garnered significant attention for its superior convergence behavior, often requiring far fewer iterations than classical methods such as the power method. Its effectiveness was first demonstrated in control theory, particularly in solving DAREs. These problems arise from the Linear Quadratic Regulator (LQR) framework, which is based on concepts such as controllability and observability [2]. While SDA was initially motivated by control-theoretic formulations, its iterative structure and stability properties make it a promising candidate for broader eigenvalue computations, such as dominant eigenspace extraction.

Further theoretical insights into SDA stem from its deep connections to Hamiltonian and symplectic matrices and their associated matrix pairs, as elaborated in [9]. In particular, this work establishes a foundational link via the Möbius transformation, a pivotal tool in structure-preserving computations. Building on this framework, [3] applies Möbius transformations to Hamiltonian matrices to construct symplectic pairs, which are subsequently brought into the first standard symplectic form (SSF-1). This transformation constitutes the initial step of the Structured Doubling Algorithm (SDA), marking the beginning of its iterative process and ensuring the preservation of essential matrix structure.

A detailed analysis of SDA's iterative process—including matrix squaring, block convergence, and numerical stability—is provided in [8]. This study demonstrates how SDA systematically preserves matrix structure across iterations, offering both theoretical convergence guarantees and practical efficiency. Despite these advances, most prior research has focused on SDA in the context of DAREs, with relatively little attention given to its potential application to general eigenspace computations.

In this work, we propose a novel framework—the *adaptive partitioning algorithm*—that generalizes the doubling strategy for computing dominant eigenspaces of arbitrary matrices. Our method constructs a matrix pair from the target matrix and the identity matrix, and applies a structure-aware partitioning scheme. Through careful analysis of submatrix convergence behavior during the iterations of the doubling algorithm, we show that this approach enables efficient extraction of the dominant left and right eigenvectors, thereby capturing the associated eigenspace.

The main contributions of this thesis are summarized as follows:

- We provide a theoretical analysis showing that the doubling algorithm converges to the dominant left and right eigenvectors when applied to irreducible and aperiodic stochastic matrices.
- We conduct a comparative study between the doubling algorithm and the power method on irreducible and aperiodic stochastic matrices, evaluating the number of iterations required to converge, the total computation time, and the approximation error with respect to the true eigenvectors.
- We extend the theoretical analysis to general (not necessarily stochastic) matrices and establish the convergence of the doubling algorithm to the dominant eigenspace. Furthermore, we derive the practical blockwise computational complexity associated with the proposed approach.
- We introduce a normalized variant of the doubling algorithm tailored for general matrices, which ensures convergence of each partitioned submatrix even in the absence of stochastic structure. Through numerical experiments, we observe and illustrate how the submatrices converge over iterations, providing empirical support for the effectiveness of the proposed approach.
- We propose the *adaptive partitioning algorithm*, which dynamically identifies a block configuration that accelerates convergence of the doubling iterations and reliably recovers the dominant eigenspace.

Organization. The remainder of this thesis is organized as follows. Section 2 introduces the Riccati equation and the Structure-Preserving Doubling Algorithm (SDA). Section 3 discusses stochastic matrices and the PageRank problem. Section 4 reviews the classical power method. Section 5 presents the doubling algorithm for arbitrary matrices and provides a convergence analysis of its submatrix components. Section 6 proposes the adaptive partition algorithm and examines how partition choices affect the approximation of the dominant eigenspace. Section 7 concludes the thesis and outlines future research directions. Appendix A provides MATLAB implementations of the algorithms and experiments for reproducibility and further reference.

2 Riccati Equation and SDA

In this section, we introduce a fundamental approach from control theory: the solution of the discrete-time algebraic Riccati equation (DARE). The DARE arises naturally in optimal control problems such as the Linear Quadratic Regulator (LQR), and its solution is often reformulated using a *Hamiltonian matrix*. However, numerical algorithms such as the Structure-Preserving Doubling Algorithm (SDA) operate more effectively on a specific type of matrix pair known as a *symplectic matrix pair*.

To bridge this gap, we employ the *Möbius transformation*, which converts the Hamiltonian matrix into a symplectic matrix pair while preserving its essential structural and spectral properties. In particular, this transformation maps eigenvalues located in the open left half-plane into the unit disk while preserving the corresponding eigenspaces. This facilitates the reduction of the matrix pair to the *first standard symplectic form (SSF-1)*, a canonical configuration that provides a numerically stable and structurally convenient framework for iterative methods such as the *Structure-Preserving Doubling Algorithm (SDA)*.

In this study, we extend these classical ideas beyond Riccati equations. We focus on a specific class of matrices, namely *irreducible and aperiodic stochastic matrices*, whose dominant eigenvalue lies on the unit circle while all others lie strictly inside it. This structure closely mirrors that of symplectic pairs and enables us to adopt similar transformation strategies. Consequently, we explore how to transform such stochastic matrices into an analogous *standard form* to preserve structural properties conducive to iterative algorithms.

Building on this idea, we adapt the *doubling technique*—originally developed for squaring symplectic matrix pairs—to the setting of stochastic matrices. We further extend this approach to *general matrices*, for which the absence of stochastic or symplectic structure introduces new theoretical and numerical challenges. Although the transformation still yields a structurally similar standard form, the *convergence of the doubling algorithm* is no longer theoretically guaranteed in such general cases.

To address this issue, we incorporate a *normalization strategy inspired by the power method*, which stabilizes the iteration and enhances numerical robustness. The concept of the power method not only informs the design of the normalization step but also plays a key role in the *convergence analysis* later in this study.

In summary, this chapter lays the theoretical foundation for the structure-preserving doubling algorithm by connecting classical control theory with modern numerical techniques. The subsequent chapters build upon this framework, examining its application to both stochastic and general matrices, presenting the normalization strategy, analyzing convergence behavior in depth, and introducing an adaptive partitioning algorithm tailored to improve computational efficiency.

2.1 Algebraic Riccati Equation (ARE)

In this subsection, we introduce the **Algebraic Riccati Equation (ARE)**, which arises naturally in control theory, particularly within the framework of the **Linear Quadratic Regulator (LQR)** problem. When applying a *state-feedback control* strategy, both the state and control variables are incorporated into a *quadratic cost function* that evaluates the performance of the control system. The primary objective is to minimize this cost function, which leads to the derivation of the ARE and the corresponding optimal feedback gain.

To begin with, we first discuss the Linear Quadratic Regulator (LQR) problem. In a closed-loop control system, there are two main components: the controller and the plant. The dynamics of the plant are typically described by the following state-space equations:

$$\dot{x} = Ax + Bu, \tag{2.1}$$

$$y = Cx + Du. \tag{2.2}$$

Here, $x \in \mathbb{R}^n$ represents the state vector, $u \in \mathbb{R}^m$ is the control input, and $y \in \mathbb{R}^p$ is the output. The matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, and $D \in \mathbb{R}^{p\times m}$ define the system dynamics. For simplicity, we consider the case where C is the identity matrix, i.e., $C = I$, and D is the zero matrix, i.e., $D = 0$. Under these assumptions, the equation (2.2) simplifies to $y = x$. From now on, we restrict our attention to the state equation (2.1).

In the LQR framework, we design a state-feedback control law of the form

$$u = -Kx, \tag{2.3}$$

where $K \in \mathbb{R}^{m\times n}$ is the feedback gain matrix. By substituting this control law into the state equation (2.1) yields the closed-loop system

$$\dot{x} = (A - BK)x.$$

The optimal feedback gain K is obtained by solving the algebraic Riccati equation (ARE), which plays a crucial role in optimal control. In the following discussion, we explore the derivation of the Riccati equation and its role in solving the LQR problem. We consider the following cost function:

$$J = \int_0^\infty \left(x^\top Q x + u^\top R u\right) dt, \tag{2.4}$$

subject to the system dynamics described by (2.1), where $Q \in \mathbb{R}^{n\times n}$ is a symmetric positive semi-definite matrix, and $R \in \mathbb{R}^{m\times m}$ is a symmetric positive definite matrix. Our objective is to determine the optimal control input u that minimizes the cost function J.

Theorem 2.1 ([2, Sec. 8.4 p. 330–331])**.** *[Algebraic Riccati Equation] Consider the linear system described by the state equation* (2.1)*. Suppose that there exists a symmetric positive semi-definite matrix P that satisfies the **Algebraic Riccati Equation (ARE)**:*

$$A^\top P + PA + Q - PBR^{-1}B^\top P = 0, \tag{2.5}$$

where Q and R are the same as defined earlier.

The optimal state feedback control law that minimizes the quadratic cost function (2.4) *is given by the equation* (2.3)*, where the feedback gain matrix K is defined as*

$$K = R^{-1}B^\top P. \tag{2.6}$$

Proof. Let $P = P^\top$ be a symmetric matrix. We rewrite the cost function J defined in equation (2.4) as follows:

$$J = x_0^\top P x_0 - x_0^\top P x_0 + \int_0^\infty \left(x^\top Q x + u^\top R u\right) dt. \tag{2.7}$$

Since the integral of a total derivative over an infinite time horizon evaluates to the boundary terms, we have

$$\int_0^\infty \frac{d}{dt}(x^\top P x)\, dt = \left[x^\top P x\right]_0^\infty = 0 - x_0^\top P x_0, \tag{2.8}$$

where x_0 represents the state vector at the initial time $t = 0$. Given that $x(t)$ asymptotically approaches zero as $t \to \infty$, only the initial condition contributes. As a consequence of equation (2.8), equation (2.7) can be reformulated as

$$J = x_0^\top P x_0 + \int_0^\infty \left(\frac{d}{dt}(x^\top P x) + x^\top Q x + u^\top R u\right) dt. \tag{2.9}$$

Next, by incorporating the system dynamics given by equation (2.1), we compute the total time derivative of $x^\top P x$ as follows:

$$\begin{aligned} \frac{d}{dt}(x^\top P x) &= \dot{x}^\top P x + x^\top P \dot{x} \\ &= (Ax + Bu)^\top P x + x^\top P (Ax + Bu) \\ &= x^\top A^\top P x + u^\top B^\top P x + x^\top P A x + x^\top P B u \end{aligned} \tag{2.10}$$

Substituting equation (2.10) into the cost function defined in equation (2.9), we obtain the following expression:

$$\begin{aligned} J &= x_0^\top P x_0 + \int_0^\infty \left[x^\top A^\top P x + u^\top B^\top P x + x^\top P A x + x^\top P B u + x^\top Q x + u^\top R u\right] dt \\ &= x_0^\top P x_0 + \int_0^\infty \left[x^\top (A^\top P + PA + Q) x + u^\top R u + x^\top P B u + u^\top B^\top P x\right] dt. \end{aligned} \tag{2.11}$$

Subsequently, we analyze the integral by splitting it into two parts, as indicated in equation (2.11). First, we focus on the terms that depend on the control input u, which form the following expression:

$$u^\top Ru + x^\top PBu + u^\top B^\top Px.$$

This can be rewritten in a quadratic form as follows:

$$(u + R^{-1}B^\top Px)^\top R(u + R^{-1}B^\top Px) - x^\top P^\top BR^{-1}B^\top Px. \tag{2.12}$$

The first term, $(u + R^{-1}B^\top Px)^\top R(u + R^{-1}B^\top Px)$, is always non-negative due to the positive definiteness of R. Therefore, we consider the second part of the integral, which consists of the remaining terms:

$$x^\top (A^\top P + PA + Q)x.$$

By combining this with the term $-x^\top P^\top BR^{-1}B^\top Px$ obtained from the previous equation (2.12), we obtain the complete quadratic form

$$x^\top (A^\top P + PA + Q - P^\top BR^{-1}B^\top P)x. \tag{2.13}$$

Therefore, our objective is to minimize the cost function. To minimize the cost function, we set both terms of the derived expression to zero, ensuring that the integrand becomes identically zero.

First, for the term involving u, setting

$$(u + R^{-1}B^\top Px)^\top R(u + R^{-1}B^\top Px) = 0$$

implies that

$$u = -R^{-1}B^\top Px. \tag{2.14}$$

Since we assume full-state feedback, the control law is given by equation (2.3). By equating equations (2.3) and (2.14), we obtain equation (2.6). Next, setting the remaining terms in equation (2.13) equal to zero yields equation (2.5). Hence, the proof is complete. $\square$

Theorem 2.1 involves the Algebraic Riccati Equation (ARE), which plays a fundamental role in optimal control theory. From the previous result, we obtained the ARE with P as its solution. Our objective is to solve this equation. To facilitate this, we introduce a new variable X to replace P and reformulate the equation in matrix form for further analysis. Rewriting the ARE, we express it in the following equivalent form:

$$\begin{aligned} O &= -XGX + A^H X + XA + H \\ &= \begin{bmatrix} -X & I \end{bmatrix} \begin{bmatrix} -A & G \\ H & A^H \end{bmatrix} \begin{bmatrix} I \\ X \end{bmatrix}, \end{aligned} \tag{2.15}$$

where $A, G, H \in \mathbb{C}^{n\times n}$, and G, H are Hermitian matrices, i.e., $G^H = G$ and $H^H = H$. This transformation allows us to analyze the structure of the equation in a matrix representation, providing insights into its solvability and solution properties.

2.2 Hamiltonian and Symplectic Relations

In this subsection, we begin by defining Hamiltonian and symplectic matrices, which play a central role in the analysis of Riccati equations and structure-preserving algorithms. We then show that the matrix arising from the algebraic Riccati equation (ARE) is Hamiltonian. In particular, we observe that one such Hamiltonian matrix corresponds to eigenvalues with positive real parts, while the other corresponds to those with negative real parts.

To construct a Möbius transformation that maps Hamiltonian matrices to symplectic matrices, we require that the transformation preserves eigenvectors and maps eigenvalues into the unit disk. However, since not all matrices involved are invertible, we generalize the framework using matrix pairs. This extension allows the Möbius transformation to remain applicable in singular cases and enables further analysis of the corresponding eigenvalues and eigenvectors.

Finally, we introduce two standard forms of symplectic matrix pairs. These canonical forms are fundamental to structure-preserving analysis, as they provide a foundation for designing numerical algorithms that maintain the symmetry and stability of the original system. Preserving such structural properties is essential for ensuring reliable and accurate numerical computations.

First, we introduce the definitions of Hamiltonian and symplectic matrices. We will present the specific conditions that a matrix must satisfy in order to be classified as Hamiltonian or symplectic.

Definition 2.1 ([1]). *A matrix $\mathcal{H} \in \mathbb{C}^{2n\times 2n}$ is said to be* ***Hamiltonian*** *if it satisfies*

$$(\mathcal{H}\mathcal{J})^H = \mathcal{H}\mathcal{J}, \quad \textit{i.e.,} \quad -\mathcal{J}\mathcal{H}^H = \mathcal{H}\mathcal{J}, \tag{2.16}$$

where

$$\mathcal{J} = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}.$$

Similarly, a matrix $\mathcal{S} \in \mathbb{C}^{2n\times 2n}$ is said to be ***symplectic*** *if it satisfies*

$$\mathcal{S}\mathcal{J}\mathcal{S}^H = \mathcal{J}. \tag{2.17}$$

From the previous subsection, we obtain the matrix form given in equation (2.15). We now focus on the middle matrix in this expression. For ease of discussion, we introduce the following notation:

$$\mathcal{H} = \begin{bmatrix} -A & G \\ H & A^H \end{bmatrix}, \quad \tilde{\mathcal{H}} = \begin{bmatrix} A & G \\ H & -A^H \end{bmatrix}. \tag{2.18}$$

However, we need to verify whether these matrices satisfy the condition given in equation (2.16). By performing direct computation, we obtain

$$(\mathcal{H}\mathcal{J})^H = \left(\begin{bmatrix} -A & G \\ H & A^H \end{bmatrix}\begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}\right)^H = \begin{bmatrix} -G & -A \\ -A^H & H \end{bmatrix}^H = \begin{bmatrix} -G & -A \\ -A^H & H \end{bmatrix} = \mathcal{H}\mathcal{J},$$
$$(\tilde{\mathcal{H}}\mathcal{J})^H = \left(\begin{bmatrix} A & G \\ H & -A^H \end{bmatrix}\begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}\right)^H = \begin{bmatrix} -G & A \\ A^H & H \end{bmatrix}^H = \begin{bmatrix} -G & A \\ A^H & H \end{bmatrix} = \tilde{\mathcal{H}}\mathcal{J}.$$

Since we have assumed that G and H are Hermitian matrices, i.e., $G^H = G$ and $H^H = H$, it follows from equation (2.16) that both $\mathcal{H}$ and $\tilde{\mathcal{H}}$ satisfy the defining property of a Hamiltonian matrix. Hence, we conclude that they are indeed Hamiltonian. Therefore, we investigate the relationship between the two Hamiltonian matrices $\mathcal{H}$ and $\tilde{\mathcal{H}}$. Since both $\mathcal{H}$ and $\tilde{\mathcal{H}}$ are matrices, we can analyze their eigenvalues and eigenvectors. In the following theorem, we establish a relationship between these two matrices using their properties.

Theorem 2.2. *The eigenvector $\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$ of the Hamiltonian matrix $\mathcal{H}$ with eigenvalue Λ satisfies the following equivalence:*

$$\mathcal{H}\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}\Lambda \iff \tilde{\mathcal{H}}\begin{bmatrix} Y_1 \\ -Y_2 \end{bmatrix} = \begin{bmatrix} Y_1 \\ -Y_2 \end{bmatrix}(-\Lambda) \tag{2.19}$$

Proof. We prove this theorem by direct computation. First, we compute the left-hand side. Since $\mathcal{H}$ is defined by equation (2.18), we have

$$\mathcal{H}\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} -A & G \\ H & A^H \end{bmatrix}\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} -AY_1 + GY_2 \\ HY_1 + A^HY_2 \end{bmatrix}.$$

By assumption, $\mathcal{H}\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}\Lambda$, which implies the equations

$$-AY_1 + GY_2 = Y_1\Lambda, \quad HY_1 + A^HY_2 = Y_2\Lambda.$$

Next, we compute the right-hand side. From the definition of $\tilde{\mathcal{H}}$, we obtain

$$\tilde{\mathcal{H}}\begin{bmatrix} Y_1 \\ -Y_2 \end{bmatrix} = \begin{bmatrix} A & G \\ H & -A^H \end{bmatrix}\begin{bmatrix} Y_1 \\ -Y_2 \end{bmatrix} = \begin{bmatrix} AY_1 - GY_2 \\ HY_1 + A^HY_2 \end{bmatrix}.$$

By using the previous equations, we get

$$\begin{bmatrix} AY_1 - GY_2 \\ HY_1 + A^HY_2 \end{bmatrix} = \begin{bmatrix} -Y_1\Lambda \\ Y_2\Lambda \end{bmatrix} = \begin{bmatrix} Y_1 \\ -Y_2 \end{bmatrix}(-\Lambda).$$

The reverse direction follows similarly. Hence, equation (2.19) is verified, and the proof is complete. □

From Theorem 2.2, we observe that the complex plane is partitioned according to the sign of the real parts of the eigenvalues. This segmentation is preserved under the Möbius transformation, which maps the right and left half-planes into distinct regions. Since this structural property is crucial to our analysis, we highlight it in the following remark.

Remark. *If* $\mathrm{Col}\begin{bmatrix} Y_1^* \\ Y_2^* \end{bmatrix}$ *is the eigenspace of* $\mathcal{H}$ *corresponding to all positive eigenvalues, then* $\mathrm{Col}\begin{bmatrix} Y_1^* \\ -Y_2^* \end{bmatrix}$ *is the eigenspace of* $\tilde{\mathcal{H}}$ *corresponding to all negative eigenvalues, and vice versa.*

Now, in the following discussion, we introduce a Möbius transformation for square matrices. Based on the previous remark, we know that the eigenspace of the Hamiltonian matrix $\tilde{\mathcal{H}}$ corresponds to all negative eigenvalues. By applying this Möbius transformation, these eigenvalues can be mapped into the unit disk.

Definition 2.2. *Let* $\gamma \in \mathbb{R}$ *with* $\gamma > 0$*. The Möbius transformation of a matrix* $X \in \mathbb{C}^{n\times n}$ *is defined as*

$$Y = C(X) = (X - \gamma I)^{-1}(X + \gamma I), \tag{2.20}$$

provided that $X - \gamma I$ *is nonsingular.*

Therefore, by the definition of the Möbius transformation, we establish in the next two theorems that this transformation preserves the eigenvectors and maps the eigenvalues into the unit disk. Moreover, the Möbius transformation has an inverse function.

Theorem 2.3. *Let* $X \in \mathbb{C}^{n\times n}$ *be a matrix with eigenvalue* λ *and corresponding eigenvector* v*. Then, under the Möbius transformation* $C(X)$*, we have*

$$Xv = \lambda v \quad \Longleftrightarrow \quad Yv = C(\lambda)v. \tag{2.21}$$

Proof. Suppose $Xv = \lambda v$. According to the definition of the Möbius transformation for matrix pairs, as given in equation (2.20), we have

$$Yv = C(X)v = (X - \gamma I)^{-1}(X + \gamma I)v.$$

Expanding the expression inside the parentheses, we obtain

$$Yv = (X - \gamma I)^{-1}(Xv + \gamma v) = (X - \gamma I)^{-1}(\lambda v + \gamma v) = (\lambda + \gamma)(X - \gamma I)^{-1}v.$$

Since $(X - \gamma I)v = (\lambda - \gamma)v$, it follows that

$$(X - \gamma I)^{-1}(X - \gamma I)v = (\lambda - \gamma)(X - \gamma I)^{-1}v \quad \Rightarrow \quad (X - \gamma I)^{-1}v = \frac{1}{\lambda - \gamma}v.$$

Therefore, we conclude that

$$Yv = \frac{\lambda + \gamma}{\lambda - \gamma}v = C(\lambda)v.$$

Conversely, if $Yv = C(\lambda)v$, we can apply the same steps in reverse order to obtain $Xv = \lambda v$. Thus, we have established equation (2.21). $\square$

Theorem 2.4. *Let* $\gamma > 0$*, and suppose that* $X \in \mathbb{C}^{n\times n}$ *is diagonalizable with eigenvalue* λ*. Then, for the Möbius transformation* $C(\lambda)$*, we have:*

- *If* $\mathrm{Re}(\lambda) < 0$*, then* $|C(\lambda)| < 1$.
- *If* $\mathrm{Re}(\lambda) > 0$*, then* $|C(\lambda)| > 1$.
- *The inverse transformation is given by*

$$C^{-1}(Y) = \gamma(Y - I)^{-1}(Y + I). \tag{2.22}$$

Proof. From the proof of Theorem 2.3, we have

$$C(\lambda) = \frac{\lambda + \gamma}{\lambda - \gamma}.$$

Combining the condition $\gamma > 0$ and $\mathrm{Re}(\lambda) < 0$, we get the inequality

$$|\lambda + \gamma| < |\lambda - \gamma|.$$

Thus, we conclude the first result

$$|C(\lambda)| = \left|\frac{\lambda + \gamma}{\lambda - \gamma}\right| < 1.$$

Similarly, for $\mathrm{Re}(\lambda) > 0$, we obtain $|C(\lambda)| > 1$. Finally, using equation (2.20), we compute directly

$$\begin{aligned} C^{-1}(Y) &= \gamma(Y-I)^{-1}(Y+I) \\ &= \gamma\left((X-\gamma I)^{-1}(X+\gamma I)-I\right)^{-1}\left((X-\gamma I)^{-1}(X+\gamma I)+I\right) \\ &= \gamma\left(2\gamma(X-\gamma I)^{-1}\right)^{-1}\left(2(X-\gamma I)^{-1}X\right) \\ &= X. \end{aligned}$$

Hence, we conclude that the inverse transformation is given by equation (2.22). □

In Theorems 2.3 and 2.4, we established several properties of the Möbius transformation. In the following theorem, we aim to demonstrate that this Möbius transformation indeed maps Hamiltonian matrices to symplectic matrices. Furthermore, while the Möbius transformation alters the eigenvalues of the original matrix, it preserves the associated eigenvectors and invariant subspaces.

Theorem 2.5. *The Möbius transformation preserves key structural relationships between Hamiltonian and symplectic matrices, as stated below:*

- *$\tilde{\mathcal{H}}$ is Hamiltonian $\Leftrightarrow$ $C(\tilde{\mathcal{H}})$ is symplectic.*
- *Suppose that $\tilde{\mathcal{H}}V = V\Lambda$, where $V \in \mathbb{C}^{2n\times m}$ and $\Lambda \in \mathbb{C}^{m\times m}$. Then*

$$C(\tilde{\mathcal{H}})V = VC(\Lambda).$$

Proof.

- When $\tilde{\mathcal{H}} - \gamma I$ is nonsingular,

$$\begin{aligned} & C(\tilde{\mathcal{H}}) \text{ is symplectic} & \\ \Longleftrightarrow\ & C(\tilde{\mathcal{H}})\mathcal{J}C(\tilde{\mathcal{H}})^H = \mathcal{J} & \text{(eq. (2.17))} \\ \Longleftrightarrow\ & (\tilde{\mathcal{H}}-\gamma I)^{-1}(\tilde{\mathcal{H}}+\gamma I)\mathcal{J}(\tilde{\mathcal{H}}+\gamma I)^H(\tilde{\mathcal{H}}-\gamma I)^{-H} = \mathcal{J} & \text{(eq. (2.20))} \\ \Longleftrightarrow\ & (\tilde{\mathcal{H}}+\gamma I)\mathcal{J}(\tilde{\mathcal{H}}+\gamma I)^H = (\tilde{\mathcal{H}}-\gamma I)\mathcal{J}(\tilde{\mathcal{H}}-\gamma I)^H & \text{(reformulate)} \\ \Longleftrightarrow\ & (\tilde{\mathcal{H}}\mathcal{J}+\gamma\mathcal{J})(\tilde{\mathcal{H}}^H+\gamma I) = (\tilde{\mathcal{H}}\mathcal{J}-\gamma\mathcal{J})(\tilde{\mathcal{H}}^H-\gamma I) & \text{(expand)} \\ \Longleftrightarrow\ & 2\gamma(\tilde{\mathcal{H}}\mathcal{J}+\mathcal{J}\tilde{\mathcal{H}}^H) = 0 & \text{(simplify)} \\ \Longleftrightarrow\ & \tilde{\mathcal{H}}\mathcal{J} = -\mathcal{J}\tilde{\mathcal{H}}^H & (\gamma{>}0) \\ \Longleftrightarrow\ & \tilde{\mathcal{H}} \text{ is Hamiltonian.} & \end{aligned}$$

- We now show that $C(\tilde{\mathcal{H}})V = VC(\Lambda)$:

$$\begin{aligned} C(\tilde{\mathcal{H}})V &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(\tilde{\mathcal{H}}+\gamma I)V & \text{(eq. (2.20))} \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(\tilde{\mathcal{H}}V+\gamma V) & \text{(expand)} \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(V\Lambda+\gamma V) & (\tilde{\mathcal{H}}V = V\Lambda) \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}V(\Lambda+\gamma I) & \text{(extract } V) \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}V(\Lambda-\gamma I)C(\Lambda) & \text{(eq. (2.20))} \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(V\Lambda-\gamma V)C(\Lambda) & \text{(expand)} \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(\tilde{\mathcal{H}}V-\gamma V)C(\Lambda) & (V\Lambda = \tilde{\mathcal{H}}V) \\ &= (\tilde{\mathcal{H}}-\gamma I)^{-1}(\tilde{\mathcal{H}}-\gamma I)VC(\Lambda) & \text{(extract } V) \\ &= VC(\Lambda). & \end{aligned}$$

□

In the previous discussion, we reviewed the definitions of Hamiltonian and symplectic matrices, as well as the concept of the Möbius transformation. This transformation establishes a connection between Hamiltonian and symplectic structures. It maps eigenvalues from the left half of the complex plane into the unit disk while preserving the associated eigenvectors. Since not all matrices are invertible, we now extend the Möbius transformation to matrix pairs. To do so, we first introduce the concept of a matrix pair along with its generalized eigenvalues and eigenvectors. The following definition formalizes this idea.

Definition 2.3. *Given $A, B \in \mathbb{C}^{n\times n}$, we find $\lambda \in \mathbb{C}$ such that there exists a nonzero vector $x \in \mathbb{C}^n$ satisfying*

$$Ax = \lambda Bx.$$

The vector x is called a generalized eigenvector *of the matrix pair (A, B) corresponding to the* generalized eigenvalue λ.

In the following two definitions, we introduce the pair form and the notions of matrix pairs, Hamiltonian pairs, and symplectic pairs. Since a matrix does not always have an inverse, performing the Möbius transformation on square matrices can be challenging. To deal with this difficulty, we utilize the pair representation. Additionally, we examine the relationship between Hamiltonian pairs and symplectic pairs.

Definition 2.4 ([9])**.** *Let $\gamma \in \mathbb{R}$ with $\gamma > 0$. The Möbius transformation of a matrix X is defined as*

$$(\mathcal{M}, \mathcal{L}) \equiv \mathcal{C}(X) \coloneqq (X + \gamma I, X - \gamma I)\,, \tag{2.23}$$

provided that $X - \gamma I$ is nonsingular. Equivalently, we can express this transformation as

$$\mathcal{C}(X) \overset{l.e.}{\sim} \left((X - \gamma I)^{-1}(X + \gamma I), I\right).$$

Definition 2.5 ([9])**.** *A matrix pair is defined as follows:*

- *A matrix pair $(\mathcal{M}_h, \mathcal{L}_h)$ is called a Hamiltonian pair if it satisfies*
$$\mathcal{M}_h \mathcal{J} \mathcal{L}_h^H = -\mathcal{L}_h \mathcal{J} \mathcal{M}_h^H. \tag{2.24}$$
- *A matrix pair $(\mathcal{M}_s, \mathcal{L}_s)$ is called a Symplectic pair if it satisfies*
$$\mathcal{M}_s \mathcal{J} \mathcal{M}_s^H = \mathcal{L}_s \mathcal{J} \mathcal{L}_s^H. \tag{2.25}$$

Based on the definitions, we aim to understand the relationship between Hamiltonian matrices and Hamiltonian pairs, as well as between symplectic matrices and symplectic pairs. In particular, we want to see how these structures are connected. Moreover, we note that the eigenvalues of the matrix pair exhibit characteristic symmetries. Therefore, we include the following remark to clarify how the matrix form and the pair form are related.

Remark ([9])**.** *The following statements obtained by the definition of the pair.*

- *If $(\mathcal{M}_h, \mathcal{L}_h)$ is a Hamiltonian pair and $\mathcal{L}_h$ is invertible, then $\mathcal{L}_h^{-1}\mathcal{M}_h$ is Hamiltonian.*
- *If λ is an eigenvalue of $(\mathcal{M}_h, \mathcal{L}_h)$, then so is $-\bar{\lambda}$.*
- *If $(\mathcal{M}_s, \mathcal{L}_s)$ is a symplectic pair and $\mathcal{L}_s$ is invertible, then $\mathcal{L}_s^{-1}\mathcal{M}_s$ is symplectic.*
- *If λ is an eigenvalue of $(\mathcal{M}_s, \mathcal{L}_s)$, then so is $\dfrac{1}{\bar{\lambda}}$.*

Proof. The proof of the above statement is given below.

- Since $(\mathcal{M}_h, \mathcal{L}_h)$ is a Hamiltonian pair, it satisfies the identity given in equation (2.24). Assuming $\mathcal{L}_h$ is invertible, we can multiply both sides on the left by $\mathcal{L}_h^{-1}$ and on the right by $\mathcal{L}_h^{-H}$ to obtain
$$\mathcal{L}_h^{-1}\mathcal{M}_h\mathcal{J} = -\mathcal{J}\mathcal{M}_h^H\mathcal{L}_h^{-H} = -\mathcal{J}\left(\mathcal{L}_h^{-1}\mathcal{M}_h\right)^H.$$
Therefore, the matrix $\mathcal{L}_h^{-1}\mathcal{M}_h$ satisfies the equation (2.16), and thus is Hamiltonian.
- By equation (2.24), the pair $(\mathcal{M}_h, \mathcal{L}_h)$ satisfies the Hamiltonian condition. Suppose that λ is a generalized eigenvalue of the pair $(\mathcal{M}_h, \mathcal{L}_h)$, with corresponding left eigenvector v, i.e.,
$$v\mathcal{M}_h = \lambda v \mathcal{L}_h.$$
Taking the Hermitian transpose of both sides yields
$$\mathcal{M}_h^H v^H = \bar{\lambda}\mathcal{L}_h^H v^H.$$

Multiplying both sides on the left by $\mathcal{L}_h\mathcal{J}$ and applying the Hamiltonian relation gives

$$\begin{aligned}&\mathcal{L}_h\mathcal{J}\mathcal{M}_h^H v^H = \bar{\lambda}\mathcal{L}_h\mathcal{J}\mathcal{L}_h^H v^H \\ \Rightarrow &-\mathcal{M}_h\mathcal{J}\mathcal{L}_h^H v^H = \bar{\lambda}\mathcal{L}_h\mathcal{J}\mathcal{L}_h^H v^H \qquad \text{(eq. (2.24))} \\ \Rightarrow &\mathcal{M}_h\mathcal{J}\mathcal{L}_h^H v^H = -\bar{\lambda}\mathcal{L}_h\mathcal{J}\mathcal{L}_h^H v^H.\end{aligned}$$

Letting $u = \mathcal{J}\mathcal{L}_h^H v^H$, we then obtain $\mathcal{M}_h u = -\bar{\lambda}\mathcal{L}_h u$, which shows that $-\bar{\lambda}$ is also a generalized eigenvalue of the pair $(\mathcal{M}_h, \mathcal{L}_h)$.

- Since $(\mathcal{M}_s, \mathcal{L}_s)$ is a symplectic pair, it satisfies the identity given in equation (2.25). Assuming that $\mathcal{L}_s$ is invertible, we can multiply both sides on the left by $\mathcal{L}_s^{-1}$ and on the right by $\mathcal{L}_s^{-H}$ to obtain

$$(\mathcal{L}_s^{-1}\mathcal{M}_s)\mathcal{J}(\mathcal{L}_s^{-1}\mathcal{M}_s)^H = \mathcal{L}_s^{-1}\mathcal{M}_s\mathcal{J}\mathcal{M}_s^H\mathcal{L}_s^{-H} = \mathcal{J}.$$

Therefore, the matrix $\mathcal{L}_s^{-1}\mathcal{M}_s$ satisfies the symplectic condition (2.17), and is thus symplectic.

- By equation (2.25), the pair $(\mathcal{M}_s, \mathcal{L}_s)$ satisfies the symplectic condition. Suppose that λ is a generalized eigenvalue of the pair $(\mathcal{M}_s, \mathcal{L}_s)$, with corresponding left eigenvector v, i.e.,

$$v\mathcal{M}_s = \lambda v\mathcal{L}_s.$$

Taking the Hermitian transpose of both sides yields

$$\mathcal{M}_s^H v^H = \bar{\lambda}\mathcal{L}_s^H v^H.$$

Multiplying both sides on the left by $\mathcal{M}_s\mathcal{J}$ and applying the symplectic relation gives

$$\begin{aligned}&\mathcal{M}_s\mathcal{J}\mathcal{M}_s^H v^H = \bar{\lambda}\mathcal{M}_s\mathcal{J}\mathcal{L}_s^H v^H \\ \Rightarrow &\mathcal{L}_s\mathcal{J}\mathcal{L}_s^H v^H = \bar{\lambda}\mathcal{M}_s\mathcal{J}\mathcal{L}_s^H v^H \qquad \text{(eq. (2.25))} \\ \Rightarrow &\frac{1}{\bar{\lambda}}\mathcal{L}_s\mathcal{J}\mathcal{L}_s^H v^H = \mathcal{M}_s\mathcal{J}\mathcal{L}_s^H v^H.\end{aligned}$$

Letting $u = \mathcal{J}\mathcal{L}_s^H v^H$, we then obtain $\mathcal{M}_s u = \frac{1}{\bar{\lambda}}\mathcal{L}_s u$, which shows that $\frac{1}{\bar{\lambda}}$ is also a generalized eigenvalue of the pair $(\mathcal{M}_s, \mathcal{L}_s)$.

□

In the previous discussion, we showed that the Möbius transformation maps a Hamiltonian matrix to a symplectic matrix while preserving the associated eigenvectors. We now present a theorem that generalizes this result from the matrix case to symplectic matrix pairs, demonstrating that the transformation retains similar structural behavior in the pair setting.

Theorem 2.6 ([9]). *The following statements hold:*

- *The matrix $\tilde{\mathcal{H}}$ is Hamiltonian if and only if the pair $(\mathcal{M}, \mathcal{L}) = \mathcal{C}(\tilde{\mathcal{H}})$ is symplectic.*
- *Suppose that $\tilde{\mathcal{H}}V = V\Lambda$, where $V \in \mathbb{R}^{2n\times m}$ and $\Lambda \in \mathbb{R}^{m\times m}$. Then,*

$$\mathcal{M}V = \mathcal{L}V\,C(\Lambda),$$

where $C(\Lambda)$ denotes the Möbius transformation of the matrix Λ, defined by the equation (2.20).

Proof. We first establish the equivalence between the Hamiltonian property of $\tilde{\mathcal{H}}$ and the symplecticity of the matrix pair $(\mathcal{M}, \mathcal{L}) = \mathcal{C}(\tilde{\mathcal{H}})$. Using equation (2.23), we consider the transformation

$$\mathcal{M} = \tilde{\mathcal{H}} + \gamma I, \quad \mathcal{L} = \tilde{\mathcal{H}} - \gamma I,$$

and verify whether

$$(\tilde{\mathcal{H}} + \gamma I)\mathcal{J}(\tilde{\mathcal{H}} + \gamma I)^H = (\tilde{\mathcal{H}} - \gamma I)\mathcal{J}(\tilde{\mathcal{H}} - \gamma I)^H.$$

Subtracting the right-hand side from the left, we compute:

$$\begin{aligned}&(\tilde{\mathcal{H}} + \gamma I)\mathcal{J}(\tilde{\mathcal{H}} + \gamma I)^H - (\tilde{\mathcal{H}} - \gamma I)\mathcal{J}(\tilde{\mathcal{H}} - \gamma I)^H \\ =&(\tilde{\mathcal{H}}\mathcal{J} + \gamma\mathcal{J})(\tilde{\mathcal{H}}^H + \gamma I) - (\tilde{\mathcal{H}}\mathcal{J} - \gamma\mathcal{J})(\tilde{\mathcal{H}}^H - \gamma I) \\ =&2\gamma(\tilde{\mathcal{H}}\mathcal{J} + \mathcal{J}\tilde{\mathcal{H}}^H).\end{aligned}$$

Since $\tilde{\mathcal{H}}$ is Hamiltonian, it satisfies the identity given in equation (2.16), so the difference vanishes and the identity holds. Hence, we conclude that $\mathcal{C}(\tilde{\mathcal{H}})$ is symplectic. The converse direction follows by reversing the implication.

For the second statement, assume $\tilde{\mathcal{H}}V = V\Lambda$. Then,

$$\mathcal{M}V = (\tilde{\mathcal{H}} + \gamma I)V = V\Lambda + \gamma V = V(\Lambda + \gamma I),$$
$$\mathcal{L}V = (\tilde{\mathcal{H}} - \gamma I)V = V\Lambda - \gamma V = V(\Lambda - \gamma I).$$

Hence,

$$\mathcal{M}V = \mathcal{L}V\,(\Lambda - \gamma I)^{-1}(\Lambda + \gamma I) = \mathcal{L}V\,C(\Lambda),$$

where $C(\Lambda)$ is given by the equation (2.20), which completes the proof. □

We now introduce the standard symplectic forms. In the following discussion, we frequently transform a matrix into one of these standard forms. There are two types of standard symplectic forms; although we primarily use the first one in this work, we present both for completeness.

Definition 2.6 ([8]). *A symplectic matrix pair $(\mathcal{M}, \mathcal{L})$ is said to be in one of the following standard symplectic forms:*

- ***First standard symplectic form (SSF-1)**:*

$$\mathcal{M} = \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} I & G \\ 0 & A^H \end{bmatrix},$$

  where $A, G, H \in \mathbb{C}^{n\times n}$, and both G and H are Hermitian matrices.

- ***Second standard symplectic form (SSF-2)**:*

$$\mathcal{M} = \begin{bmatrix} A & 0 \\ Q & -I \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} -P & I \\ A^H & 0 \end{bmatrix},$$

  where $A, P, Q \in \mathbb{C}^{n\times n}$, and both P and Q are Hermitian matrices.

By the definition of the standard symplectic form, we can formulate a theorem stating that certain matrix pairs preserve the symplectic structure. This structural preservation is essential for ensuring the stability of the system. Moreover, the converse of the theorem corresponds precisely to the definition of the standard symplectic form. In the following discussion, we will also use this converse to verify whether a given matrix pair is in the standard symplectic form.

Theorem 2.7. *The following statements hold:*

- *Suppose*

$$\mathcal{M} = \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} I & G \\ 0 & B \end{bmatrix},$$

  where $A, B, G, H \in \mathbb{C}^{n\times n}$. If $(\mathcal{M}, \mathcal{L})$ is a symplectic pair, then

$$B = A^H, \quad G = G^H, \quad H = H^H.$$

- *Suppose*

$$\mathcal{M} = \begin{bmatrix} A & 0 \\ Q & -I \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} -P & I \\ B & 0 \end{bmatrix},$$

  where $A, B, P, Q \in \mathbb{C}^{n\times n}$. If $(\mathcal{M}, \mathcal{L})$ is a symplectic pair, then

$$B = A^H, \quad Q = Q^H, \quad P = P^H.$$

Proof. We prove the statements by direct computation.

- Suppose that $(\mathcal{M}, \mathcal{L})$ is a symplectic pair. By definition, we have the equation (2.25). Compute each side separately:

$$\begin{aligned}
\mathcal{M}\mathcal{J}\mathcal{M}^H &= \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix}^H \\
&= \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} A^H & -H^H \\ 0 & I \end{bmatrix} \\
&= \begin{bmatrix} 0 & A \\ -A^H & H^H - H \end{bmatrix},
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{L}\mathcal{J}\mathcal{L}^H &= \begin{bmatrix} I & G \\ 0 & B \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} I & G \\ 0 & B \end{bmatrix}^H \\
&= \begin{bmatrix} I & G \\ 0 & B \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ G^H & B^H \end{bmatrix} \\
&= \begin{bmatrix} -G + G^H & B^H \\ -B & 0 \end{bmatrix}.
\end{aligned}$$

Equating both sides, we obtain the identities:

$$-G + G^H = 0, \quad A = B^H, \quad H^H - H = 0,$$

which imply that

$$B = A^H, \quad G = G^H, \quad H = H^H.$$

- Suppose that $(\mathcal{M}, \mathcal{L})$ is a symplectic pair. By definition, we have the equation (2.25). Compute each side separately:

$$\begin{aligned}
\mathcal{M}\mathcal{J}\mathcal{M}^H &= \begin{bmatrix} A & 0 \\ Q & -I \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} A & 0 \\ Q & -I \end{bmatrix}^H \\
&= \begin{bmatrix} A & 0 \\ Q & -I \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} A^H & Q^H \\ 0 & -I \end{bmatrix} \\
&= \begin{bmatrix} 0 & -A \\ A^H & Q^H - Q \end{bmatrix},
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{L}\mathcal{J}\mathcal{L}^H &= \begin{bmatrix} -P & I \\ B & 0 \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} -P & I \\ B & 0 \end{bmatrix}^H \\
&= \begin{bmatrix} -P & I \\ B & 0 \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} -P^H & B^H \\ I & 0 \end{bmatrix} \\
&= \begin{bmatrix} P^H - P & -B^H \\ B & 0 \end{bmatrix}.
\end{aligned}$$

Equating both sides, we obtain the identities:

$$P^H - P = 0, \quad A = B^H, \quad Q^H - Q = 0,$$

which imply that

$$B = A^H, \quad P = P^H, \quad Q = Q^H.$$

□

Before presenting the next theorem, we introduce the concept of left equivalence. This definition is crucial, as it is frequently used throughout this thesis to transform a given matrix into a desired form.

Definition 2.7 (Left Equivalence)**.** *Let T be a nonsingular matrix, and let (M, L) be a matrix pair. The matrix pair (TM, TL), obtained by left-multiplying both matrices by T, is said to be* left-equivalent *to* (M, L).

The following theorem is the most important result in this subsection. It describes the process of transforming a Hamiltonian matrix into a symplectic pair via the Möbius transformation. We then further transform the resulting symplectic pair into the standard symplectic form. This proof is particularly significant, as it verifies key conditions needed to preserve structural stability. Moreover, the same procedure frequently appears in subsequent transformations to the standard symplectic form.

Theorem 2.8 ([3]). *Let the Hamiltonian matrix be*

$$\tilde{\mathcal{H}} = \begin{bmatrix} A & G \\ H & -A^H \end{bmatrix},$$

where $A, G, H \in \mathbb{C}^{n\times n}$, with $G = G^H$, $H = H^H$. Define the Möbius transformation for matrix pair by

$$\mathcal{M} = \begin{bmatrix} A+\gamma I & G \\ H & -A^H+\gamma I \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} A-\gamma I & G \\ H & -A^H-\gamma I \end{bmatrix},$$

for some scalar $\gamma \in \mathbb{R}$ with $\gamma > 0$. Then, the pair $(\mathcal{M}, \mathcal{L})$ is left-equivalent to the first standard symplectic form:

$$\mathcal{M}' = \begin{bmatrix} A' & 0 \\ -H' & I \end{bmatrix}, \quad \mathcal{L}' = \begin{bmatrix} I & G' \\ 0 & A'^H \end{bmatrix}, \tag{2.26}$$

where the matrices $A', G', H' \in \mathbb{C}^{n\times n}$ are given by

$$\begin{aligned} A' &:= I + 2\gamma W_\gamma^{-H}, \\ G' &:= -2\gamma W_\gamma^{-H} G A_\gamma^{-H}, \\ H' &:= -2\gamma A_\gamma^{-H} H W_\gamma^{-H}, \end{aligned} \tag{2.27}$$

with $A_\gamma := A - \gamma I$, and $W_\gamma := A_\gamma^H + H A_\gamma^{-1} G$. These matrices satisfy the structural conditions $G' = G'^H$, $H' = H'^H$, and the $(1,1)$-block of $\mathcal{M}'$ equals the Hermitian transpose of the $(2,2)$-block of $\mathcal{L}'$, i.e.,

$$(A')^H = A'^H.$$

Proof. Let $A_\gamma = A - \gamma I$, $W_\gamma = A_\gamma^H + H A_\gamma^{-1} G$

$$\begin{aligned}
&\left[\begin{array}{cc|cc} A+\gamma I & G & A-\gamma I & G \\ H & -A^H+\gamma I & H & -A^H-\gamma I \end{array}\right] \\
=&\left[\begin{array}{cc|cc} A_\gamma+2\gamma I & G & A_\gamma & G \\ H & -A_\gamma^H & H & -A_\gamma^H-2\gamma I \end{array}\right] \\
\xrightarrow{T^{(1)}}&\left[\begin{array}{cc|cc} A_\gamma+2\gamma I & G & A_\gamma & G \\ -A_\gamma^{-H}H & I & -A_\gamma^{-H}H & I+2\gamma A_\gamma^{-H} \end{array}\right] \\
\xrightarrow{T^{(2)}}&\left[\begin{array}{cc|cc} A_\gamma+2\gamma I+GA_\gamma^{-H}H & 0 & A_\gamma+GA_\gamma^{-H}H & -2\gamma GA_\gamma^{-H} \\ -A_\gamma^{-H}H & I & -A_\gamma^{-H}H & I+2\gamma A_\gamma^{-H} \end{array}\right] \\
=&\left[\begin{array}{cc|cc} W_\gamma^H+2\gamma I & 0 & W_\gamma^H & -2\gamma GA_\gamma^{-H} \\ -A_\gamma^{-H}H & I & -A_\gamma^{-H}H & I+2\gamma A_\gamma^{-H} \end{array}\right] \\
\xrightarrow{T^{(3)}}&\left[\begin{array}{cc|cc} I+2\gamma W_\gamma^{-H} & 0 & I & -2\gamma W_\gamma^{-H}GA_\gamma^{-H} \\ -A_\gamma^{-H}H & I & -A_\gamma^{-H}H & I+2\gamma A_\gamma^{-H} \end{array}\right] \\
\xrightarrow{T^{(4)}}&\left[\begin{array}{cc|cc} I+2\gamma W_\gamma^{-H} & 0 & I & -2\gamma W_\gamma^{-H}GA_\gamma^{-H} \\ 2\gamma A_\gamma^{-H}HW_\gamma^{-H} & I & 0 & I+2\gamma A_\gamma^{-H}-2\gamma A_\gamma^{-H}HW_\gamma^{-H}GA_\gamma^{-H} \end{array}\right]
\end{aligned}$$

where the transformation matrices are given by

$$T^{(1)} = \begin{bmatrix} I & 0 \\ 0 & -A_\gamma^{-H} \end{bmatrix}, \quad T^{(2)} = \begin{bmatrix} I & -G \\ 0 & I \end{bmatrix}, \quad T^{(3)} = \begin{bmatrix} W_\gamma^{-H} & 0 \\ 0 & I \end{bmatrix}, \quad T^{(4)} = \begin{bmatrix} I & 0 \\ A_\gamma^{-H}H & I \end{bmatrix}.$$

By applying the Möbius transformation and reducing to the standard symplectic form, we obtain a result that, when compared with equation (2.26), yields equation (2.27).

We now show that $H' := -2\gamma A_\gamma^{-H} H W_\gamma^{-H}$ is Hermitian. Observe that

$$\left(W_\gamma^H H^{-1} A_\gamma^H\right)^{-1} = A_\gamma^{-H} H W_\gamma^{-H} = W_\gamma^{-1} H A_\gamma^{-1},$$

which holds if and only if

$$W_\gamma^H H^{-1} A_\gamma^H = A_\gamma H^{-1} W_\gamma.$$

We now verify this identity. Since $W_\gamma = A_\gamma^H + HA_\gamma^{-1}G$, we compute:

$$\begin{aligned} W_\gamma^H H^{-1} A_\gamma^H &= (A_\gamma + GA_\gamma^{-H}H)H^{-1}A_\gamma^H = A_\gamma H^{-1} A_\gamma^H + G, \\ A_\gamma H^{-1} W_\gamma &= A_\gamma H^{-1}(A_\gamma^H + HA_\gamma^{-1}G) = A_\gamma H^{-1} A_\gamma^H + G. \end{aligned}$$

In both cases, we obtain the same result $A_\gamma H^{-1} A_\gamma^H + G$. Hence, we conclude that

$$A_\gamma^{-H} H W_\gamma^{-H} = \left(A_\gamma^{-H} H W_\gamma^{-H}\right)^H,$$

which implies that H' is Hermitian.

Next, we show that $G' := -2\gamma W_\gamma^{-H} G A_\gamma^{-H}$ is Hermitian. To prove this, we aim to show that

$$W_\gamma^{-H} G A_\gamma^{-H} = \left(W_\gamma^{-H} G A_\gamma^{-H}\right)^H = A_\gamma^{-1} G W_\gamma^{-1}.$$

This equality can be derived by taking the matrix inverse on both sides, which leads to the condition

$$A_\gamma^H G^{-1} W_\gamma^H = W_\gamma G^{-1} A_\gamma.$$

We now verify this identity. Using the definition $W_\gamma = A_\gamma^H + HA_\gamma^{-1}G$, we expand both sides:

$$\begin{aligned} A_\gamma^H G^{-1} W_\gamma^H &= A_\gamma^H G^{-1}\left(A_\gamma + GA_\gamma^{-H}H\right) = A_\gamma^H G^{-1} A_\gamma + H, \\ W_\gamma G^{-1} A_\gamma &= \left(A_\gamma^H + HA_\gamma^{-1}G\right) G^{-1} A_\gamma = A_\gamma^H G^{-1} A_\gamma + H. \end{aligned}$$

We find that both sides simplify to $A_\gamma^H G^{-1} A_\gamma + H$. Thus, the identity holds, and we conclude that

$$W_\gamma^{-H} G A_\gamma^{-H} = \left(W_\gamma^{-H} G A_\gamma^{-H}\right)^H,$$

which proves that G' is Hermitian.

Finally, we verify the Hermitian relation between the $(1,1)$-block of $\mathcal{M}'$ and the $(2,2)$-block of $\mathcal{L}'$, that is,

$$(A')^H = A'^H.$$

Here, the matrix $A' \in \mathbb{C}^{n\times n}$ is defined by

$$A' := I + 2\gamma W_\gamma^{-H},$$

which arises as the upper-left block of $\mathcal{M}'$ after transformation. On the other hand, the $(2,2)$-block of $\mathcal{L}'$ is given explicitly as

$$A'^H = I + 2\gamma A_\gamma^{-H} - 2\gamma A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}.$$

We now show that these two expressions are indeed Hermitian conjugates, we compute the Hermitian transpose of A':

$$(A')^H = (I + 2\gamma W_\gamma^{-H})^H = I + 2\gamma W_\gamma^{-1}.$$

Hence, it suffices to prove that

$$I + 2\gamma W_\gamma^{-1} = I + 2\gamma A_\gamma^{-H} - 2\gamma A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H},$$

which is equivalent to

$$W_\gamma^{-1} = A_\gamma^{-H} - A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}.$$

We verify this identity by left-multiplying both sides by W_γ:

$$\begin{aligned} I =& W_\gamma\left(A_\gamma^{-H} - A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}\right) \\ =& (A_\gamma^H + HA_\gamma^{-1}G)(A_\gamma^{-H} - A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}) \\ =& I + HA_\gamma^{-1} G A_\gamma^{-H} - H W_\gamma^{-H} G A_\gamma^{-H} - HA_\gamma^{-1} G A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}. \end{aligned}$$

Thus, to conclude the identity, it suffices to show that

$$HA_\gamma^{-1} G A_\gamma^{-H} = H W_\gamma^{-H} G A_\gamma^{-H} + HA_\gamma^{-1} G A_\gamma^{-H} H W_\gamma^{-H} G A_\gamma^{-H}.$$

To verify this, we use the identity

$$A_\gamma^{-1} = W_\gamma^{-H} + A_\gamma^{-1} G A_\gamma^{-H} H W_\gamma^{-H},$$

which can be rewritten as

$$\begin{aligned} A_\gamma^{-1} &= (I + A_\gamma^{-1} G A_\gamma^{-H} H) W_\gamma^{-H} \\ &= (A_\gamma^{-1} A_\gamma + A_\gamma^{-1} G A_\gamma^{-H} H) W_\gamma^{-H} \\ &= A_\gamma^{-1} (A_\gamma + G A_\gamma^{-H} H) W_\gamma^{-H}. \end{aligned}$$

Since $W_\gamma = A_\gamma^H + H A_\gamma^{-1} G$, we have $W_\gamma^H = A_\gamma + G A_\gamma^{-H} H$, and thus

$$A_\gamma^{-1} = A_\gamma^{-1} W_\gamma^H W_\gamma^{-H}.$$

Therefore, the identity holds. This confirms the Hermitian relation between the $(1,1)$-block of $\mathcal{M}'$ and the $(2,2)$-block of $\mathcal{L}'$, that is,

$$(A')^H = A'^H.$$

□

This transformation ensures that the resulting pair $(\mathcal{M}', \mathcal{L}')$ retains the symplectic structure, with blockwise Hermitian properties:

$$G' = G'^H, \quad H' = H'^H, \quad (A')^H = A'^H.$$

Such structure preservation is crucial for maintaining numerical stability in the following computations, especially in structure-preserving algorithms such as the doubling algorithm or other symplectic iterations. By working within this structured form, we avoid numerical artifacts that could arise from symmetry loss, and ensure better convergence behavior in iteration methods.

2.3 SDA: Numerical Methods for Solving DAREs Efficiently

In this subsection, we aim to explore how the structure-preserving doubling algorithm can be derived and applied efficiently. The key idea is based on the spectral property that if an eigenvalue lies strictly within the unit circle, repeated multiplication will drive it toward zero. This decay property plays an important role in analyzing the stability and structure of the system. To achieve this, we begin by considering the square of a matrix pair. Through this construction, we ensure that eigenvalues within the unit circle decay to zero under iteration. By further analyzing the resulting matrix form, we derive the structure-preserving doubling algorithm, which enables efficient computation while preserving the system's symplectic structure.

First, we introduce the definition of the square of a matrix pair. This concept allows eigenvalues whose absolute values are less than one to converge quickly to zero, which is important for efficient computation.

Definition 2.8. *We define a matrix pair $(\widehat{\mathcal{M}}, \widehat{\mathcal{L}})$ to be a square of $(\mathcal{M}, \mathcal{L})$ if, whenever an eigenpair (Y, Λ) satisfies*

$$\mathcal{M}Y = \mathcal{L}Y\Lambda \quad \textit{(equivalently, } \mathcal{M}Y\Lambda = \mathcal{L}Y\textit{)},$$

then it follows that

$$\widehat{\mathcal{M}}Y = \widehat{\mathcal{L}}Y\Lambda^2 \quad \textit{(equivalently, } \widehat{\mathcal{M}}Y\Lambda^2 = \widehat{\mathcal{L}}Y\textit{)}.$$

We begin with a matrix pair that is already a square of another pair. Before transforming it into the first standard symplectic form, we note that left multiplication by a nonsingular matrix preserves the square structure. This is stated in the following remark.

Remark. *Let L be a nonsingular matrix. If the pair $(\widehat{\mathcal{M}}, \widehat{\mathcal{L}})$ is a square of $(\mathcal{M}, \mathcal{L})$, then the transformed pair $(L\widehat{\mathcal{M}}, L\widehat{\mathcal{L}})$ is also a square of $(\mathcal{M}, \mathcal{L})$.*

In the next theorem, we introduce the concept of the square of a matrix pair. This theorem requires that both $\mathcal{M}$ and $\mathcal{L}$ are nonsingular. However, we know that not all matrices have this property. Therefore, after presenting the theorem, we will discuss how to construct appropriate matrices $\mathcal{M}_*$ and $\mathcal{L}_*$ to overcome this limitation.

Theorem 2.9 ([8]). *The matrix pair $(\widehat{\mathcal{M}},\ \widehat{\mathcal{L}})$, given by*

$$\widehat{\mathcal{M}} = \mathcal{M}_*\mathcal{M}, \qquad \widehat{\mathcal{L}} = \mathcal{L}_*\mathcal{L}, \tag{2.28}$$

constitutes a square of the pair $(\mathcal{M},\ \mathcal{L})$ via the matrices $\mathcal{M}_*$ and $\mathcal{L}_*$, provided that $\mathcal{M}$ and $\mathcal{L}$ are both nonsingular, and

$$\mathcal{M}_* := C\mathcal{L}^{-1}, \qquad \mathcal{L}_* := C\mathcal{M}^{-1}, \tag{2.29}$$

for some invertible matrix C.

Proof. Suppose that $(\mathcal{M},\ \mathcal{L})$ is a matrix pair, and that both $\mathcal{M}$ and $\mathcal{L}$ are nonsingular. Let (Y, Λ) be a generalized eigenpair of $(\mathcal{M},\ \mathcal{L})$. Starting from the defining equation, we obtain the following sequence of equivalent expressions:

$$\begin{aligned} \mathcal{M}Y = \mathcal{L}Y\Lambda &\Rightarrow (\mathcal{L}^{-1}\mathcal{M})Y = Y\Lambda \\ &\Rightarrow (\mathcal{L}^{-1}\mathcal{M})^2 Y = Y\Lambda^2 \\ &\Rightarrow (\mathcal{L}^{-1}\mathcal{M})Y = (\mathcal{M}^{-1}\mathcal{L})Y\Lambda^2. \end{aligned}$$

Now define

$$\widehat{\mathcal{M}} := C(\mathcal{L}^{-1}\mathcal{M}), \qquad \widehat{\mathcal{L}} := C(\mathcal{M}^{-1}\mathcal{L}),$$

which can be rewritten as

$$\widehat{\mathcal{M}} = \mathcal{M}_*\mathcal{M}, \qquad \widehat{\mathcal{L}} = \mathcal{L}_*\mathcal{L},$$

where $\mathcal{M}_* := C\mathcal{L}^{-1}$ and $\mathcal{L}_* := C\mathcal{M}^{-1}$, with C being any invertible matrix. Substituting the identity above, we obtain

$$\widehat{\mathcal{M}}Y = C(\mathcal{L}^{-1}\mathcal{M})Y = C(\mathcal{M}^{-1}\mathcal{L})Y\Lambda^2 = \widehat{\mathcal{L}}Y\Lambda^2.$$

Therefore, by definition, the matrix pair $(\widehat{\mathcal{M}}, \widehat{\mathcal{L}})$ is a square of $(\mathcal{M}, \mathcal{L})$. □

Now, we consider the case where either $\mathcal{M}$ or $\mathcal{L}$ is singular. In general, either matrix in the pair may be singular, so we need to find an alternative approach to handle this case. From equation (2.29), we observe the following identity:

$$\mathcal{M}_*\mathcal{L} = \mathcal{L}_*\mathcal{M} \quad \Longleftrightarrow \quad \mathcal{M}_*\mathcal{L} - \mathcal{L}_*\mathcal{M} = 0 \quad \Longleftrightarrow \quad \begin{bmatrix} \mathcal{M}_* & \mathcal{L}_* \end{bmatrix} \begin{bmatrix} \mathcal{L} \\ -\mathcal{M} \end{bmatrix} = 0.$$

This means that the block row vector $\begin{bmatrix} \mathcal{M}_* & \mathcal{L}_* \end{bmatrix}$ lies in the left null space of the block column matrix $\begin{bmatrix} \mathcal{L} \\ -\mathcal{M} \end{bmatrix}$. Therefore, when either $\mathcal{M}$ or $\mathcal{L}$ is singular, we may still determine appropriate matrices $\mathcal{M}_*$ and $\mathcal{L}_*$ by solving the associated homogeneous linear system.

In the next theorem, we encounter a situation in which the computation can be significantly simplified by applying a well-known identity. To facilitate this, we introduce the Sherman–Morrison–Woodbury formula—a classical result in linear algebra that provides an efficient way to compute the inverse of a matrix subject to a low-rank update. This identity, derived from basic linear algebra techniques, will play a crucial role in improving the efficiency of later computations.

Theorem 2.10 (Sherman–Morrison–Woodbury Identity: Special Case). *Let $U, V \in \mathbb{C}^{n\times n}$, and let $I \in \mathbb{C}^{n\times n}$ denote the identity matrix. Assume that $I + VU$ is invertible. Then the following identity holds:*

$$(I + UV)^{-1} = I - U(I + VU)^{-1}V. \tag{2.30}$$

Proof. We verify the identity by direct computation:

$$\begin{aligned} &(I + UV)(I - U(I + VU)^{-1}V) \\ =&I - U(I + VU)^{-1}V + UV - UVU(I + VU)^{-1}V \\ =&I - U(I + VU)^{-1}V + U(I - VU(I + VU)^{-1})V \\ =&I - U(I + VU)^{-1}V + U((I + VU)(I + VU)^{-1} - VU(I + VU)^{-1})V \\ =&I - U(I + VU)^{-1}V + U(I + VU)^{-1}V \\ =&I. \end{aligned}$$

Therefore, $(I + UV)(I - U(I + VU)^{-1}V) = I$, which shows that

$$(I + UV)^{-1} = I - U(I + VU)^{-1}V.$$

□

We now state the following theorem and again assume that both $\mathcal{M}$ and $\mathcal{L}$ are nonsingular. We apply a left multiplication by the matrix

$$\left[\mathcal{L}^{-1}|\mathcal{M}^{-1}\right] \quad \longrightarrow \quad \left[\mathcal{M}_*|\mathcal{L}_*\right],$$

so that the resulting matrix pair is transformed into the first standard symplectic form. Then, by using equation (2.28), we obtain a square of the original matrix pair. Finally, we verify whether this square form preserves the symplectic structure.

Theorem 2.11 ([8]). *Based on equation* (2.26)*, the matrix pair* $(\widehat{\mathcal{M}}, \widehat{\mathcal{L}})$ *is given by:*

$$\widehat{\mathcal{M}} = \begin{bmatrix} \widehat{A} & 0 \\ -\widehat{H} & I \end{bmatrix}, \qquad \widehat{\mathcal{L}} = \begin{bmatrix} I & \widehat{G} \\ 0 & \widehat{A}^H \end{bmatrix},$$

where the matrices $\widehat{A}, \widehat{G}, \widehat{H} \in \mathbb{C}^{n\times n}$ *are given by*

$$\begin{aligned} \widehat{A} &= A(I+GH)^{-1}A, \\ \widehat{G} &= G + AG(I+HG)^{-1}A^H, \\ \widehat{H} &= H + A^H(I+HG)^{-1}HA. \end{aligned} \tag{2.31}$$

Moreover, both $\widehat{H}$ *and* $\widehat{G}$ *are Hermitian matrices, and the upper-left block of* $\widehat{\mathcal{M}}$ *is the Hermitian transpose of the lower-right block of* $\widehat{\mathcal{L}}$*, i.e.,*

$$(\widehat{\mathcal{M}})_{11} = \widehat{A}, \quad (\widehat{\mathcal{L}})_{22} = \widehat{A}^H.$$

Proof. We begin with the matrix pair given in equation (2.26), and factor each matrix as follows:

$$\mathcal{M} = \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} I & 0 \\ -H & I \end{bmatrix} \quad \Rightarrow \quad \mathcal{M}^{-1} = \begin{bmatrix} I & 0 \\ H & I \end{bmatrix}\begin{bmatrix} A^{-1} & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} A^{-1} & 0 \\ HA^{-1} & I \end{bmatrix},$$

$$\mathcal{L} = \begin{bmatrix} I & 0 \\ 0 & A^H \end{bmatrix}\begin{bmatrix} I & G \\ 0 & I \end{bmatrix} \quad \Rightarrow \quad \mathcal{L}^{-1} = \begin{bmatrix} I & -G \\ 0 & I \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & A^{-H} \end{bmatrix} = \begin{bmatrix} I & -GA^{-H} \\ 0 & A^{-H} \end{bmatrix}.$$

We now compute the block matrix $\left[\mathcal{L}^{-1}|\mathcal{M}^{-1}\right]$ and apply a sequence of elementary transformations.

$$\begin{aligned} \left[\mathcal{L}^{-1}|\mathcal{M}^{-1}\right] &= \left[\begin{array}{cc|cc} I & -GA^{-H} & A^{-1} & 0 \\ 0 & A^{-H} & HA^{-1} & I \end{array}\right] \\ &\xrightarrow{T^{(1)}} \left[\begin{array}{cc|cc} I & -GA^{-H} & A^{-1} & 0 \\ 0 & I & A^HHA^{-1} & A^H \end{array}\right] \\ &\xrightarrow{T^{(2)}} \left[\begin{array}{cc|cc} I & 0 & (I+GH)A^{-1} & G \\ 0 & I & A^HHA^{-1} & A^H \end{array}\right] \\ &\xrightarrow{T^{(3)}} \left[\begin{array}{cc|cc} A(I+GH)^{-1} & 0 & I & A(I+GH)^{-1}G \\ 0 & I & A^HHA^{-1} & A^H \end{array}\right] \\ &\xrightarrow{T^{(4)}} \left[\begin{array}{cc|cc} A(I+GH)^{-1} & 0 & I & A(I+GH)^{-1}G \\ -A^HH(I+GH)^{-1} & I & 0 & A^H(I-H(I+GH)^{-1}G) \end{array}\right] \\ &= \left[\mathcal{M}_*|\mathcal{L}_*\right] \end{aligned}$$

where the transformation matrices are given by

$$T^{(1)} = \begin{bmatrix} I & 0 \\ 0 & A^H \end{bmatrix}, \quad T^{(2)} = \begin{bmatrix} I & GA^{-H} \\ 0 & I \end{bmatrix}, \quad T^{(3)} = \begin{bmatrix} A(I+GH)^{-1} & 0 \\ 0 & I \end{bmatrix}, \quad T^{(4)} = \begin{bmatrix} I & 0 \\ -A^HHA^{-1} & I \end{bmatrix}.$$

We then proceed to compute $\widehat{\mathcal{M}}$ and $\widehat{\mathcal{L}}$.

$$\begin{aligned} \widehat{\mathcal{M}} = \mathcal{M}_*\mathcal{M} &= \begin{bmatrix} A(I+GH)^{-1} & 0 \\ -A^HH(I+GH)^{-1} & I \end{bmatrix}\begin{bmatrix} A & 0 \\ -H & I \end{bmatrix} \\ &= \begin{bmatrix} A(I+GH)^{-1}A & 0 \\ -A^HH(I+GH)^{-1}A - H & I \end{bmatrix} \\ \widehat{\mathcal{L}} = \mathcal{L}_*\mathcal{L} &= \begin{bmatrix} I & A(I+GH)^{-1}G \\ 0 & A^H(I-H(I+GH)^{-1}G) \end{bmatrix}\begin{bmatrix} I & G \\ 0 & A^H \end{bmatrix} \\ &= \begin{bmatrix} I & G + A(I+GH)^{-1}GA^H \\ 0 & A^H(I-H(I+GH)^{-1}G)A^H \end{bmatrix} \\ &= \begin{bmatrix} I & G + A(I+GH)^{-1}GA^H \\ 0 & A^H(I+HG)^{-1}A^H \end{bmatrix} \end{aligned}$$

The last equality follows from the identity

$$(I - H(I + GH)^{-1}G) = (I + HG)^{-1},$$

which can be derived using the matrix inversion lemma or verified directly. This identity is applied to simplify the structure-preserving transformation given in equation (2.30) of Theorem 2.10. From the final expressions, we identify the result stated in equation (2.31).

Now, we aim to prove that the matrices $A(I + GH)^{-1}A$ and $A^H(I + HG)^{-1}A^H$ are Hermitian, and that the matrices $-A^H H(I + GH)^{-1}A - H$ and $G + A(I + GH)^{-1}GA^H$ are also Hermitian.

We begin by showing that $\big(A(I + GH)^{-1}A\big)^H = A^H(I + HG)^{-1}A^H$. Taking the Hermitian transpose, we obtain

$$\big(A(I + GH)^{-1}A\big)^H = A^H(I + GH)^{-H}A^H = A^H(I + HG)^{-1}A^H.$$

The second equality holds because $(I+GH)^H = I+H^H G^H = I+HG$, assuming that $G = G^H$ and $H = H^H$.

Next, we aim to prove that the matrix $-A^H H(I+GH)^{-1}A-H$ is Hermitian. Taking the Hermitian transpose, we compute:

$$\big(-A^H H(I + GH)^{-1}A - H\big)^H = -A^H(I + GH)^{-H}H^H A - H^H = -A^H(I + HG)^{-1}HA - H.$$

This follows from the assumptions $G = G^H$ and $H = H^H$, which imply $(I + GH)^H = I + HG$, and thus $(I + GH)^{-H} = (I + HG)^{-1}$. Since the Hermitian transpose of the matrix equals itself, the matrix is Hermitian. To justify the identity $H(I + GH)^{-1} = (I + HG)^{-1}H$, we verify that:

$$\begin{aligned} H(I + GH)^{-1} = (I + HG)^{-1}H &\Leftrightarrow (I + HG)H = H(I + GH) \\ &\Leftrightarrow (I + HG)H = H + HGH = H(I + GH), \end{aligned}$$

which confirms the desired equality.

Finally, we aim to prove that the matrix $G+A(I+GH)^{-1}GA^H$ is Hermitian. Taking the Hermitian transpose, we compute:

$$\big(G + A(I + GH)^{-1}GA^H\big)^H = G^H + AG^H(I + GH)^{-H}A^H = G + AG(I + HG)^{-1}A^H.$$

This follows from the assumptions $G = G^H$ and $H = H^H$, which imply $(I + GH)^H = I + HG$, and thus $(I+GH)^{-H} = (I+HG)^{-1}$. Therefore, the Hermitian transpose of the matrix equals itself, which confirms that the matrix is Hermitian. To further justify the equality, we verify the identity $(I + GH)^{-1}G = G(I + HG)^{-1}$ as follows:

$$\begin{aligned} (I + GH)^{-1}G = G(I + HG)^{-1} &\Leftrightarrow G(I + HG) = (I + GH)G \\ &\Leftrightarrow G + GHG = G + GHG, \end{aligned}$$

which confirms the desired equality. This completes the proof of the theorem. □

By the recurrence relations in equation (2.31), the structure-preserving doubling algorithm (SDA) updates the matrices as follows:

$$\begin{aligned} A_{k+1} &= A_k(I + G_k H_k)^{-1}A_k, \\ G_{k+1} &= G_k + A_k G_k(I + H_k G_k)^{-1}A_k^H, \\ H_{k+1} &= H_k + A_k^H(I + H_k G_k)^{-1}H_k A_k. \end{aligned}$$

In the final part of this section, we illustrate how the structure-preserving doubling algorithm can be efficiently used to compute the stabilizing solution of the discrete algebraic Riccati equation (DARE). The following theorem provides a theoretical justification.

Theorem 2.12. *Let X^* be the stabilizing solution of the discrete-time algebraic Riccati equation. Then,*

$$\begin{aligned} \mathcal{H}\begin{bmatrix} I \\ X^* \end{bmatrix} &= \begin{bmatrix} I \\ X^* \end{bmatrix}\Lambda, \\ \mathcal{M}_0\begin{bmatrix} I \\ X^* \end{bmatrix} &= \mathcal{L}_0\begin{bmatrix} I \\ X^* \end{bmatrix}\Lambda', \\ \mathcal{M}_k\begin{bmatrix} I \\ X^* \end{bmatrix} &= \mathcal{L}_k\begin{bmatrix} I \\ X^* \end{bmatrix}(\Lambda')^{2^k}, \end{aligned}$$

where $\mathcal{H}$ is the Hamiltonian matrix and $(\mathcal{M}_k, \mathcal{L}_k)$ are the transformed matrix pairs at the k-th iteration.

Suppose the sequences $\{A_k\}$, $\{H_k\}$, and $\{G_k\}$ generated by the algorithm are convergent. Then:

- $A_k \to 0$ *as* $k \to \infty$*;*

- $H_k \to X^*$ *as* $k \to \infty$.

Proof. Suppose $A_k \to A^*$, $H_k \to H^*$, and $G_k \to G^*$ as $k \to \infty$. Consider the identity

$$\begin{bmatrix} A_k & 0 \\ -H_k & I \end{bmatrix} \begin{bmatrix} I \\ X^* \end{bmatrix} = \begin{bmatrix} I & G_k \\ 0 & A_k^H \end{bmatrix} \begin{bmatrix} I \\ X^* \end{bmatrix} (\Lambda')^{2^k}.$$

Taking the limit on both sides as $k \to \infty$, and using the assumption that $(\Lambda')^{2^k} \to 0$, we obtain:

$$\begin{bmatrix} A^* & 0 \\ -H^* & I \end{bmatrix} \begin{bmatrix} I \\ X^* \end{bmatrix} = \begin{bmatrix} I & G^* \\ 0 & (A^*)^H \end{bmatrix} \begin{bmatrix} I \\ X^* \end{bmatrix} \cdot 0 = 0.$$

Hence,

$$\begin{bmatrix} A^* & 0 \\ -H^* & I \end{bmatrix} \begin{bmatrix} I \\ X^* \end{bmatrix} = 0.$$

Expanding this equation gives:

$$\begin{aligned} A^* \cdot I + 0 \cdot X^* &= 0, \\ -H^* \cdot I + I \cdot X^* &= 0, \end{aligned}$$

which implies:

$$\begin{aligned} A^* &= 0, \\ H^* &= X^*. \end{aligned}$$

Therefore, we conclude that $A_k \to 0$ and $H_k \to X^*$, completing the proof. □

This result demonstrates that the Structure-Preserving Doubling Algorithm (SDA) achieves quadratic convergence toward the stabilizing solution X^*, while preserving the underlying structure of the problem. Under mild assumptions, the method guarantees both efficiency and numerical stability.

In this section, we have introduced the complete structure of the *Structure-Preserving Doubling Algorithm (SDA)*. Starting from the Linear Quadratic Regulator (LQR) problem, we derived the Algebraic Riccati Equation (ARE), which can be rewritten in matrix form to obtain the associated Hamiltonian matrix. Applying a Möbius transformation, the Hamiltonian matrix is converted into a symplectic matrix pair, which is then further reduced to the first standard symplectic form (SSF-1). By iteratively squaring the matrix pair in this form, the algorithm generates a sequence of matrix pairs, and it can be shown that the resulting solution corresponds to the stabilizing solution of the original ARE. This entire procedure constitutes the core of the doubling process.

In the following chapters, we shift our attention to a special class of matrices—*irreducible and aperiodic stochastic matrices*—whose dominant eigenvalue lies on the unit circle. These matrices serve as an important test case for evaluating the behavior of the doubling algorithm. We will investigate how the doubling process behaves when applied to such matrices and observe how convergence manifests in this context.

To further support the analysis, we introduce the *power method* as a conceptual benchmark. It is not only used for numerical comparison with the doubling algorithm (DA), but its iterative structure also provides valuable insights into the convergence behavior of DA. In particular, the conceptual parallels between the power method and DA help motivate certain interpretations and guide the theoretical understanding developed later in this thesis.

In the subsequent sections, we will first review the spectral properties of irreducible and aperiodic stochastic matrices, followed by an introduction to the concept of the power method as a foundational iterative approach. We then present a detailed examination of how these matrices behave under the doubling process. This framework ultimately allows us to extend the doubling algorithm to general matrices.

3 Stochastic matrices and Pageranks

In this section, we examine some properties of stochastic matrices. If a stochastic matrix is irreducible and aperiodic, then its eigenspace corresponding to the eigenvalue 1 is one-dimensional, and all other eigenvalues have absolute values strictly less than 1. In fact, there is exactly one eigenvalue on the unit circle, while all other eigenvalues lie strictly inside the unit circle. This property is very similar to how the Möbius transformation turns a Hamiltonian matrix into a symplectic matrix with all eigenvalues inside the unit disk. Inspired by this, we apply a similar idea using the Doubling Algorithm (DA) to explore its effectiveness for stochastic matrices.

Since the Doubling Algorithm (DA) aims to compute the dominant eigenspace, its convergence behavior—especially with respect to the dominant eigenvector—is of particular interest. To better understand and validate its effectiveness, we compare it with the classical power method, which also targets the dominant eigenvector through a simple iterative scheme.

Given that all eigenvalues of an irreducible and aperiodic stochastic matrix, except for the leading eigenvalue 1, lie strictly inside the unit circle, repeated matrix multiplication in the power method leads to convergence toward the corresponding eigenspace. This spectral property aligns closely with the convergence mechanism of the doubling algorithm.

Therefore, we conduct numerical experiments on irreducible and aperiodic stochastic matrices, applying both methods under the same settings. By observing the number of iterations required for convergence and the accuracy of the resulting eigenvectors, we are able to assess the comparative performance of the two methods. This provides not only empirical evidence for the reliability of the doubling algorithm but also insights into its behavior in relation to more traditional approaches.

To make the discussion more rigorous, we proceed by formally defining the notion of convergence for a matrix sequence.

Definition 3.1 ([4, Sec. 5.3 Definition p. 284])**.** *Suppose $L, A_1, A_2, \ldots$ are $n \times p$ matrices with complex entries. We say that the sequence $A_1, A_2, \ldots$* ***converges*** *to the matrix L, referred to as the* ***limit*** *of the sequence, if*

$$\lim_{m\to\infty} (A_m)_{ij} = L_{ij}$$

holds for every $1 \leq i \leq n$ and $1 \leq j \leq p$. In this case, we denote the convergence by

$$\lim_{m\to\infty} A_m = L.$$

Now, we introduce two lemmas related to the Jordan canonical form. Since not every matrix is diagonalizable, it is important to consider more general cases. In particular, when the algebraic multiplicity of an eigenvalue does not equal its geometric multiplicity, the matrix cannot be diagonalized but instead admits a Jordan canonical form. Therefore, we need to extend our discussion to include Jordan forms so that our results apply to all matrices.

The following two lemmas present key properties related to Jordan canonical forms and Jordan blocks. The first lemma provides fundamental results about the Jordan canonical form and the diagonalizability of matrices. It shows that the difference between the Jordan form and its diagonal part is nilpotent, that the difference commutes with the diagonal part, and that the powers of the Jordan canonical form can be expanded by applying the binomial theorem. The second lemma focuses specifically on Jordan blocks. It first establishes that the difference between a Jordan block and a scalar multiple of the identity matrix is nilpotent. Furthermore, it describes the explicit structure of the powers of a Jordan block. Moreover, it characterizes precisely when the sequence of powers of a Jordan block converges. These lemmas provide essential technical tools that will be employed in the subsequent theoretical developments and analyses.

Lemma 3.1 ([4, Sec. 7.2 Exercise 18. P. 513–514])**.** *Let T be a linear operator on a finite-dimensional vector space V, and let J be the Jordan canonical form of T. Let D be the diagonal matrix whose diagonal entries are the diagonal entries of J, and let $M = J - D$. We have the following results.*

(a) M is nilpotent.

(b) MD=DM.

(c) If p is the smallest positive integer for which $M^p = O$, then, for any positive integer $r < p$,

$$J^r = D^r + rD^{r-1}M + \frac{r(r-1)}{2!}D^{r-2}M^2 + \cdots + rDM^{r-1} + M^r, \tag{3.1}$$

and, for any positive integer $r \geq p$,

$$\begin{aligned} J^r = D^r + rD^{r-1}M + \frac{r(r-1)}{2!}D^{r-2}M^2 + \cdots \\ + \frac{r!}{(r-p+1)!(p-1)!}D^{r-p+1}M^{p-1}. \end{aligned} \tag{3.2}$$

Proof. (a) Denote D_k as the diagonal consisting of all (i, j)-entries such that $j - i = k$. In particular, D_0 is the main diagonal. Suppose that M is an upper triangular matrix with all zeros on D_0, then for any positive integer p, the entries of M^p in D_k are all zero for $0 \leq k < p$. This implies that for sufficiently large p, all entries of M^p become zero, so M is nilpotent.

(b) Since D is the diagonal matrix whose diagonal entries are the same as those of J, we can write $D = dI$ for some scalar d. Then we have

$$MD = M(dI) = dM = (dI)M = DM.$$

Thus, M and D commute, which completes the proof.

(c) Since M and D commute, we expand J^r using the binomial theorem:

$$J^r = (M + D)^r = \sum_{k=0}^{r} \binom{r}{k} M^k D^{r-k}.$$

If $r < p$, then equation (3.1) follows directly. If $r \geq p$, observing that $M^k = O$ for all $k \geq p$, the summation reduces appropriately, leading to equation (3.2) and completing the proof.

□

Lemma 3.2 ([4, Sec. 7.2 Exercise 19. P. 514]). *Let*

$$J = \begin{bmatrix} \lambda & 1 & 0 & \cdots & 0 \\ 0 & \lambda & 1 & \cdots & 0 \\ 0 & 0 & \lambda & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & \lambda \end{bmatrix}$$

be the $m \times m$ Jordan block associated with the eigenvalue λ. Define $N = J - \lambda I_m$. Then, we have the following results.

(a) $N^m = O$, and for $1 \leq r < m$,

$$N_{ij}^r = \begin{cases} 1, & \text{if } j = i + r, \\ 0, & \text{otherwise}. \end{cases}$$

(b) For any integer $r \geq m$,

$$J^r = \begin{bmatrix} \lambda^r & r\lambda^{r-1} & \frac{r(r-1)}{2!}\lambda^{r-2} & \cdots & \frac{r(r-1)\cdots(r-m+2)}{(m-1)!}\lambda^{r-m+1} \\ 0 & \lambda^r & r\lambda^{r-1} & \cdots & \frac{r(r-1)\cdots(r-m+3)}{(m-2)!}\lambda^{r-m+2} \\ 0 & 0 & \lambda^r & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda^r \end{bmatrix} \tag{3.3}$$

(c) $\lim_{r\to\infty} J^r$ *exists if and only if one of the following holds:*

(i) $|\lambda| < 1$.

(ii) $\lambda = 1$ *and* $m = 1$.

Proof. (a) Since $N = J - \lambda I_m$, multiplying N on the right shifts all columns one position to the right. Formally, for any power r, the entries of N^r satisfy the above condition. This follows from the structure of N, where each multiplication shifts nonzero entries to the right until they reach the last column or disappear. Thus, we complete this proof.

(b) Since $N = J - \lambda I_m$, we have $J = N + \lambda I_m$. By applying Lemma 3.1 (c) with $M = N$ and $D = \lambda I_m$, and in view of the preceding discussion, we arrive at equation (3.3).

(c) If $|\lambda| < 1$, then the limit is the zero matrix. If $\lambda = 1$ and $m = 1$, then the limit is the identity matrix of dimension 1. Conversely, if $|\lambda| \geq 1$ but $\lambda \neq 1$, then the diagonal entries do not converge. If $\lambda = 1$ but $m > 1$, the $(1,2)$-entry will diverge.

□

In the following theorem, we apply the result of Lemma 3.2(c) to easily derive the desired conclusion. This theorem defines the unit disk and establishes two important properties: first, that eigenvalues lying within the unit disk will converge to zero, and second, that the eigenvalue 1 has algebraic multiplicity one. These properties are concerned with whether the powers of the matrix converge as the exponent tends to infinity. Therefore, we need to verify whether the two required conditions are satisfied.

Theorem 3.3 ([4, Sec. 5.3 Theorem 5.13. P. 285])**.** *Let A be a square matrix with complex entries. Then* $\lim_{m\to\infty} A^m$ *exists if and only if both of the following conditions hold.*

(a) Every eigenvalue of A is contained in S, where

$$S = \{\lambda \in \mathbb{C} : |\lambda| < 1 \text{ or } \lambda = 1\}.$$

(b) If 1 *is an eigenvalue of A, then the dimension of the eigenspace corresponding to* 1 *equals the multiplicity of* 1 *as an eigenvalue of A.*

Proof. To study the convergence of a matrix power sequence, we begin by relating a matrix $A \in \mathbb{C}^{n\times n}$ to its Jordan canonical form. Specifically, there exists an invertible matrix Q such that

$$A = QJQ^{-1},$$

where J is the Jordan form of A. This implies

$$A^m = QJ^mQ^{-1},$$

so the limit $\lim_{m\to\infty} A^m$ exists if and only if $\lim_{m\to\infty} J^m$ exists.

Now, since J is a direct sum of Jordan blocks J_i, that is,

$$J = \bigoplus_i J_i,$$

we have

$$J^m = \bigoplus_i J_i^m.$$

Therefore, $\lim_{m\to\infty} J^m$ exists if and only if $\lim_{m\to\infty} J_i^m$ exists for every i. Hence, by applying Lemma 3.2 (c), the desired result is established. □

Before proceeding, we introduce the notions of *row sum* and *column sum*, which will be used repeatedly in the following statements. For clarity and completeness, we state these definitions explicitly here.

Definition 3.2 ([4, Sec. 5.3 Definitions. P. 295]). *Let $A \in M_{n\times n}(\mathbb{C})$. For each $1 \le i, j \le n$, define $\tau_i(A)$ as the sum of the absolute values of the entries in the ith row of A, and define $\nu_j(A)$ as the sum of the absolute values of the entries in the jth column of A. That is,*

$$\tau_i(A) = \sum_{j=1}^{n} |A_{ij}| \quad \text{for } i = 1, 2, \ldots, n,$$

and

$$\nu_j(A) = \sum_{i=1}^{n} |A_{ij}| \quad \text{for } j = 1, 2, \ldots, n.$$

*The **row sum** of A, denoted by $\tau(A)$, and the **column sum** of A, denoted by $\nu(A)$, are defined respectively as*

$$\tau(A) = \max\{\tau_i(A) : 1 \le i \le n\} \quad \text{and} \quad \nu(A) = \max\{\nu_j(A) : 1 \le j \le n\}.$$

The following lemma presents a fundamental property in linear algebra, describing the relationship between a matrix and its transpose. This result will play an important role in the proof of the subsequent corollary.

Lemma 3.4 ([4, Sec. 5.1 Exercise 14 p. 259]). *For any square matrix A, the transpose $A^\top$ has the same characteristic polynomial as A. Consequently, A and $A^\top$ have identical eigenvalues.*

Proof. It is known that $\det(A) = \det(A^\top)$ for any square matrix A. Consider the characteristic polynomial of A, given by $\det(A - \lambda I)$. Taking the transpose on both sides, we obtain

$$(A - \lambda I)^\top = A^\top - \lambda I^\top = A^\top - \lambda I.$$

Since the determinant of a matrix remains unchanged under transposition, we have

$$\det(A - \lambda I) = \det((A - \lambda I)^\top) = \det(A^\top - \lambda I).$$

which shows that A and $A^\top$ have the same characteristic polynomial. □

In the following, we introduce the concept of the Gerschgorin disk and the associated theorem. The Gerschgorin disk can be regarded as an analogue of the unit disk; however, it is defined by using the row sums of a matrix minus its corresponding diagonal entries. The associated theorem, known as the Gerschgorin disk theorem, asserts that all eigenvalues of a matrix lie within the union of its Gerschgorin disks.

Definition 3.3 ([4, Sec. 5.3 p. 296]). *(Gerschgorin Disk) Given an $n \times n$ matrix A, the ith **Gerschgorin disk** C_i is defined as the set of complex numbers z satisfying*

$$|z - A_{ii}| < r_i,$$

where the center of the disk is A_{ii}, and the radius is given by $r_i = \tau_i(A) - |A_{ii}|$, with $\tau_i(A)$ denoting the sum of the absolute values of the entries in the ith row of A. Thus,

$$C_i = \{z \in \mathbb{C} : |z - A_{ii}| < r_i\}.$$

Theorem 3.5 ([4, Sec. 5.3 Theorem 5.16. P. 296–297]). *(Gerschgorin's Disk Theorem) Let $A \in M_{n\times n}(\mathbb{C})$. Then every eigenvalue of A lies within at least one of the Gerschgorin disks associated with A.*

Proof. Let λ be an eigenvalue of $A \in \mathbb{R}^{n\times n}$, and let $v = [v_1, v_2, \ldots, v_n]^\top$ be a corresponding eigenvector such that $Av = \lambda v$. Since $v \neq 0$, at least one entry of v is nonzero. Let $k \in \{1, 2, \ldots, n\}$ be an index such that

$$|v_k| = \max_{1\le i\le n} |v_i|.$$

Then $v_k \neq 0$, and by examining the k-th entry of the equation $Av = \lambda v$, we obtain

$$\sum_{j=1}^{n} A_{kj} v_j = \lambda v_k.$$

This scalar equation will be useful for estimating or bounding the eigenvalue λ using the entries of A and the relative magnitudes of v_j. For $i = k$,

$$
\begin{aligned}
|\lambda v_k - A_{kk} v_k| &= \left| \sum_{j=1}^{n} A_{kj} v_j - A_{kk} v_k \right| = \left| \sum_{j \neq k} A_{kj} v_j \right| \\
&\leq \sum_{j \neq k} |A_{kj}||v_j| \leq \sum_{j \neq k} |A_{kj}||v_k| \\
&= |v_k| \sum_{j \neq k} |A_{kj}| = |v_k| r_k
\end{aligned}
$$

Thus

$$|v_k||\lambda - A_{kk}| \leq |v_k| r_k;$$

so

$$|\lambda - A_{kk}| \leq r_k$$

because $|v_k| > 0$. This completes the proof of Theorem 3.5. ☐

The next three corollaries are all consequences of the Gerschgorin Disk Theorem, with the third being the most significant. Specifically, they establish that all eigenvalues of a stochastic matrix have absolute value less than or equal to 1. This fact will play a crucial role in the subsequent analysis.

Corollary 3.1 ([4, Sec. 5.3 Corollary 1. P. 297]). *Let λ be an eigenvalue of $A \in M_{n\times n}(\mathbb{C})$. Then the modulus of λ is bounded above by the row sum of A; that is, $|\lambda| \leq \tau(A)$.*

Proof. By Gerschgorin's disk theorem, $|\lambda - A_{kk}| \leq r_k$ for some k. Hence

$$
\begin{aligned}
|\lambda| &= |(\lambda - A_{kk}) + A_{kk}| \leq |\lambda - A_{kk}| + |A_{kk}| \\
&\leq r_k + |A_{kk}| = \tau_k(A) \leq \tau(A).
\end{aligned}
$$

☐

Corollary 3.2 ([4, Sec. 5.3 Corollary 2. P. 297]). *Let $A \in M_{n\times n}(\mathbb{C})$, and let λ be any eigenvalue of A. Then the modulus of λ satisfies*

$$|\lambda| \leq \min\{\tau(A), \nu(A)\}, \tag{3.4}$$

where $\tau(A)$ and $\nu(A)$ denote the maximum absolute row sum and column sum of A, respectively.

Proof. Since $|\lambda| \leq \tau(A)$ by Corollary 3.1, if we can show that $|\lambda| \leq \nu(A)$, the proof is complete. By Lemma 3.4, we know that λ is also an eigenvalue of A^T. Using a similar argument and applying Corollary 3.1, we have

$$|\lambda| \leq \tau(A^\top) = \nu(A).$$

Therefore, $|\lambda| \leq \nu(A)$. Combining this with the earlier result, we obtain the equation (3.4). ☐

Corollary 3.3 ([4, Sec. 5.3 Corollary 3. P. 298]). *If λ is an eigenvalue of a stochastic matrix, then $|\lambda| \leq 1$.*

Proof. By Corollary 3.2, we have the equation (3.4). Noting that the column sum $\nu(A)$ is defined as the maximum of the column sums of A, and that each column sum of a stochastic matrix is 1, it follows that $\nu(A) = 1$. Thus, we obtain $|\lambda| \leq \min\{\tau(A), 1\}$. If $\tau(A) > 1$, then $|\lambda| \leq 1$ follows immediately. On the other hand, if $\tau(A) < 1$, then $|\lambda| \leq \tau(A) < 1$, and in particular $|\lambda| \leq 1$ also holds. Therefore, in both cases, we conclude that $|\lambda| \leq 1$. ☐

In the previous corollary, we established that all eigenvalues of a stochastic matrix have absolute value less than or equal to 1. The next lemma ensures that 1 is indeed an eigenvalue of a stochastic matrix. In particular, this implies that the spectral radius of a stochastic matrix is 1.

Lemma 3.6 ([4, Sec. 5.3 Theorem 5.17. P. 298]). *Every stochastic matrix has 1 as an eigenvalue.*

Proof. Let A be an $n\times n$ stochastic matrix, and let $u \in \mathbb{R}^n$ be the column vector where each coordinate is 1. By the properties of stochastic matrices, we have $A^\top u = 1u$. This shows that u is an eigenvector of $A^\top$ corresponding to the eigenvalue 1. Furthermore, by Lemma 3.4, which states that A and $A^\top$ share the same eigenvalues, it follows that 1 is also an eigenvalue of A. Therefore, the proof of Lemma 3.6 is complete. ☐

In the following discussion, we introduce several fundamental properties of the inner product. While linear algebra generally considers vector spaces over arbitrary fields, in this context we specialize to the complex field $\mathbb{C}$. First, we introduce some basic definitions related to the inner product.

Definition 3.4 ([4, Sec. 6.1 Definition. P. 329–330]). *Let V be a vector space over $\mathbb{C}$. An* ***inner product*** *on V is a function that assigns, to every ordered pair of vectors x and y in V, a scalar in F, denoted $\langle x, y\rangle$, such that for all x, y, and z in V and all c in $\mathbb{C}$, the following hold:*

(a) $\langle x+z, y\rangle = \langle x, y\rangle + \langle z, y\rangle$.

(b) $\langle cx, y\rangle = c\langle x, y\rangle$.

(c) $\overline{\langle x, y\rangle} = \langle y, x\rangle$, *where the bar denotes complex conjugation.*

(d) $\langle x, x\rangle > 0$ *if* $x \neq 0$.

Based on the definitions, we can derive some additional properties. Thus, we present the following lemma to state these results. Although the proofs are elementary, the properties will be useful in the subsequent discussion. In particular, the proofs frequently utilize the conjugate symmetry of the inner product.

Lemma 3.7 ([4, Sec. 6.1 Theorem 6.1. P. 333]). *Let V be an inner product space. Then for all $x, y, z \in V$ and $c \in \mathbb{C}$,the following statements are true.*

(a) $\langle x, y+z\rangle = \langle x, y\rangle + \langle x, z\rangle$.

(b) $\langle x, cy\rangle = \bar{c}\langle x, y\rangle$.

(c) $\langle x, 0\rangle = \langle 0, x\rangle = 0$.

(d) $\langle x, x\rangle = 0$ *if and only if* $x = 0$.

(e) If $\langle x, y\rangle = \langle x, z\rangle$ for all $x \in V$, then $y = z$.

Proof. We now prove each statement as follows.

(a) $\langle x, y+z\rangle = \overline{\langle y+z, x\rangle} = \overline{\langle y, x\rangle + \langle z, x\rangle} = \langle x, y\rangle + \langle x, z\rangle$.

(b) $\langle x, cy\rangle = \overline{\langle cy, x\rangle} = \overline{c\langle y, x\rangle} = \overline{c}\overline{\langle y, x\rangle} = \overline{c}\langle x, y\rangle$.

(c) $\langle x, 0\rangle = 0$ and $\langle 0, x\rangle = 0$, since the inner product with the zero vector is always zero.

(d) If $x = 0$, then by the previous rule, we have $\langle 0, 0\rangle = 0$. If $x \neq 0$, then $\langle x, x\rangle \neq 0$ by the properties of the inner product.

(e) If $\langle x, y\rangle = \langle x, z\rangle$ for all $x \in V$, then we have $\langle x, y-z\rangle = 0$ for all $x \in V$. By the properties of the inner product, this implies that $\langle y-z, y-z\rangle = 0$. Using the previous rule, it follows that $y - z = 0$. Hence, $y = z$.

□

In the next definition, we introduce the notion of the norm, or length, induced by the inner product, which is a crucial concept in the theory of inner product spaces.

Definition 3.5 ([4, Sec. 6.1 Definition. P. 333]). *Let V be an inner product space. For $x \in V$, we define the* ***norm*** *or* ***length*** *of x by $\|x\| = \sqrt{\langle x, x\rangle}$.*

In the following lemma, we state results related to the norm, or length, induced by the inner product. In particular, we focus on the proofs of the Cauchy–Schwarz Inequality and the triangle inequality, both of which will be needed in the subsequent discussion.

Lemma 3.8 ([4, Sec. 6.1 Theorem 6.2. P. 333–334]). *Let V be an inner product space over $\mathbb{C}$. Then for all $x, y \in V$ and $c \in \mathbb{C}$,the following statements hold.*

(a) $\|cx\| = |c| \cdot \|x\|$.

(b) $\|x\| = 0$ *if and only if* $x = 0$. *In any case,* $\|x\| \geq 0$.

(c) (Cauchy–Schwarz Inequality) $\big|\langle x, y\rangle\big| \leq \|x\| \cdot \|y\|$.

(d) (Triangle Inequality) $||x+y|| \leq ||x|| + ||y||$.

Proof. The proofs of the following results are presented below.

(a) $\|cx\| = \langle cx, cx\rangle^{\frac{1}{2}} = (c\bar{c}\langle x, x\rangle)^{\frac{1}{2}} = |c| \cdot \|x\|$.

(b) This follows from the definition $\|x\| = \langle x, x\rangle^{\frac{1}{2}}$ and, by Lemma 3.7 (d), we obtain that $\|x\| = 0$ if and only if $x = 0$.

(c) If $y = 0$, then the result obtain immediate. Assume that $y \neq 0$. For any $c \in \mathbb{C}$, we have

$$\begin{aligned} 0 \leq \|x - cy\|^2 &= \langle x - cy, x - cy\rangle = \langle x, x - cy\rangle - c\langle y, x - cy\rangle \\ &= \langle x, x\rangle - \bar{c}\langle x, y\rangle - c\langle y, x\rangle + c\bar{c}\langle y, y\rangle. \end{aligned}$$

In particular, if we let

$$c = \frac{\langle x, y\rangle}{\langle y, y\rangle},$$

the inequality becomes

$$0 \leq \langle x, x\rangle - \frac{|\langle x, y\rangle|^2}{\langle y, y\rangle} = \|x\|^2 - \frac{|\langle x, y\rangle|^2}{\|y\|^2}.$$

Hence, we deduce that $|\langle x, y\rangle|^2 \leq \|x\|^2\|y\|^2$, and therefore we obtain the Cauchy–Schwarz inequality $|\langle x, y\rangle| \leq \|x\| \cdot \|y\|$.

(d) We have

$$\begin{aligned} \|x + y\|^2 &= \langle x + y, x + y\rangle \\ &= \langle x, x\rangle + \langle y, x\rangle + \langle x, y\rangle + \langle y, y\rangle \\ &= \|x\|^2 + 2\,\mathrm{Re}\langle x, y\rangle + \|y\|^2 \\ &\leq \|x\|^2 + 2|\langle x, y\rangle| + \|y\|^2 \\ &\leq \|x\|^2 + 2\|x\| \cdot \|y\| + \|y\|^2 \\ &= (\|x\| + \|y\|)^2. \end{aligned}$$

Here, $\mathrm{Re}\langle x, y\rangle$ denotes the real part of the complex number $\langle x, y\rangle$. Taking the square root on both sides, we obtain the Triangle Inequality:

$$\|x + y\| \leq \|x\| + \|y\|.$$

□

In the following lemma, we discuss the conditions under which equality holds in the Cauchy–Schwarz Inequality. In addition, we observe when equality occurs in the triangle inequality and in its generalized form.

Lemma 3.9 ([4, Sec. 6.1 Exercise 15. P. 337–338])**.** *Let V be an inner product space.*

(a) Prove that $|\langle x, y\rangle| = \|x\| \cdot \|y\|$ if and only if one of the vectors x or y is a scalar multiple of the other.

(b) Derive a similar result for the equality $\|x + y\| = \|x\| + \|y\|$, and generalize this result to the case of n vectors.

Proof. The following statements will be proved as follows.

(a) First, we need to prove that if one of the vectors x or y is a multiple of the other, then $|\langle x, y\rangle| = \|x\| \cdot \|y\|$. If one of x or y is zero, say $y = 0 \cdot x = 0$ or $x = 0 \cdot y = 0$, the equality clearly holds. Now, consider the case where $y \neq 0$ and $x = cy$ for some scalar c. Then we have

$$|\langle x, y\rangle| = |\langle cy, y\rangle| = |c| \cdot \|y\|^2$$

and

$$\|x\| \cdot \|y\| = \|cy\| \cdot \|y\| = |c| \cdot \|y\|^2.$$

Thus, the equality holds, and we complete the proof of the first statement. Therefore, we need to prove that if $|\langle x, y\rangle| = \|x\| \cdot \|y\|$, then one of the vectors x or y is a multiple of the other. By the proof of Lemma 3.8 (c), the equality holds if and only if $\|x - cy\| = 0$ for some scalar c. Since the norm of a vector is zero only when the vector itself is zero, this simplifies to $x = cy$, completing the proof.

(b) By the proof of Lemma 3.8 (d), we have the equality $\mathrm{Re}\langle x, y\rangle = |\langle x, y\rangle| = \|x\|\|y\|$. The case where $y = 0$ is trivial. Now, assume $y \neq 0$ and that $x = cy$ for some scalar $c \in \mathbb{C}$. Then we have

$$\mathrm{Re}(c)\|y\|^2 = \mathrm{Re}\langle cy, y\rangle = |\langle cy, y\rangle| = |c| \cdot \|y\|^2,$$

and so $\mathrm{Re}(c) = |c|$. This implies that c is a nonnegative real number. If $x = cy$ for some nonnegative real number c, then

$$\|x + y\| = |c + 1| \cdot \|y\| = (c + 1)\|y\|$$

and

$$\|x\| + \|y\| = |c| \cdot \|y\| + \|y\| = (c + 1)\|y\|.$$

Hence, we obtain the equality $\|x + y\| = \|x\| + \|y\|$, completing the proof. Finally, we extend the equality $\|x_1 + x_2 + \cdots + x_n\| = \|x_1\| + \|x_2\| + \cdots + \|x_n\|$. This equality holds if and only if there exists a vector among $\{x_1, x_2, \ldots, x_n\}$ such that all other vectors are nonnegative real multiples of that vector.

□

In the next theorem, we discuss a crucial property. The statement asserts that if the absolute value of an eigenvalue equals the row sum, then the eigenvalue itself equals the row sum, and then its geometric eigenspace is one-dimensional and is spanned by the vector u. This theorem is a powerful tool for determining the geometric multiplicity of a stochastic matrix. Furthermore, combining this result with information about the algebraic multiplicity enables us to obtain a deeper understanding of the structure of stochastic matrices.

Theorem 3.10 ([4, Sec. 5.3 Theorem 5.18. P. 298–299])**.** *Let $A \in M_{n\times n}(\mathbb{C})$ be a matrix in which each entry is positive, and let λ be an eigenvalue of A such that $|\lambda| = \tau(A)$. Then $\lambda = \tau(A)$ and $\{u\}$ is a basis of E_λ, where $u \in \mathbb{C}^n$ is the column vector in which each coordinate equals 1.*

Proof. Let v be an eigenvector of A corresponding to the eigenvalue λ, with coordinates $v_1, v_2, \ldots, v_n$. Suppose that v_k is the coordinate of v with the largest absolute value, and let $b = |v_k|$. Then

$$\begin{aligned} |\lambda|b = |\lambda||v_k| = |\lambda v_k| &= \left|\sum_{j=1}^{n} A_{kj}v_j\right| \leq \sum_{j=1}^{n}|A_{kj}v_j| \\ &= \sum_{j=1}^{n}|A_{kj}||v_j| \leq \sum_{j=1}^{n}|A_{kj}|b = \tau_k(A)b \leq \tau(A)b. \end{aligned} \tag{3.5}$$

Since $|\lambda| = \tau(A)$, the three inequalities from equation (3.5) become equalities; that is,

(a) $\left|\sum_{j=1}^{n} A_{kj}v_j\right| = \sum_{j=1}^{n}|A_{kj}v_j|$,

(b) $\sum_{j=1}^{n}|A_{kj}||v_j| = \sum_{j=1}^{n}|A_{kj}|b$, and

(c) $\tau_k(A) = \tau(A)$.

By the equalities given in Lemma 3.9 (b) and in item (a) established earlier, we obtain an important result, all the terms $A_{kj}v_j$ for $j = 1, 2, \ldots, n$ are nonnegative multiples of some nonzero complex number z. Without loss of generality, we assume that $|z| = 1$. Thus, there exist nonnegative real numbers $c_1, c_2, \ldots, c_n$ such that

$$A_{kj}v_j = c_j z. \tag{3.6}$$

By the equality in (b) and the assumption that $A_{kj} \neq 0$ for all k and j, we have

$$|v_j| = b \quad \text{for } j = 1, 2, \ldots, n. \tag{3.7}$$

Combining (3.6) and (3.7), we obtain

$$b = |v_j| = \left|\frac{c_j}{A_{kj}}z\right| = \frac{c_j}{A_{kj}} \quad \text{for } j = 1, 2, \ldots, n,$$

and therefore by (3.6), we have $v_j = bz$ for all j. So

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} bz \\ bz \\ \vdots \\ bz \end{bmatrix} = bzu,$$

Thus, $\{u\}$ is a basis for the eigenspace E_λ. Finally, according to the given condition, A is a matrix in which each entry is positive, and u is a column vector with all positive entries. Since we know that $Au = \lambda u$, it follows that λ must be positive. Therefore, we have $\lambda = |\lambda| = \tau(A)$. $\square$

In the following two corollaries, we connect the crucial theorem with the properties of stochastic matrices. In particular, we establish that if a stochastic matrix has all positive entries, then the eigenvalue 1 has geometric multiplicity one.

Corollary 3.4 ([4, Sec. 5.3 Corollary 1. P. 299]). *Let $A \in M_{n\times n}(\mathbb{C})$ be a matrix with all entries positive, and let λ be an eigenvalue of A such that $|\lambda| = \nu(A)$. Then it follows that $\lambda = \nu(A)$, and the eigenspace E_λ corresponding to λ is one-dimensional.*

Proof. Applying Theorem 3.10 to the matrix $A^\top$ and using Lemma 3.4, we know that A and $A^\top$ have the same eigenvalues. Consequently, $\lambda = \nu(A)$, and the dimension of the eigenspace of $A^\top$ corresponding to λ is 1. $\square$

Corollary 3.5 ([4, Sec. 5.3 Corollary 2. P. 299]). *Let $A \in M_{n\times n}(\mathbb{C})$ be a stochastic matrix with all entries positive. Then every eigenvalue λ of A other than 1 satisfies $|\lambda| < 1$. Furthermore, the eigenspace associated with the eigenvalue 1 is one-dimensional.*

Proof. Since A is a stochastic matrix, we know that $\nu(A) = 1$. If $\lambda \neq 1$, then by Corollary 3.3 and Corollary 3.4, we have $|\lambda| < 1$. Moreover, the eigenspace corresponding to the eigenvalue 1 has dimension 1 by Corollary 3.4. $\square$

Before presenting the formal definitions, we briefly highlight the significance of several fundamental concepts related to stochastic matrices—namely, irreducibility, aperiodicity, and primitivity. These properties play a crucial role in determining the long-term behavior of stochastic processes and are central to many convergence results in matrix analysis.

Definition 3.6 ([10]). *Let $A \in \mathbb{R}^{n\times n}$ be a stochastic matrix. Then:*

- *A is called **irreducible** if, for all indices i, j, there exists a positive integer k such that $(A^k)_{ij} > 0$. In other words, every state is reachable from every other state in some finite number of steps.*
- *A is called **aperiodic** if, for each index i, the greatest common divisor of the set $\{k \in \mathbb{N} \mid (A^k)_{ii} > 0\}$ is equal to 1. That is, the system can return to any given state at irregular time intervals.*
- *A is called **primitive** if there exists a positive integer m such that $A^m > 0$, i.e., all entries of A^m are strictly positive.*

In the following lemma, we introduce the relationship between irreducibility, aperiodicity, and primitivity. This connection is essential because the next theorem assumes that the matrix is nonnegative, irreducible, and aperiodic, and under these conditions, the matrix is equivalent to being primitive. The lemma below formally establishes this equivalence.

Lemma 3.11 ([5]). *Let $A \in \mathbb{R}^{n\times n}$ be a nonnegative matrix. Then A is primitive if and only if A is irreducible and aperiodic.*

Proof. We prove both directions of the equivalence. First, we prove the direction from primitivity to irreducibility and aperiodicity. Suppose A is primitive. Then there exists a positive integer s such that $A^s > 0$, meaning that all entries of A^s are strictly positive. This implies that for every pair of indices i, j, we have $(A^s)_{ij} > 0$, so there exists a path of length s from state i to state j. Hence, A is irreducible. Moreover, for each index i, we have $(A^s)_{ii} > 0$, and since $A^{s+1} = A^sA > 0$, it follows that $(A^{s+1})_{ii} > 0$ as well. Therefore, the set $\{k \in \mathbb{N} \mid (A^k)_{ii} > 0\}$ contains two consecutive integers s and $s+1$, and hence has greatest common divisor equal to 1. Thus, every state has period 1, and A is aperiodic.

Second, we prove the direction from irreducibility and aperiodicity to primitivity. Since A is irreducible, for any pair of indices i, j, there exists a positive integer k_{ij} such that $(A^{k_{ij}})_{ij} > 0$. In other words, each state is reachable from every other state in some number of steps. Moreover, since A is aperiodic, for each index i, the set $\{k \in \mathbb{N} \mid (A^k)_{ii} > 0\}$ has greatest common divisor equal to 1. By a classical result in number theory, this implies that there exists an integer N_i such that for all $m \geq N_i$, we have $(A^m)_{ii} > 0$. Now fix any pair i, j. By irreducibility, choose k_0 such that $(A^{k_0})_{ij} > 0$, and by aperiodicity, there exists a sufficiently large m such that $(A^{m-k_0})_{jj} > 0$. Then,

$$(A^m)_{ij} \geq (A^{k_0})_{ij} \cdot (A^{m-k_0})_{jj} > 0.$$

Since i, j are arbitrary and m can be chosen large enough to satisfy all such combinations, there exists a common integer m_0 such that $A^{m_0} > 0$. Therefore, A is primitive. This completes the proof of Lemma 3.11. □

In the next theorem, we extend the result of Corollary 3.5. We relax the condition by requiring only that the stochastic matrix is irreducible and aperiodic, without assuming that all entries are initially positive. We establish that an irreducible and aperiodic stochastic matrix also possesses the same property.

Theorem 3.12 ([4, Sec. 5.3 Theorem 5.19. P. 300]). *Let A be an irreducible and aperiodic stochastic matrix, and let λ be any eigenvalue of A. Then:*

(a) $|\lambda| \leq 1$.

(b) If $|\lambda| = 1$, then $\lambda = 1$, and the eigenspace corresponding to λ is one-dimensional; that is, $\dim(E_\lambda) = 1$.

Proof. We now provide the proof of the following statements.

(a) Since A is a stochastic matrix, the statement follows directly from Corollary 3.3.

(b) Since A is irreducible and aperiodic, it follows from Lemma 3.11 that there exists a positive integer s such that A^s has all entries strictly positive. Since A is a stochastic matrix and the entries of A^s are positive, it follows that the entries of $A^{s+1} = A^s A$ are also positive. Now, suppose that $|\lambda| = 1$. Then, λ^s and λ^{s+1} are eigenvalues of A^s and A^{s+1}, respectively, both having absolute value 1. By Corollary 3.5, we obtain $\lambda^s = \lambda^{s+1} = 1$, which implies $\lambda = 1$. Let E_λ and $E_\lambda^{'}$ denote the eigenspaces of A and A^s, respectively, corresponding to $\lambda = 1$. We want to show that $E_\lambda \subseteq E_\lambda'$. Indeed, if $v \in E_\lambda$, then by definition, we have $Av = v$. Applying A^s to v, we get

$$A^s v = (A^{s-1}A)v = A^{s-1}(Av) = A^{s-1}v = \cdots = v.$$

Thus, $v \in E_\lambda^{'}$, which proves that $E_\lambda \subseteq E_\lambda^{'}$. By Corollary 3.5, we have $\dim(E_\lambda^{'}) = 1$. Hence, it follows that

$$E_\lambda = E_\lambda^{'} \quad \text{and} \quad \dim(E_\lambda) = 1.$$

□

In the following discussion, we focus on the algebraic multiplicity of stochastic matrices. Although we previously established that the eigenvalue 1 of a stochastic matrix has geometric multiplicity one, this condition alone is not sufficient to guarantee the convergence of the powers of the matrix as the exponent tends to infinity. Thus, we must further verify that the eigenvalue 1 also has algebraic multiplicity one. To this end, we first introduce the definition of the matrix norm and then present a lemma to state some fundamental properties of matrix norms.

Definition 3.7 ([4, Sec. 7.2 Definition. P. 515]). *For any $A \in M_{n\times n}(\mathbb{C})$, define the norm of A by*

$$||A|| = \max\{|A_{ij}| : 1 \leq i, j \leq n\}.$$

Lemma 3.13 ([4, Sec. 7.2 Exercise 20. P. 515]). *Let $A, B \in M_{n\times n}(\mathbb{C})$. The following properties are satisfied.*

(a) $||A|| \geq 0$ *and* $||A|| = 0$ *if and only if* $A = O$.

(b) $||cA|| = |c| \cdot ||A||$ *for any scalar c.*

(c) $||A + B|| \leq ||A|| + ||B||$.

(d) $||AB|| \leq n||A||||B||$.

Proof. We now present the proof of the following statements.

(a) The norm $\|A\| \geq 0$ since $|A_{ij}| \geq 0$ for all i and j. Moreover, $\|A\| = 0$ if and only if $|A_{ij}| = 0$ for all i and j. Therefore, we conclude that $\|A\| = 0$ if and only if $A = O$.

(b) Compute
$$\|cA\| = \max\{|cA_{ij}|\} = |c|\max\{|A_{ij}|\} = |c|\|A\|.$$

(c) By the triangle inequality, we have
$$|A_{ij} + B_{ij}| \leq |A_{ij}| + |B_{ij}|, \quad \text{for all } i \text{ and } j.$$
Taking the maximum on both sides, we obtain
$$\|A + B\| \leq \|A\| + \|B\|.$$

(d) Compute
$$\begin{aligned}\|AB\| &= \max\{|(AB)_{ij}|\} = \max\{\left|\sum_{k=1}^{n} A_{ik}B_{kj}\right|\} \\ &\leq \max\{\left|\sum_{k=1}^{n} \|A\|\|B\|\right|\} = n|A||B|.\end{aligned}$$

□

The next lemma establishes that the eigenvalue 1 of a stochastic matrix has algebraic multiplicity one. This is a crucial result in the study of stochastic matrices, and it will be used to demonstrate that the powers of the matrix converge as the exponent tends to infinity.

Lemma 3.14 ([4, Sec. 7.2 Exercise 21. P. 515]). *Let $A \in M_{n\times n}(\mathbb{C})$ be a stochastic matrix. Since $\mathbb{C}$ is an algebraically closed field, A is similar to its Jordan canonical form J; that is, there exists an invertible matrix P such that $P^{-1}AP = J$. The following results hold:*

(a) $\|A^m\| \leq 1$ for every positive integer m.

(b) There exists a constant $c > 0$ such that $\|J^m\| \leq c$ for all positive integers m.

(c) Every Jordan block of J corresponding to the eigenvalue $\lambda = 1$ is of size 1×1.

(d) The limit $\lim\limits_{m\to\infty} A^m$ exists if and only if 1 is the only eigenvalue of A with modulus equal to 1.

Proof. We proceed to prove the following statements.

(a) Since A is a stochastic matrix, then A^m is also a stochastic matrix for every positive integer m. By the definition of norm and the properties of stochastic matrices, all entries of A^m are less than or equal to 1. Hence, we have
$$\|A^m\| \leq 1, \quad \text{for every positive integer } m.$$

(b) By applying Lemma 3.13 (d) and the previous result, we obtain
$$\|J^m\| = \|P^{-1}A^mP\| \leq n^2\|P^{-1}\|\|A^m\|\|P\| \leq n^2\|P^{-1}\|\|P\|.$$
Letting $c = n^2\|P^{-1}\|\|P\|$, we complete the proof.

(c) By the previous argument, the norm $\|J^m\|$ is bounded. Suppose that J_1 is a Jordan block corresponding to the eigenvalue 1, and its size is greater than 1. Then, the $(1, 2)$-entry of J_1^m grows unbounded as m increases, which contradicts the previous argument. Therefore, each Jordan block of J corresponding to the eigenvalue $\lambda = 1$ must be a 1×1 matrix.

(d) By Theorem 3.12 (b), we have $\dim(E_1) = 1$. Combining this with the item (c), it follows that $K_1 = E_1$. Consequently, the multiplicity of the eigenvalue 1 is given by $\dim(K_1) = \dim(E_1) = 1$. This completes the proof.

□

In this section, we establish that the eigenvalue 1 of an irreducible and aperiodic stochastic matrix has both algebraic and geometric multiplicities equal to one. In the final theorem of this section, we summarize several important properties of irreducible and aperiodic stochastic matrices, which have been obtained through the preceding results. These properties will later be utilized, in a subsequent section, to apply the doubling algorithm for further investigating the behavior of stochastic matrices.

Theorem 3.15 ([4, Sec. 5.3 Theorem 5.20. P. 300–301])**.** *Let A be an $n \times n$ irreducible and aperiodic stochastic matrix. Then the following statements hold:*

(a) The eigenvalue 1 has algebraic multiplicity 1, and its eigenspace is one-dimensional.

(b) The limit $\lim\limits_{m\to\infty} A^m$ exists.

(c) The matrix $L = \lim\limits_{m\to\infty} A^m$ is itself a stochastic matrix.

(d) The matrix L is invariant under multiplication by A; that is, $AL = LA = L$.

(e) All columns of L are identical. Specifically, each column equals the unique probability vector v such that $Av = v$.

(f) For any probability vector w, the sequence $A^m w$ converges to v; that is, $\lim\limits_{m\to\infty} A^m w = v$.

Proof. We provide the following proofs.

(a) By Lemma 3.14 (d), we have this statement.

(b) By Lemma 3.14 (d), Theorem 3.12, and Theorem 3.3, we have this statement.

(c) Let u be the column vector in which each coordinate equals 1. We want to show that $u^\top L = u^\top$. Since A is the stochastic matrix, we have $u^\top A = u^\top$. Thus,

$$u^\top L = u^\top \lim_{m\to\infty} A^m = \lim_{m\to\infty} u^\top A^m = \lim_{m\to\infty} u^\top = u^\top,$$

which implies that L is a stochastic matrix.

(d) We compute

$$AL = A \lim_{m\to\infty} A^m = \lim_{m\to\infty} AA^m = \lim_{m\to\infty} A^{m+1} = L.$$

Similarly, $LA = L$.

(e) Since $AL = L$ by (d), each column of L is an eigenvector of A corresponding to the eigenvalue 1. Moreover, by (c), each column of L is a probability vector. Thus, by (a), each column of L must be equal to the unique probability vector v corresponding to the eigenvalue 1 of A.

(f) Let w be any probability vector, and set $y = \lim\limits_{m\to\infty} A^m w = Lw$. Then y is a probability vector, and by (d), we have

$$Ay = ALw = Lw = y.$$

Hence, y is an eigenvector of A corresponding to the eigenvalue 1. By (e), we conclude that $y = v$.

This completes the proof of Theorem 3.15. □

In this section, we introduce the concept of an irreducible and aperiodic stochastic matrix. In Theorem 3.15, we establish a stability property of the associated vector sequence, which serves as a foundation for the development of the PageRank algorithm. This algorithm is closely related to irreducible and aperiodic stochastic matrices. Before formally introducing PageRank, we first present the definition of the random walk model, which serves as a probabilistic framework underlying the computation of stationary distributions in Markov chains.

Definition 3.8 ([6])**.** *Let $v_1, \ldots, v_n$ be the vertices of a directed graph with edge set E, and let $\overrightarrow{e_{ij}}$ denote the directed edge from vertex v_i to vertex v_j. Assume that the stochastic matrix induced by the graph assigns equal probability to each out-neighbor of a vertex. Then the associated row-stochastic matrix $M = [m_{ij}] \in \mathbb{R}^{n\times n}$ is defined by*

$$m_{ij} = \begin{cases} \frac{1}{\deg^+(v_i)}, & \text{if } \overrightarrow{e_{ij}} \in E, \\ 0, & \text{otherwise}, \end{cases}$$

where $\deg^+(v_i)$ denotes the out-degree of vertex v_i.

Let G be a strongly connected and aperiodic directed graph, and let M be the associated stochastic matrix. Consider a random walk on G, and let $\mathbf{r}_0$ denote the initial probability distribution over the vertices. Then the probability distributions at times $0, 1, \ldots, t, \ldots$ evolve as

$$\mathbf{r}_0, \quad \mathbf{r}_1 = M\mathbf{r}_0, \quad \mathbf{r}_2 = M^2\mathbf{r}_0, \quad \ldots, \quad \mathbf{r}_t = M^t\mathbf{r}_0, \quad \ldots$$

Since M is irreducible and aperiodic, the sequence $\{M^t\mathbf{r}_0\}$ converges to a unique stationary distribution

$$\lim_{t\to\infty} M^t\mathbf{r}_0 = \mathbf{r},$$

where $\mathbf{r}$ satisfies $M\mathbf{r} = \mathbf{r}$. This stationary distribution $\mathbf{r}$ is referred to as the *PageRank* of the directed graph associated with M. It corresponds to the result established in part (f) of Theorem 3.15.

In this section, we have examined irreducible and aperiodic stochastic matrices, which exhibit a well-understood eigenvalue structure. Such matrices have a unique eigenvalue $\lambda = 1$ with multiplicity one, and all other eigenvalues have absolute values strictly less than one. As the matrix is repeatedly applied, the influence of the smaller eigenvalues gradually diminishes, and the behavior of the system converges toward the eigenvector associated with $\lambda = 1$, also known as the steady-state distribution.

The doubling algorithm leverages this property: with each iteration, the components corresponding to non-dominant eigenvalues decay toward zero, while the part associated with $\lambda = 1$ remains unchanged. This enables the algorithm to efficiently isolate the dominant eigenvector.

To evaluate the performance and behavior of the doubling algorithm, we now compare it with a classical iterative method—namely, the *power method*. Although both methods share the same goal of identifying the dominant eigenvector, they approach the problem in fundamentally different ways. In the following section, we provide a detailed explanation of the power method and analyze its effectiveness when applied to irreducible and aperiodic stochastic matrices.

4 Power method

In this section, we begin by reviewing a fundamental technique in matrix computation, namely the *power method*. This classical algorithm is widely used to compute the dominant eigenvalue and its corresponding eigenvector of a matrix. The power method serves as a foundation for our subsequent discussion, as its convergence behavior, normalization strategy, and theoretical analysis share strong similarities with the doubling algorithm developed later in this work. For this reason, we introduce it here as a point of reference. To begin with, we formally define the notion of the dominant eigenvalue of a matrix, along with its corresponding eigenvector.

Definition 4.1. *If the distinct eigenvalues of a matrix A are λ_1, λ_2,...,λ_k, and if $|\lambda_1|$ is larger than $|\lambda_2|$,...,$|\lambda_k|$, then λ is called a* ***dominant eigenvalue*** *of A. Any eigenvector corresponding to a dominant eigenvalue is called a* ***dominant eigenvector*** *of A.*

In the following introduction, we outline the key idea behind the power method. Assuming that the matrix is diagonalizable, it can be expressed in terms of its eigenvalues and eigenvectors. Under this condition, repeated multiplication by the matrix amplifies the component associated with the dominant eigenvalue, while the contributions from the remaining eigenvalues, whose magnitudes are strictly less than that of the dominant one, decay geometrically. As the iteration count $k \to \infty$, these smaller components vanish, allowing the method to recover the eigenvector corresponding to the dominant eigenvalue. This fundamental mechanism also appears in our doubling algorithm, which motivates the introduction of this concept as a conceptual foundation. Let A be a diagonalizable matrix, $Ax_i = \lambda x_i$, $i = 1, 2, \dots, n$

$$Ax_i = \lambda_i x_i, \text{ for } i = 1, \dots, n$$

with

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n| \tag{4.1}$$

and let $u_0 \neq 0$ be a given vector. From the expansion

$$u_0 = \sum_{i=1}^{n} \alpha_i x_i \tag{4.2}$$

follows that

$$A^k u_0 = \sum_{i=1}^{n} \alpha_i A^k x_i = \sum_{i=1}^{n} \alpha_i \lambda_i^k x_i = \lambda_1^k \left\{ \alpha_1 x_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k x_i \right\} \tag{4.3}$$

Thus, the sequence $\lambda_1^{-k} A^k u_0$ converges to a multiple of x_1. We consider two possible normalizations: one based on the two-norm of a given vector and one based on a linear functional.

In the following discussion, we introduce two variants of the power method, each presented in its own subsection. The first variant is based on normalization using the two-norm of the iterated vector, while the second employs a linear functional to perform normalization. These methods differ in how they adjust the vector at each iteration, and this adjustment mechanism closely resembles that used in our doubling algorithm. For both variants, we present the corresponding algorithm and state a theorem that establishes convergence to the dominant eigenvalue and its associated eigenvector. Since the underlying proof techniques are highly analogous to those used in our doubling framework, particular attention is given to the reasoning of these proofs.

4.1 Two-norm of a given vector

We consider the classical power method, in which the normalization at each step is performed using the Euclidean norm (two-norm). The iteration proceeds as follows:

$$\begin{aligned}
&\text{For } k = 0, 1, 2, \cdots : \\
&\quad v_{k+1} = A u_k, \\
&\quad \mu_{k+1} = \|v_{k+1}\|_2 , \\
&\quad u_{k+1} = \frac{v_{k+1}}{\mu_{k+1}} \quad \text{with initial } u_0. \\
&\text{End}
\end{aligned} \tag{4.4}$$

Theorem 4.1 ([7])**.** *Under the assumption* (4.1) *and the condition* $\alpha_1 \neq 0$ *in* (4.2)*, the equation defined by* (4.4) *satisfies the following.*

$$\lim_{k\to\infty} \mu_k = |\lambda_1|,$$
$$\lim_{k\to\infty} \epsilon^k u_k = \frac{x_1}{\|x_1\|}\frac{\alpha_1}{|\alpha_1|}, \ \textit{where } \epsilon = \frac{|\lambda_1|}{\lambda_1}.$$

Proof. Based on the definition of (4.4), it follows that

$$u_k = \frac{A^k u_0}{\|A^k u_0\|}, \ \mu_k = \frac{\|A^k u_0\|}{\|A^{k-1} u_0\|}.$$

By the results from equation (4.3), we obtain $\lambda_1^{-k} A^k u_0 \to \alpha_1 x_1$. By mathematical induction, we derive two equations.

$$|\lambda_1|^{-k} \|A^k u_0\| \to |\alpha_1| \|x_1\|,$$
$$|\lambda_1|^{-k+1} \|A^{k-1} u_0\| \to |\alpha_1| \|x_1\|,$$

by dividing the two equations, we obtain the following equation

$$\frac{|\lambda_1|^{-1} \|A^k u_0\|}{\|A^{k-1} u_0\|} = |\lambda_1|^{-1} \mu_k \to 1.$$

Therefore, we conclude that

$$\lim_{k\to\infty} \mu_k = |\lambda_1|.$$

From the results of equation (4.3) and under the condition $k \to \infty$, we obtain the following equation

$$\begin{aligned}
\epsilon^k u_k = \epsilon^k \frac{A^k u_0}{\|A^k u_0\|} &= \frac{\alpha_1 x_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1}\right)^k x_i}{\|\alpha_1 x_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1}\right)^k x_i\|} \\
&\to \frac{\alpha_1 x_1}{\|\alpha_1 x_1\|} = \frac{x_1}{\|x_1\|}\frac{\alpha_1}{|\alpha_1|}.
\end{aligned}$$

Hence, we arrive at the following result

$$\lim_{k\to\infty} \epsilon^k u_k = \frac{x_1}{\|x_1\|}\frac{\alpha_1}{|\alpha_1|}, \ \text{where } \epsilon = \frac{|\lambda_1|}{\lambda_1}.$$

□

4.2 Linear functional

Let $\ell\colon \mathbb{R}^n \to \mathbb{R}$ be a linear functional. A more general form of the iteration replaces the two-norm with ℓ, allowing for greater flexibility in normalization strategies:

$$
\begin{aligned}
&\text{For } \ k = 0, 1, 2, \cdots : \\
&\qquad v_{k+1} = A u_k, \\
&\qquad \mu_{k+1} = \ell\left(v_{k+1}\right), \\
&\qquad u_{k+1} = \frac{v_{k+1}}{\mu_{k+1}} \quad \text{with initial } u_0. \\
&\text{End}
\end{aligned}
\tag{4.5}
$$

Theorem 4.2 ([7])**.** *Under the assumption of Theorem 4.1, consider the method defined by* (4.5) *and suppose that* $\ell(v_k) \neq 0$*, for* $k = 1, 2, \ldots,$ *and* $\ell(x_1) \neq 0$*. Then it holds*

$$\lim_{k\to\infty} \mu_k = \lambda_1 \quad \textit{and} \quad \lim_{k\to\infty} u_k = \frac{x_1}{\ell(x_1)}.$$

Proof. From the definition in equation (4.5), it is clear that

$$u_k = \frac{A^k u_0}{\ell\left(A^k u_0\right)},\ \mu_k = \frac{\ell\left(A^k u_0\right)}{\ell\left(A^{k-1} u_0\right)}.$$

By analogy with the above theorem and using the results of equations (4.3) and (4.5), we arrive at a similar equation $\lambda_1^{-k} l\left(A^k u_0\right) \to \alpha_1 \ell\left(x_1\right)$. As $k \to \infty$, the following two equations hold.

$$
\begin{aligned}
\lambda_1^{-k} \ell\left(A^k u_0\right) &\to \alpha_1 \ell\left(x_1\right), \\
\lambda_1^{-k+1} \ell\left(A^{k-1} u_0\right) &\to \alpha_1 \ell\left(x_1\right),
\end{aligned}
$$

Dividing the two equations above, we obtain the following equation:

$$\frac{\lambda_1^{-1} \ell\left(A^k u_0\right)}{\ell\left(A^{k-1} u_0\right)} = \lambda_1^{-1} \mu_k \to 1,$$

which leads to the result

$$\lim_{k\to\infty} \mu_k = \lambda_1.$$

Similarly, using the results from equation (4.3) and taking the limit as $k \to \infty$, we obtain the following equation:

$$
\begin{aligned}
u_k = \frac{A^k u_0}{\ell\left(A^k u_0\right)} &= \frac{\alpha_1 x_1 + \sum_{j=2}^{n} \alpha_j \left(\frac{\lambda_j}{\lambda_1}\right)^k x_j}{\ell\left(\alpha_1 x_1 + \sum_{j=2}^{n} \alpha_j \left(\frac{\lambda_j}{\lambda_1}\right)^k x_j\right)} \\
&\to \frac{\alpha_1 x_1}{\ell\left(\alpha_1 x_1\right)} = \frac{\alpha_1 x_1}{\alpha_1 \ell\left(x_1\right)} = \frac{x_1}{\ell\left(x_1\right)}.
\end{aligned}
$$

Therefore, we deduce that

$$\lim_{k\to\infty} u_k = \frac{x_1}{\ell(x_1)}$$

. □

At the end of this section, we provide a complete description of the power method algorithm. Our goal is to use it as a reference when comparing with the doubling algorithm in later sections, in order to evaluate their differences in experimental results and convergence properties.

Algorithm 1 Power Method for Dominant Eigenvector

Input: A square matrix $A \in \mathbb{R}^{n \times n}$, initial vector $x^{(0)} \neq 0$, tolerance $\varepsilon > 0$, maximum iteration number $N_{\max}$
Output: Approximate dominant eigenvector $x^{(k)}$

Normalize the initial vector: $x \leftarrow x^{(0)}/\|x^{(0)}\|_2$
Set $k \leftarrow 0$
repeat
  $y \leftarrow Ax$
  $x_{\text{new}} \leftarrow y/\|y\|_2$
  $k \leftarrow k + 1$
  if $\|x_{\text{new}} - x\|_2 < \varepsilon$ **then**
    break
  end if
  $x \leftarrow x_{\text{new}}$
until $k \geq N_{\max}$
return x

In this section, we have presented a complete overview of the power method. Its iterative strategy, particularly the use of normalization and the analysis of convergence, provides valuable insights that will influence our approach to the doubling algorithm. In particular, we include the power method in our numerical experiments to compare its performance with that of the doubling algorithm on irreducible and aperiodic stochastic matrices.

Moreover, for matrices that are not irreducible or aperiodic, the standard convergence guarantees may no longer hold. In such cases, incorporating normalization techniques inspired by the power method can help ensure convergence and improve numerical stability. The analysis of convergence order, as used in the power method, also serves as a foundation for establishing theoretical results in our upcoming discussion.

In the next section, we begin the central focus of this thesis—the *Doubling Algorithm (DA)*. We will explore its structure, analyze its behavior, and compare it with traditional approaches such as the power method.

5 Doubling Algorithm for Arbitrary Matrices

In this section, we investigate the application of the structure-preserving doubling algorithm (SDA) to general matrices using a specific partitioning strategy. In particular, we consider an arbitrary matrix paired with the identity matrix to form a matrix pair. The doubling algorithm is then applied by performing structured updates on the matrices M and L. We partition both M and L into four blocks of sizes $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1. Our observation is that the $1\times(n-1)$ block of M tends to converge to the left eigenvector associated with the dominant eigenvalue, while the $(n-1)\times 1$ block of L converges to the corresponding right eigenvector. However, since the remaining blocks of M and L do not converge, an adjustment is required at each iteration to ensure stability and accuracy. This iterative process defines what we refer to as the **Doubling Algorithm (DA)**. Notably, the number of iterations required by this algorithm is typically fewer than that needed by the power method.

To begin with, we recall that the structure-preserving doubling algorithm operates on matrices expressed in a standard symplectic form. In the previous section, we verified that the Hermitian property is preserved, which is a key aspect of the structure-preserving framework. In this section, we introduce an alternative representation referred to as the **standard form**. While it shares similarities with the standard symplectic form, it provides greater flexibility due to adjustable matrix partitioning. This increased degree of freedom enables more adaptable formulations in subsequent analysis. The following discussion will formally define the standard form.

Definition 5.1 (Standard Form)**.** *Let $n\in\mathbb{N}$ and let $p\in\{1,2,\dots,n-1\}$ be an index that specifies a block partitioning of an $n\times n$ matrix pair into four blocks. A matrix pair (M,L) is said to be in* standard form *if it takes the following block structure*

$$\left[\begin{array}{cc|cc} A & 0 & I_{n-p} & C \\ B & I_p & 0 & D \end{array}\right],$$

where $A\in\mathbb{R}^{(n-p)\times(n-p)}$, $B\in\mathbb{R}^{p\times(n-p)}$, $C\in\mathbb{R}^{(n-p)\times p}$, $D\in\mathbb{R}^{p\times p}$.

In this section, we focus on the special case where the partition index is set to $p=1$. This case is particularly important because the associated dominant eigenspace corresponds directly to the left and right eigenvectors.

We begin our analysis by considering an irreducible and aperiodic stochastic matrix and applying the doubling algorithm to it. We examine the convergence behavior of the matrix blocks under iteration and provide a formal proof of convergence in this specific setting.

After establishing the convergence result for the case of an irreducible and aperiodic stochastic matrix, we extend our discussion to more general matrices that are not necessarily stochastic. We analyze the convergence behavior of the doubling algorithm in this broader context and provide a theoretical justification for the observed convergence.

In particular, we examine how the block structure evolves across iterations, gradually eliminating the influence of non-dominant components while isolating the dominant eigenspace in a stable block form. Furthermore, we investigate the asymptotic order of each block during the iteration, which reveals how rapidly different submatrices converge and contributes to a deeper understanding of the numerical stability and efficiency of the algorithm.

To facilitate the analysis, we begin by considering the case in which a matrix is diagonal. This setting allows us to introduce a key linear algebraic tool—namely, the *rank-one decomposition* of diagonalizable matrices. This decomposition enables us to express the matrix as a sum of rank-one terms and plays a central role in proving the main convergence theorem that follows.

Theorem 5.1 ([11, Sec. I.2 p. 10–12])**.** *[Rank-1 Decomposition of Diagonalizable Matrices] Let $A\in\mathbb{C}^{n\times n}$ be a diagonalizable matrix. Suppose there exists an invertible matrix P whose columns $v_1,v_2,\dots,v_n$ are eigenvectors of A corresponding to eigenvalues $\lambda_1,\lambda_2,\dots,\lambda_n$, respectively. Let P^{-1} have rows $w_1^\top,w_2^\top,\dots,w_n^\top$, where each $w_i^\top$ is viewed as a dual vector to v_i. Then A admits the decomposition*

$$A=\sum_{i=1}^{n}\lambda_i v_i w_i^\top, \tag{5.1}$$

where each $v_i w_i^\top$ is a rank-1 matrix.

Proof. Since A is a diagonalizable matrix, there exists an invertible matrix P such that

$$A = PDP^{-1},$$

where $D = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is a diagonal matrix whose diagonal entries are the eigenvalues of A. Writing

$$P = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}, \quad P^{-1} = \begin{bmatrix} w_1^\top \\ w_2^\top \\ \vdots \\ w_n^\top \end{bmatrix},$$

we have

$$\begin{aligned} A &= \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} w_1^\top \\ w_2^\top \\ \vdots \\ w_n^\top \end{bmatrix} \\ &= \begin{bmatrix} \lambda_1 v_1 & \lambda_2 v_2 & \cdots & \lambda_n v_n \end{bmatrix} \begin{bmatrix} w_1^\top \\ w_2^\top \\ \vdots \\ w_n^\top \end{bmatrix} \\ &= \sum_{i=1}^{n} \lambda_i v_i w_i^\top, \end{aligned}$$

which is exactly the decomposition given in equation (5.1). □

In the following discussion, we examine whether the assumptions used in the proof are satisfied. In the first step of the proof, we combine a power of the matrix and the identity matrix to form a pair, and then transform this pair into a standard form. During this process, we need to ensure that the matrix is invertible before proceeding, which imposes certain conditions that must be satisfied. Initially, we might hope that the entire $n \times n$ matrix is invertible. However, it turns out that it is sufficient to verify the invertibility of a smaller matrix instead. In the next theorem, we formally investigate this requirement.

Theorem 5.2. *Let $A \in \mathbb{C}^{n\times n}$ be any matrix. Form a matrix pair by combining A and the identity matrix I_n. Suppose that we partition A and I_n into blocks according to integers $n-p$ and p, so that*

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad I_n = \begin{bmatrix} I_{11} & 0 \\ 0 & I_{22} \end{bmatrix},$$

where $A_{11} \in \mathbb{C}^{(n-p)\times(n-p)}$, $A_{12} \in \mathbb{C}^{(n-p)\times p}$, $A_{21} \in \mathbb{C}^{p\times(n-p)}$, and $A_{22} \in \mathbb{C}^{p\times p}$. Initially, we might want to check the invertibility of the matrix

$$B = \begin{bmatrix} A_{12} & I_{11} \\ A_{22} & 0 \end{bmatrix}.$$

However, since I_{11} is an identity matrix and $I_{21} = 0$, it suffices to verify the invertibility of the submatrix A_{22}. Moreover, if $p = 1$, then A_{22} becomes a scalar, and the condition reduces to requiring that $A_{22} \neq 0$.

Proof. We first observe that the invertibility of the matrix B is equivalent to $\operatorname{rank}(B) = n$. The block I_{11} is a $(n-p)\times(n-p)$ identity matrix, so $\operatorname{rank}(I_{11}) = n - p$. Since $I_{21} = 0$, the contribution to the rank from the lower left block comes entirely from A_{22}. Thus, to achieve $\operatorname{rank}(B) = n$, it is necessary and sufficient that A_{22} contributes the remaining p rank. This means that A_{22} must be invertible. Moreover, when $p = 1$, the block A_{22} becomes a scalar. In this case, requiring A_{22} to be invertible simply reduces to the condition $A_{22} \neq 0$. □

5.1 Doubling Algorithm for Irreducible and Aperiodic Stochastic Matrices

In the next theorem, we start with an irreducible and aperiodic stochastic matrix. It is well known that such a matrix has a simple eigenvalue equal to 1, while all other eigenvalues lie strictly inside the unit circle. This spectral property resembles the behavior observed in the efficient solution of discrete-time algebraic Riccati equations (DAREs). Motivated by this similarity, we are led to consider a doubling-type algorithm in our setting.

Theorem 5.3. *Let $S \in \mathbb{R}^{n\times n}$ be an irreducible and aperiodic stochastic matrix. By pairing S with the identity matrix I_n to form a matrix pair and applying the doubling procedure, we observe that the iterates reveal the dominant eigenvector in a structured block form. Specifically, we partition S and I_n into four blocks according to $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1 submatrices. Then, as $k \to \infty$, the iterated form satisfies*

$$\left[S^k \,\middle|\, I_n\right] \overset{i.e.}{\sim} \left[\begin{array}{cc|cc} M_{11} & 0 & I & L_{12} \\ M_{21} & 1 & 0 & L_{22} \end{array}\right],$$

where the row vector

$$\begin{bmatrix} M_{21} & 1 \end{bmatrix} \tag{5.2}$$

converges to a left eigenvector, and the column vector

$$\begin{bmatrix} L_{12} \\ -1 \end{bmatrix} \tag{5.3}$$

converges to a right eigenvector corresponding to the maximal eigenvalue $\lambda = 1$. Moreover, the block M_{11} converges to the zero matrix as $k \to \infty$.

Proof. According to Theorem 5.1, the stochastic matrix $S \in \mathbb{R}^{n\times n}$ admits the representation given in equation (5.1), where $\lambda_i \in \mathbb{R}$ are the eigenvalues of S, and $v_i, w_i \in \mathbb{R}^n$ are the corresponding right and left eigenvectors, respectively. Then, for each positive integer k, we have

$$S^k = \sum_{i=1}^{n} \lambda_i^k v_i w_i^\top.$$

Since S is an irreducible and aperiodic stochastic matrix, all its eigenvalues satisfy $|\lambda_i| \le 1$, and the dominant eigenvalue $\lambda_1 = 1$ has both algebraic and geometric multiplicity equal to one. Therefore, as $k \to \infty$, all terms associated with eigenvalues $|\lambda_i| < 1$ vanish, and the limit becomes

$$\lim_{k\to\infty} S^k = v_1 w_1^\top,$$

and we now proceed to examine the structure of this limiting matrix. By explicitly writing out the outer product of the dominant right and left eigenvectors, we obtain

$$\begin{aligned}
\lim_{k\to\infty} S^k &= v_1 w_1^\top \\
&= \begin{bmatrix} v_{1,1} \\ v_{2,1} \\ \vdots \\ v_{n-1,1} \\ v_{n,1} \end{bmatrix} \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n-1} & w_{1,n} \end{bmatrix} \\
&= \begin{bmatrix} v_{1,1}w_{1,1} & v_{1,1}w_{1,2} & \cdots & v_{1,1}w_{1,n-1} & v_{1,1}w_{1,n} \\ v_{2,1}w_{1,1} & v_{2,1}w_{1,2} & \cdots & v_{2,1}w_{1,n-1} & v_{2,1}w_{1,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_{n-1,1}w_{1,1} & v_{n-1,1}w_{1,2} & \cdots & v_{n-1,1}w_{1,n-1} & v_{n-1,1}w_{1,n} \\ v_{n,1}w_{1,1} & v_{n,1}w_{1,2} & \cdots & v_{n,1}w_{1,n-1} & v_{n,1}w_{1,n} \end{bmatrix}
\end{aligned}$$

We partition the matrix $S \in \mathbb{R}^{n\times n}$ into four blocks according to the dimensions $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1, namely

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}, \quad \text{where} \begin{cases} S_{11} \in \mathbb{R}^{(n-1)\times(n-1)}, \\ S_{12} \in \mathbb{R}^{(n-1)\times 1}, \\ S_{21} \in \mathbb{R}^{1\times(n-1)}, \\ S_{22} \in \mathbb{R}^{1\times 1}. \end{cases}$$

We then form the matrix pair (S, I) by combining S with the identity matrix I_n. Since S is an irreducible and aperiodic stochastic matrix, there exists an integer $k \in \mathbb{N}$ such that S^k has all positive entries, which in particular implies that the $(2,2)$ block S_{22} contains a strictly positive entry. Therefore, the conditions required to apply Theorem 5.2 are satisfied, and the matrix pair can be transformed into the standard form accordingly. To illustrate this transformation, we apply a sequence of structure-preserving similarity transformations, denoted by $T^{(1)}$ and $T^{(2)}$, to the matrix pair $[S^k \mid I_n]$, where $S^k \to v_1 w_1^\top$ as $k \to \infty$. The resulting form is given below

$$
\left[\begin{array}{cccc|c|cccc|c}
v_{1,1}w_{1,1} & v_{1,1}w_{1,2} & \cdots & v_{1,1}w_{1,n-1} & v_{1,1}w_{1,n} & 1 & 0 & \cdots & 0 & 0 \\
v_{2,1}w_{1,1} & v_{2,1}w_{1,2} & \cdots & v_{2,1}w_{1,n-1} & v_{2,1}w_{1,n} & 0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
v_{n-1,1}w_{1,1} & v_{n-1,1}w_{1,2} & \cdots & v_{n-1,1}w_{1,n-1} & v_{n-1,1}w_{1,n} & 0 & 0 & \cdots & 1 & 0 \\
\hline
v_{n,1}w_{1,1} & v_{n,1}w_{1,2} & \cdots & v_{n,1}w_{1,n-1} & v_{n,1}w_{1,n} & 0 & 0 & \cdots & 0 & 1
\end{array}\right]
$$

$$
\xrightarrow{T^{(1)}}
\left[\begin{array}{cccc|c|cccc|c}
v_{1,1}w_{1,1} & v_{1,1}w_{1,2} & \cdots & v_{1,1}w_{1,n-1} & v_{1,1}w_{1,n} & 1 & 0 & \cdots & 0 & 0 \\
v_{2,1}w_{1,1} & v_{2,1}w_{1,2} & \cdots & v_{2,1}w_{1,n-1} & v_{2,1}w_{1,n} & 0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
v_{n-1,1}w_{1,1} & v_{n-1,1}w_{1,2} & \cdots & v_{n-1,1}w_{1,n-1} & v_{n-1,1}w_{1,n} & 0 & 0 & \cdots & 1 & 0 \\
\hline
\frac{w_{1,1}}{w_{1,n}} & \frac{w_{1,2}}{w_{1,n}} & \cdots & \frac{w_{1,n-1}}{w_{1,n}} & 1 & 0 & 0 & \cdots & 0 & \frac{1}{v_{n,1}w_{1,n}}
\end{array}\right]
$$

$$
\xrightarrow{T^{(2)}}
\left[\begin{array}{cccc|c|cccc|c}
0 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 & -v_{1,1}/v_{n,1} \\
0 & 0 & \cdots & 0 & 0 & 0 & 1 & \cdots & 0 & -v_{2,1}/v_{n,1} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 1 & -v_{n-1,1}/v_{n,1} \\
\hline
\frac{w_{1,1}}{w_{1,n}} & \frac{w_{1,2}}{w_{1,n}} & \cdots & \frac{w_{1,n-1}}{w_{1,n}} & 1 & 0 & 0 & \cdots & 0 & \frac{1}{v_{n,1}w_{1,n}}
\end{array}\right]
$$

$$
= \left[\begin{array}{cc|cc}
M_{11} & 0 & I & L_{12} \\
M_{21} & 1 & 0 & L_{22}
\end{array}\right]
$$

where the transformations $T^{(1)}$ and $T^{(2)}$ are given by

$$
T^{(1)} = \begin{bmatrix} I_{n-1} & 0 \\ 0 & \dfrac{1}{v_{n,1}w_{1,n}} \end{bmatrix}, \qquad
T^{(2)} = \begin{bmatrix}
1 & 0 & \cdots & 0 & -v_{1,1}w_{1,n} \\
0 & 1 & \cdots & 0 & -v_{2,1}w_{1,n} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & -v_{n-1,1}w_{1,n} \\
0 & 0 & \cdots & 0 & 1
\end{bmatrix}.
$$

Hence, we observe that $M_{11} \to 0$, as a consequence of the properties of an irreducible and aperiodic stochastic matrix. We proceed by analyzing the row vector $\begin{bmatrix} M_{21} & 1 \end{bmatrix}$ and the column vector $\begin{bmatrix} L_{12} \\ -1 \end{bmatrix}$, both of which admits the following explicit representations

$$
\begin{aligned}
\begin{bmatrix} M_{21} & 1 \end{bmatrix} &= \begin{bmatrix} \dfrac{w_{1,1}}{w_{1,n}} & \dfrac{w_{1,2}}{w_{1,n}} & \cdots & \dfrac{w_{1,n-1}}{w_{1,n}} & 1 \end{bmatrix} \\
&= \frac{1}{w_{1,n}} \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n-1} & w_{1,n} \end{bmatrix}
\end{aligned}
$$

$$
\begin{bmatrix} L_{12} \\ -1 \end{bmatrix} = \begin{bmatrix} -v_{1,1}/v_{n,1} \\ -v_{2,1}/v_{n,1} \\ \vdots \\ -v_{n-1,1}/v_{n,1} \\ -1 \end{bmatrix} = -\frac{1}{v_{n,1}} \begin{bmatrix} v_{1,1} \\ v_{2,1} \\ \vdots \\ v_{n-1,1} \\ v_{n,1} \end{bmatrix}
$$

Therefore, by the above representation, the row vector in equation (5.2) is identified as a left eigenvector corresponding to the dominant eigenvalue 1, and the column vector in equation (5.3) is the corresponding right eigenvector. This completes the proof. □

Moreover, by Theorem 5.3, we know that the doubling algorithm converges when applied to an irreducible and aperiodic stochastic matrix. Building on this result, our goal is to construct a structure-preserving version of the algorithm that operates more efficiently by focusing on carefully selected submatrices at each iteration. This blockwise update strategy leverages key stochastic properties—such as nonnegativity and row-normalization—to ensure stable convergence toward the dominant eigenvector. The following theorem presents a tailored doubling scheme for this matrix class, whose update mechanism also extends naturally to general matrices by applying the same blockwise iteration process.

Theorem 5.4. *Let $S \in \mathbb{R}^{n\times n}$ be an irreducible and aperiodic stochastic matrix. Consider the matrix pair (S, I_n), where I_n is the identity matrix. Partition both S and I_n into four blocks according to the sizes $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1. Suppose this matrix pair is transformed into the standard form, yielding matrices M and L. Define the updated matrices $\widehat{M}$ and $\widehat{L}$ via the doubling algorithm. Then the resulting iterates take the form*

$$\widehat{M} = \begin{bmatrix} \widehat{M}_{11} & 0 \\ \widehat{M}_{21} & 1 \end{bmatrix}, \quad \widehat{L} = \begin{bmatrix} I & \widehat{L}_{12} \\ 0 & \widehat{L}_{22} \end{bmatrix},$$

where

$$\begin{aligned} \widehat{M}_{11} &= M_{11}(I - L_{12}M_{21})^{-1}M_{11}, \\ \widehat{M}_{21} &= L_{22}M_{21}(I - L_{12}M_{21})^{-1}M_{11} + M_{21}, \\ \widehat{L}_{12} &= L_{12} + M_{11}(I - L_{12}M_{21})^{-1}L_{12}L_{22}, \\ \widehat{L}_{22} &= L_{22}(1 - M_{21}L_{12})^{-1}L_{22}. \end{aligned} \tag{5.4}$$

Proof. To begin with, we consider the matrix pair (S, I_n), where S is a stochastic matrix. Both S and I_n are partitioned into four blocks with sizes $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1, respectively. After applying a structure-preserving transformation, the matrix pair is transformed into the standard form

$$\left[\ S \mid I_n\ \right] \longrightarrow \left[\ M \mid L\ \right] = \left[\begin{array}{cc|cc} M_{11} & 0 & I & L_{12} \\ M_{21} & 1 & 0 & L_{22} \end{array}\right].$$

Next, we consider the inverses of the matrices M and L, which admit convenient factorizations due to their block structure.

$$\begin{aligned} M &= \begin{bmatrix} M_{11} & 0 \\ M_{21} & 1 \end{bmatrix} = \begin{bmatrix} M_{11} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ M_{21} & 1 \end{bmatrix}, \\ \Rightarrow \quad M^{-1} &= \begin{bmatrix} I & 0 \\ -M_{21} & 1 \end{bmatrix}\begin{bmatrix} M_{11}^{-1} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} M_{11}^{-1} & 0 \\ -M_{21}M_{11}^{-1} & 1 \end{bmatrix}, \\ L &= \begin{bmatrix} I & L_{12} \\ 0 & L_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & L_{22} \end{bmatrix}\begin{bmatrix} I & L_{12} \\ 0 & 1 \end{bmatrix}, \\ \Rightarrow \quad L^{-1} &= \begin{bmatrix} I & -L_{12} \\ 0 & 1 \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & L_{22}^{-1} \end{bmatrix} = \begin{bmatrix} I & -L_{12}L_{22}^{-1} \\ 0 & L_{22}^{-1} \end{bmatrix}. \end{aligned}$$

We now consider the pair (L^{-1}, M^{-1}), and apply a sequence of left multiplications that transform it into the standard form. As a result, we obtain the updated matrices M_* and L_* as follows

$$\begin{aligned} &\left[\ L^{-1} \mid M^{-1}\ \right] = \left[\begin{array}{cc|cc} I & -L_{12}L_{22}^{-1} & M_{11}^{-1} & 0 \\ 0 & L_{22}^{-1} & -M_{21}M_{11}^{-1} & 1 \end{array}\right] \\ &\xrightarrow{T^{(1)}} \left[\begin{array}{cc|cc} I & -L_{12}L_{22}^{-1} & M_{11}^{-1} & 0 \\ 0 & 1 & -L_{22}M_{21}M_{11}^{-1} & L_{22} \end{array}\right] \\ &\xrightarrow{T^{(2)}} \left[\begin{array}{cc|cc} I & 0 & (I - L_{12}M_{21})M_{11}^{-1} & L_{12} \\ 0 & 1 & -L_{22}M_{21}M_{11}^{-1} & L_{22} \end{array}\right] \\ &\xrightarrow{T^{(3)}} \left[\begin{array}{cc|cc} M_{11}(I - L_{12}M_{21})^{-1} & 0 & I & M_{11}(I - L_{12}M_{21})^{-1}L_{12} \\ 0 & 1 & -L_{22}M_{21}M_{11}^{-1} & L_{22} \end{array}\right] \\ &\xrightarrow{T^{(4)}} \left[\begin{array}{cc|cc} M_{11}(I - L_{12}M_{21})^{-1} & 0 & I & M_{11}(I - L_{12}M_{21})^{-1}L_{12} \\ L_{22}M_{21}(I - L_{12}M_{21})^{-1} & 1 & 0 & L_{22}M_{21}(I - L_{12}M_{21})^{-1}L_{12} + L_{22} \end{array}\right] \\ &= \left[\ M_* \mid L_*\ \right]. \end{aligned}$$

where

$$T^{(1)} = \begin{bmatrix} I & 0 \\ 0 & L_{22} \end{bmatrix}, \qquad T^{(2)} = \begin{bmatrix} I & L_{12}L_{22}^{-1} \\ 0 & 1 \end{bmatrix},$$

$$T^{(3)} = \begin{bmatrix} M_{11}(I - L_{12}M_{21})^{-1} & 0 \\ 0 & 1 \end{bmatrix}, \quad T^{(4)} = \begin{bmatrix} I & 0 \\ L_{22}M_{21}M_{11}^{-1} & 1 \end{bmatrix}.$$

We then proceed to compute $\widehat{M}$ and $\widehat{L}$.

$$\begin{aligned} \widehat{M} = M_*M &= \begin{bmatrix} M_{11}(I - L_{12}M_{21})^{-1} & 0 \\ L_{22}M_{21}(I - L_{12}M_{21})^{-1} & 1 \end{bmatrix} \begin{bmatrix} M_{11} & 0 \\ M_{21} & 1 \end{bmatrix} \\ &= \begin{bmatrix} M_{11}(I - L_{12}M_{21})^{-1}M_{11} & 0 \\ L_{22}M_{21}(I - L_{12}M_{21})^{-1}M_{11} + M_{21} & 1 \end{bmatrix}, \\ \widehat{L} = L_*L &= \begin{bmatrix} I & M_{11}(I - L_{12}M_{21})^{-1}L_{12} \\ 0 & L_{22}M_{21}(I - L_{12}M_{21})^{-1}L_{12} + L_{22} \end{bmatrix} \begin{bmatrix} I & L_{12} \\ 0 & L_{22} \end{bmatrix} \\ &= \begin{bmatrix} I & L_{12} + M_{11}(I - L_{12}M_{21})^{-1}L_{12}L_{22} \\ 0 & L_{22}(I + M_{21}(I - L_{12}M_{21})^{-1}L_{12})L_{22} \end{bmatrix} \\ &= \begin{bmatrix} I & L_{12} + M_{11}(I - L_{12}M_{21})^{-1}L_{12}L_{22} \\ 0 & L_{22}(1 - M_{21}L_{12})^{-1}L_{22} \end{bmatrix}. \end{aligned}$$

The final equality follows from the Sherman–Morrison–Woodbury identity, as given in equation (2.30) of Theorem 2.10. Comparing the resulting expression with equation (5.4), we obtain the desired result. This completes the proof. □

By Theorem 5.4, the computation involves inverting two key expressions: the matrix $I - L_{12}M_{21}$ and the scalar $1 - M_{21}L_{12}$. To ensure that the algorithm is well-defined at each iteration, it is essential to establish the nonsingularity of these terms. The following theorem provides a sufficient condition under which $I - L_{12}M_{21}$ is nonsingular. This condition not only guarantees the existence of the required inverse but also implies that $1 - M_{21}L_{12}$ is nonzero. In fact, the nonsingularity of these two expressions is mathematically equivalent in the context of the algorithm. Once this condition is satisfied, the doubling algorithm can be constructed and executed successfully.

Theorem 5.5. *Let $I \in \mathbb{R}^{n\times n}$ be the identity matrix, and let $u, v \in \mathbb{R}^n$ be column vectors. Then the following identity holds*

$$\det(I - uv^\top) = 1 - v^\top u. \tag{5.5}$$

Proof. Consider the matrix identity

$$\begin{bmatrix} I & 0 \\ -v^\top & 1 \end{bmatrix} \begin{bmatrix} I - uv^\top & u \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ v^\top & 1 \end{bmatrix} = \begin{bmatrix} I & u \\ 0 & 1 - v^\top u \end{bmatrix}.$$

Taking determinants on both sides, and noting that the outer matrices are invertible with determinant one, we obtain equation (5.5), which completes the proof. □

From Theorem 5.5, we know that the invertibility of the matrix $I - L_{12}M_{21}$ depends on the inner product between the first $n-1$ entries of the left and right eigenvectors. In particular, if this inner product equals one, the matrix becomes singular and the inverse does not exist. However, in our construction, the eigenvectors are not normalized, and thus the inner product is generically not equal to one. In fact, this value is typically far from one, which ensures the invertibility of the matrix in practice. If, in a rare case, the inner product does equal one, one may replace the use of the inverse with a nullspace-based formulation to maintain structure preservation. Nevertheless, such a nullspace approach no longer yields a form suitable for the doubling algorithm. Therefore, we begin by introducing a doubling algorithm that adopts the inverse-based formulation, specifically designed for irreducible and aperiodic stochastic matrices. We now present two algorithms.

The first algorithm transforms a given matrix, combined with the identity matrix to form a matrix pair, into the standard form through a structure-preserving transformation. The second algorithm is the main doubling algorithm, which is specifically tailored for irreducible and aperiodic stochastic matrices. This algorithm fully characterizes the iterative procedure, and each step is constructed directly from the block expressions derived in equation (5.4). In particular, the updates of the subblocks of M and L are explicitly obtained through algebraic manipulations based on the structure-preserving framework developed earlier. According to Theorem 5.4, as the iteration proceeds, the block M_{11} converges to zero. This convergence guarantees that both the left and right eigenvectors associated with the dominant eigenvalue simultaneously converge. Hence, the algorithm not only preserves the underlying structure but also reliably yields the dominant eigenspace.

Algorithm 2 Transform Matrix Pair (S, I_n) into Standard Form

Input: An irreducible and aperiodic stochastic matrix $S \in \mathbb{R}^{n\times n}$
Output: A matrix pair (M, L) in standard form

Partition S into four blocks
$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}, \quad I_n = \begin{bmatrix} I_{n-1} & 0 \\ 0 & 1 \end{bmatrix}$$
Define the first transformation
$$T^{(1)} = \begin{bmatrix} I_{n-1} & 0 \\ 0 & S_{22}^{-1} \end{bmatrix}$$
Define the second transformation
$$T^{(2)} = \begin{bmatrix} I_{n-1} & -S_{12} \\ 0 & 1 \end{bmatrix}$$
Compute
$$[M \mid L] = T^{(2)} T^{(1)} [S \mid I_n]$$

Algorithm 3 Doubling Algorithm for Irreducible and Aperiodic Stochastic Matrices

Input: An irreducible and aperiodic stochastic matrix $S \in \mathbb{R}^{n\times n}$
Output: A left eigenvector and a right eigenvector corresponding to the dominant eigenvalue

Obtain (M, L) from Algorithm 2 (standard form transformation)
Set $k \leftarrow 1$
while $\|M_{11}\| > 10^{-15}$ **do**
    $Q \leftarrow (I - L_{12}M_{21})^{-1}$ # Using the Sherman–Morrison–Woodbury identity
    $M_{11} \leftarrow M_{11} Q M_{11}$
    $M_{21} \leftarrow L_{22} M_{21} Q M_{11} + M_{21}$
    $L_{12} \leftarrow L_{12} + M_{11} Q L_{12} L_{22}$
    $L_{22} \leftarrow L_{22}(1 - M_{21}L_{12})^{-1} L_{22}$
    $k \leftarrow k + 1$
end while
return The dominant left eigenvector from $\begin{bmatrix} M_{21} & 1 \end{bmatrix}$ and right eigenvector from $\begin{bmatrix} L_{12} \\ -1 \end{bmatrix}$

In the first experiment, we compare the performance of the power method and the doubling algorithm for computing the dominant eigenvector of a stochastic matrix. To ensure that the eigenvalues and eigenvectors are real, we generate a random real matrix $R \in \mathbb{R}^{n\times n}$, and construct a symmetric matrix A by setting

$$A = \frac{1}{2}(R + R^\top).$$

Since A is symmetric, all of its eigenvalues are guaranteed to be real. We then normalize each row of A to transform it into an irreducible and aperiodic stochastic matrix. This guarantees a well-defined dominant eigenvalue at 1, with real-valued left and right eigenvectors, and allows us to evaluate both algorithms under a consistent stochastic setting.

In this experiment, we apply Algorithm 1 and Algorithm 3 to evaluate their performance on matrices of varying sizes. For each method, we record the following metrics:

- computation time,
- number of iterations until convergence,
- approximation error of the dominant eigenvector.

The results are presented in the table below.

In this experiment, we observe that the power method generally requires less computation time compared to the doubling algorithm. Although the Sherman–Morrison–Woodbury identity is employed to avoid explicit matrix inversion, the need to perform multiple matrix multiplications across four blocks still increases the overall computational cost.

However, in terms of iteration count, the doubling algorithm converges significantly faster than the power method. This is due to the recursive nature of the doubling scheme, which accelerates convergence through repeated squaring.

Table 1: Comparison of the Power Method and Doubling Algorithm for Computing the Dominant Eigenvector of Irreducible and Aperiodic Stochastic Matrices.

	Power method			**Doubling Algorithm**			
Size	**Time(s)**	**Iter.**	**ev err**	**Time(s)**	**Iter.**	**l.ev err.**	**r.ev err.**
500×500	0.0003	10	2.30e-14	0.0140	4	1.24e-11	1.37e-11
1000×1000	0.0009	9	2.50e-14	0.0978	4	4.31e-13	4.56e-13
3000×3000	0.0088	8	3.97e-14	1.4754	3	2.76e-11	2.58e-11
5000×5000	0.0323	8	2.06e-13	4.3894	3	2.23e-10	3.01e-10
8000×8000	0.0776	8	4.49e-13	16.2023	3	7.98e-12	7.69e-12
10000×10000	0.1243	8	4.15e-13	46.3557	3	2.87e-10	2.76e-10

Regarding accuracy, the doubling algorithm does not necessarily yield more precise results in terms of numerical error. Nevertheless, unlike the power method—which only approximates the dominant right eigenvector—the doubling algorithm simultaneously approximates both the left and right eigenvectors.

In summary, although the doubling algorithm incurs a higher computational cost per iteration, it achieves convergence in significantly fewer iterations, making it a more robust and efficient method for computing the dominant eigenspace of stochastic matrices.

Beyond performance comparison, this experiment also serves to validate the theoretical results established earlier. In particular, the observed convergence behavior of the matrix blocks—especially the simultaneous approximation of both eigenvectors—aligns well with the convergence guarantees stated in Theorem 5.3, thus providing empirical confirmation of the theoretical analysis.

5.2 Extension to General Matrices and Convergence Adjustment

In the following subsection, we extend our investigation from irreducible and aperiodic stochastic matrices to more general matrices. Our objective is to examine whether the dominant eigenvector can still be extracted through the doubling process under these broader conditions.

However, we observe that convergence is no longer guaranteed in all matrix blocks. In particular, the $(1,1)$-block of M and the $(2,2)$-block of L may fail to converge. Despite this, the computation of the dominant eigenspace remains valid, since the left and right eigenvectors are associated with the $(2,1)$-block of M and the $(1,2)$-block of L, respectively.

Nevertheless, to gain a deeper understanding of the algorithm's behavior, we aim to analyze how and why the divergent blocks evolve. In particular, we analyze the convergence order of the matrix blocks M_{11} and L_{22} to better understand their asymptotic behavior under the doubling process. Rather than simply observing whether these blocks converge, we perform a block-wise quantitative analysis by tracking how their norms evolve across iterations. This allows us to determine how fast or slow these blocks decay, or whether they diverge.

Such analysis is crucial, especially in the context of general matrices where convergence is not guaranteed. Even though the dominant eigenvectors are extracted from the $(2,1)$-block of M and the $(1,2)$-block of L, observing the behavior of M_{11} and L_{22} still plays an important role in understanding the overall numerical stability of the algorithm.

Furthermore, monitoring the evolution of these blocks provides valuable practical insights. For example, identifying slow decay or divergence in M_{11} or L_{22} can serve as a diagnostic tool for adapting the algorithm dynamically. This could involve introducing regularization, adjusting iteration strategies, or switching to a stabilized scheme when instability is detected. Therefore, the convergence order of these blocks is not only of theoretical interest but also contributes directly to the design of more robust and adaptive variants of the doubling algorithm.

In the following theorem, we extend the doubling algorithm to arbitrary matrices, without requiring the matrix to be stochastic. Compared to Theorem 5.3, the conclusion is weaker in that it no longer includes the convergence of the $(1,1)$-block M_{11} to zero.

Instead, this result focuses on the convergence of the $(2,1)$-block of M and the $(1,2)$-block of L to the left and right eigenvectors associated with the dominant eigenvalue. These two properties are sufficient for recovering the dominant eigenspace, even though full convergence of all matrix blocks is not guaranteed.

Theorem 5.6. *Let $A \in \mathbb{R}^{n\times n}$ be an arbitrary matrix such that the (n,n)-entry of A^k is nonzero for some $k \in \mathbb{N}$. By pairing A with the identity matrix I_n to form a matrix pair and applying the doubling procedure, the iterates converge to the dominant eigenvector associated with the dominant eigenvalue. Specifically, we partition both A and I_n into four blocks according to submatrix sizes $(n-1) \times (n-1)$, $(n-1) \times 1$, $1 \times (n-1)$, and 1×1. Then,* *as $k \to \infty$, the iterated form satisfies*

$$\left[A^k \,\middle|\, I_n\right] \overset{\text{l.e.}}{\sim} \left[\begin{array}{cc|cc} M_{11} & 0 & I & L_{12} \\ M_{21} & 1 & 0 & L_{22} \end{array}\right],$$

where the vectors in equations (5.2) *and* (5.3) *converge to the left and right eigenvectors corresponding to the dominant eigenvalue, respectively.*

Proof. The argument is similar to that of Theorem 5.3, except that the irreducible and aperiodic stochastic assumption is not required. We demonstrate the proof as follows. According to Theorem 5.1, any matrix $A \in \mathbb{R}^{n\times n}$ that is diagonalizable admits the spectral representation given in equation (5.1), where $\lambda_i \in \mathbb{R}$ are the eigenvalues of A, and $v_i, w_i \in \mathbb{R}^n$ are the corresponding right and left eigenvectors, respectively. Then, for each positive integer k, we have

$$\begin{aligned}
A^k &= \sum_{i=1}^{n} \lambda_i^k v_i w_i^\top = \sum_{i=1}^{n} \lambda_i^k \begin{bmatrix} v_{1,i} \\ v_{2,i} \\ \vdots \\ v_{n,i} \end{bmatrix} \begin{bmatrix} w_{i,1} & w_{i,2} & \cdots & w_{i,n} \end{bmatrix} \\
&= \sum_{i=1}^{n} \lambda_i^k \begin{bmatrix} v_{1,i}w_{i,1} & v_{1,i}w_{i,2} & \cdots & v_{1,i}w_{i,n-1} & v_{1,i}w_{i,n} \\ v_{2,i}w_{i,1} & v_{2,i}w_{i,2} & \cdots & v_{2,i}w_{i,n-1} & v_{2,i}w_{i,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ v_{n-1,i}w_{i,1} & v_{n-1,i}w_{i,2} & \cdots & v_{n-1,i}w_{i,n-1} & v_{n-1,i}w_{i,n} \\ v_{n,i}w_{i,1} & v_{n,i}w_{i,2} & \cdots & v_{n,i}w_{i,n-1} & v_{n,i}w_{i,n} \end{bmatrix} \\
&= \sum_{i=1}^{n} \begin{bmatrix} \lambda_i^k v_{1,i}w_{i,1} & \lambda_i^k v_{1,i}w_{i,2} & \cdots & \lambda_i^k v_{1,i}w_{i,n-1} & \lambda_i^k v_{1,i}w_{i,n} \\ \lambda_i^k v_{2,i}w_{i,1} & \lambda_i^k v_{2,i}w_{i,2} & \cdots & \lambda_i^k v_{2,i}w_{i,n-1} & \lambda_i^k v_{2,i}w_{i,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \lambda_i^k v_{n-1,i}w_{i,1} & \lambda_i^k v_{n-1,i}w_{i,2} & \cdots & \lambda_i^k v_{n-1,i}w_{i,n-1} & \lambda_i^k v_{n-1,i}w_{i,n} \\ \lambda_i^k v_{n,i}w_{i,1} & \lambda_i^k v_{n,i}w_{i,2} & \cdots & \lambda_i^k v_{n,i}w_{i,n-1} & \lambda_i^k v_{n,i}w_{i,n} \end{bmatrix}
\end{aligned}$$

We partition the matrix $A \in \mathbb{R}^{n\times n}$ into four blocks according to the dimensions $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1, namely

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad \text{where} \begin{cases} A_{11} \in \mathbb{R}^{(n-1)\times(n-1)}, \\ A_{12} \in \mathbb{R}^{(n-1)\times 1}, \\ A_{21} \in \mathbb{R}^{1\times(n-1)}, \\ A_{22} \in \mathbb{R}^{1\times 1}. \end{cases}$$

We then form the matrix pair (A, I) by combining A with the identity matrix I_n. Since the (n,n)-entry of A^k is nonzero for some $k \in \mathbb{N}$, this ensures that the $(2,2)$-block A_{22} contains a strictly positive entry. Therefore, the conditions required to apply Theorem 5.2 are satisfied, and the matrix pair can be transformed into the standard form accordingly. To illustrate this transformation, we apply a sequence of structure-preserving similarity transformations, denoted by $T^{(1)}$ and $T^{(2)}$, to the matrix pair $[A^k \mid I_n]$. The resulting form is given below.

$$
\begin{aligned}
&\sum_{i=1}^{n}\left[\begin{array}{cccc|c|cccc|c}
\lambda_i^k v_{1,i}w_{i,1} & \lambda_i^k v_{1,i}w_{i,2} & \cdots & \lambda_i^k v_{1,i}w_{i,n-1} & \lambda_i^k v_{1,i}w_{i,n} & \frac{1}{n} & 0 & \cdots & 0 & 0\\
\lambda_i^k v_{2,i}w_{i,1} & \lambda_i^k v_{2,i}w_{i,2} & \cdots & \lambda_i^k v_{2,i}w_{i,n-1} & \lambda_i^k v_{2,i}w_{i,n} & 0 & \frac{1}{n} & \cdots & 0 & 0\\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
\lambda_i^k v_{n-1,i}w_{i,1} & \lambda_i^k v_{n-1,i}w_{i,2} & \cdots & \lambda_i^k v_{n-1,i}w_{i,n-1} & \lambda_i^k v_{n-1,i}w_{i,n} & 0 & 0 & \cdots & \frac{1}{n} & 0\\
\hline
\lambda_i^k v_{n,i}w_{i,1} & \lambda_i^k v_{n,i}w_{i,2} & \cdots & \lambda_i^k v_{n,i}w_{i,n-1} & \lambda_i^k v_{n,i}w_{i,n} & 0 & 0 & \cdots & 0 & \frac{1}{n}
\end{array}\right]\\
&\xrightarrow{T^{(1)}}\sum_{i=1}^{n}\left[\begin{array}{cccc|c|cccc|c}
\lambda_i^k v_{1,i}w_{i,1} & \lambda_i^k v_{1,i}w_{i,2} & \cdots & \lambda_i^k v_{1,i}w_{i,n-1} & \lambda_i^k v_{1,i}w_{i,n} & \frac{1}{n} & 0 & \cdots & 0 & 0\\
\lambda_i^k v_{2,i}w_{i,1} & \lambda_i^k v_{2,i}w_{i,2} & \cdots & \lambda_i^k v_{2,i}w_{i,n-1} & \lambda_i^k v_{2,i}w_{i,n} & 0 & \frac{1}{n} & \cdots & 0 & 0\\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
\lambda_i^k v_{n-1,i}w_{i,1} & \lambda_i^k v_{n-1,i}w_{i,2} & \cdots & \lambda_i^k v_{n-1,i}w_{i,n-1} & \lambda_i^k v_{n-1,i}w_{i,n} & 0 & 0 & \cdots & \frac{1}{n} & 0\\
\hline
\frac{\lambda_i^k v_{n,i}w_{i,1}}{m_{n,n}} & \frac{\lambda_i^k v_{n,i}w_{i,2}}{m_{n,n}} & \cdots & \frac{\lambda_i^k v_{n,i}w_{i,n-1}}{m_{n,n}} & \frac{\lambda_i^k v_{n,i}w_{i,n}}{m_{n,n}} & 0 & 0 & \cdots & 0 & \frac{1}{n\cdot m_{n,n}}
\end{array}\right]\\
&\xrightarrow{T^{(2)}}\sum_{i=1}^{n}\left[\begin{array}{cccc|c|cccc|c}
m_{1,1} & m_{1,2} & \cdots & m_{1,n-1} & 0 & \frac{1}{n} & 0 & \cdots & 0 & -\frac{\lambda_i^k v_{1,i}w_{i,n}}{m_{n,n}}\\
m_{2,1} & m_{2,2} & \cdots & m_{2,n-1} & 0 & 0 & \frac{1}{n} & \cdots & 0 & -\frac{\lambda_i^k v_{2,i}w_{i,n}}{m_{n,n}}\\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
m_{n-1,1} & m_{n-1,2} & \cdots & m_{n-1,n-1} & 0 & 0 & 0 & \cdots & \frac{1}{n} & -\frac{\lambda_i^k v_{n-1,i}w_{i,n}}{m_{n,n}}\\
\hline
\frac{\lambda_i^k v_{n,i}w_{i,1}}{m_{n,n}} & \frac{\lambda_i^k v_{n,i}w_{i,2}}{m_{n,n}} & \cdots & \frac{\lambda_i^k v_{n,i}w_{i,n-1}}{m_{n,n}} & \frac{\lambda_i^k v_{n,i}w_{i,n}}{m_{n,n}} & 0 & 0 & \cdots & 0 & \frac{1}{n\cdot m_{n,n}}
\end{array}\right] \qquad (5.6)\\
&= \left[\begin{array}{cc|cc}
M_{11} & 0 & I & L_{12}\\
M_{21} & 1 & 0 & L_{22}
\end{array}\right]
\end{aligned}
$$

where the transformations $T^{(1)}$ and $T^{(2)}$ are given by

$$
T^{(1)} = \begin{bmatrix} I_{n-1} & 0\\ 0 & \frac{1}{m_{n,n}} \end{bmatrix} = \begin{bmatrix} I_{n-1} & 0\\ 0 & \frac{1}{\sum\limits_{i=1}^{n}\lambda_i^k v_{n,i}w_{i,n}} \end{bmatrix}, \quad
T^{(2)} = \begin{bmatrix}
1 & 0 & \cdots & 0 & -\sum\limits_{i=1}^{n}\lambda_i^k v_{1,i}w_{i,n}\\
0 & 1 & \cdots & 0 & -\sum\limits_{i=1}^{n}\lambda_i^k v_{2,i}w_{i,n}\\
\vdots & \vdots & \ddots & \vdots & \vdots\\
0 & 0 & \cdots & 1 & -\sum\limits_{i=1}^{n}\lambda_i^k v_{n-1,i}w_{i,n}\\
0 & 0 & \cdots & 0 & 1
\end{bmatrix}.
$$

We now consider the $(2,1)$-block of M and the $(1,2)$-block of L. We begin with the $(2,1)$-block of M, and without loss of generality, focus on the $(n,1)$-entry. The following identity holds

$$
\begin{aligned}
\frac{\sum\limits_{i=1}^{n}\lambda_i^k v_{n,i}w_{i,1}}{m_{n,n}} &= \frac{\sum\limits_{i=1}^{n}\lambda_i^k v_{n,i}w_{i,1}}{\sum\limits_{i=1}^{n}\lambda_i^k v_{n,i}w_{i,n}} = \frac{\sum\limits_{i=1}^{n}\left(\frac{\lambda_i}{\lambda_1}\right)^k v_{n,i}w_{i,1}}{\sum\limits_{i=1}^{n}\left(\frac{\lambda_i}{\lambda_1}\right)^k v_{n,i}w_{i,n}}\\
&\longrightarrow \frac{v_{n,1}w_{1,1}}{v_{n,1}w_{1,n}} = \frac{w_{1,1}}{w_{1,n}}, \quad \text{as } k\to\infty.
\end{aligned}
$$

From the limiting structure of the matrix M, we observe that the last row, given by $\begin{bmatrix} M_{21} & 1 \end{bmatrix}$, corresponds to a left eigenvector associated with the dominant eigenvalue. In particular, we have

$$\begin{aligned} \begin{bmatrix} M_{21} & 1 \end{bmatrix} &= \begin{bmatrix} \frac{w_{1,1}}{w_{1,n}} & \frac{w_{1,2}}{w_{1,n}} & \cdots & \frac{w_{1,n-1}}{w_{1,n}} & 1 \end{bmatrix} \\ &= \frac{1}{w_{1,n}} \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n-1} & w_{1,n} \end{bmatrix}, \end{aligned}$$

which is a scaled version of the left eigenvector $w_1^\top$.

In a similar manner, we now consider the $(1,2)$-block of L, and without loss of generality, focus on the $(1,n)$-entry. The following identity holds

$$\begin{aligned} -\frac{\sum_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,n}}{m_{n,n}} = -\frac{\sum_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,n}}{\sum_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,n}} &= -\frac{\sum_{i=1}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^k v_{1,i} w_{i,n}}{\sum_{i=1}^{n} \left(\frac{\lambda_i}{\lambda_1}\right)^k v_{n,i} w_{i,n}} \\ &\longrightarrow -\frac{v_{1,1} w_{1,n}}{v_{n,1} w_{1,n}} = -\frac{v_{1,1}}{v_{n,1}}, \quad \text{as } k \to \infty. \end{aligned}$$

From the limiting structure of the matrix L, we observe that the last column, given by $\begin{bmatrix} L_{12} \\ -1 \end{bmatrix}$, corresponds to a right eigenvector associated with the dominant eigenvalue. In particular, we have

$$\begin{bmatrix} L_{12} \\ -1 \end{bmatrix} = \begin{bmatrix} -\frac{v_{1,1}}{v_{n,1}} \\ -\frac{v_{2,1}}{v_{n,1}} \\ \vdots \\ -\frac{v_{n-1,1}}{v_{n,1}} \\ -1 \end{bmatrix} = -\frac{1}{v_{n,1}} \begin{bmatrix} v_{1,1} \\ v_{2,1} \\ \vdots \\ v_{n-1,1} \\ v_{n,1} \end{bmatrix},$$

which is a scaled version of the right eigenvector v_1. Consequently, the vectors given in equations (5.2) and (5.3) respectively converge to the left and right eigenvectors associated with the dominant eigenvalue. This completes the proof. $\square$

In this theorem, it is clearly shown that the $(2,1)$-block of M and the $(1,2)$-block of L converge to the left and right eigenvectors associated with the dominant eigenvalue, respectively. Notably, this result does not require the $(1,1)$-block of M or the $(2,2)$-block of L to converge. However, from a numerical perspective, divergence of these matrix blocks is undesirable, as it may affect the overall stability of the algorithm. Therefore, in the following discussion, we investigate the convergence behavior of the $(1,1)$-block of M and the $(2,2)$-block of L in greater detail. In particular, we analyze their convergence order using the representation given in equation (5.6). This analysis provides deeper insight into the growth or decay rates of the divergent components, and serves as the foundation for the next theorem.

Theorem 5.7. *Let $A \in \mathbb{R}^{n\times n}$ be a diagonalizable matrix whose eigenvalues satisfy $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. Then, as $k \to \infty$, the matrix pair $[A^k \mid I_n]$ asymptotically satisfies*

$$\left[A^k \,\middle|\, I_n \right] \sim \left[\begin{array}{cc|cc} \mathcal{O}(|\lambda_2|^k) & 0 & I & \mathcal{O}(1) \\ \mathcal{O}(1) & 1 & 0 & \mathcal{O}(|\lambda_1|^{-k}) \end{array} \right],$$

under a block partitioning conforming to $(n-1)\times(n-1)$, $(n-1)\times 1$, $1\times(n-1)$, and 1×1 submatrices.

Proof. By equation (5.6), we obtain explicit expressions for the $(1,1)$-block of M and the $(2,2)$-block of L. We begin by analyzing the $(1,1)$-block of M. Since this block is of size $(n-1)\times(n-1)$, we may, without loss of generality, focus on the $(1,1)$-entry of M. This leads to the following identity

$$
\begin{aligned}
|m_{1,1}| &= \left| \sum_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,1} - \sum_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,n} \cdot \frac{\sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,1}}{m_{n,n}} \right| \\
&= \frac{\left| \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,1} \right) m_{n,n} - \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,n} \right) \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,1} \right) \right|}{|m_{n,n}|} \\
&= \frac{\left| \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,1} \right) \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,n} \right) - \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{1,i} w_{i,n} \right) \left(\sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,1} \right) \right|}{\left| \sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,n} \right|} \\
&= \frac{\left| \sum\limits_{i \neq j} \lambda_i^k \lambda_j^k \left(v_{1,i} w_{i,1} v_{n,j} w_{j,n} - v_{1,i} w_{i,n} v_{n,j} w_{j,1} \right) \right|}{\left| \sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,n} \right|} \\
&\leq \frac{c_1 \cdot |\lambda_1|^k \, |\lambda_2|^k}{c_2 \cdot |\lambda_1|^k} = \mathcal{O}(|\lambda_2|^k), \quad \text{as } k \to \infty,
\end{aligned}
$$

where $c_1, c_2 > 0$ are constants depending on the eigenvectors. The fourth equality follows from the fact that all terms with $i = j$ cancel out, since

$$
v_{1,i} w_{i,1} v_{n,i} w_{i,n} - v_{1,i} w_{i,n} v_{n,i} w_{i,1} = 0.
$$

Thus, only the cross terms with $i \neq j$ remain. The inequality in the final line holds because for any $i \neq j$, we have $|\lambda_i \lambda_j| \leq |\lambda_1 \lambda_2|$, and therefore $|\lambda_i^k \lambda_j^k| \leq |\lambda_1|^k |\lambda_2|^k$. In contrast, the denominator is asymptotically dominated by the term involving $|\lambda_1|^k$, and hence remains bounded away from zero as $k \to \infty$. As a result, the entire expression decays at a rate of $\mathcal{O}(|\lambda_2|^k)$. Each entry of the $(1,1)$-block M_{11} decays at a rate of $\mathcal{O}(|\lambda_2|^k)$, as can be shown using similar reasoning applied to other positions in the block. Next, we turn our attention to the $(2,2)$-block of L, which is a scalar entry. In this case, the asymptotic behavior can be computed directly.

$$
\begin{aligned}
\left| \frac{1}{m_{n,n}} \right| &= \left| \frac{1}{\sum\limits_{i=1}^{n} \lambda_i^k v_{n,i} w_{i,n}} \right| \\
&= \frac{1}{\left| \lambda_1^k \right| \left| \sum\limits_{i=1}^{n} \left(\dfrac{\lambda_i}{\lambda_1} \right)^k v_{n,i} w_{i,n} \right|} \\
&\leq \frac{1}{c \cdot |\lambda_1|^k} = \mathcal{O}(|\lambda_1|^{-k}), \quad \text{as } k \to \infty,
\end{aligned}
$$

In the final inequality, we observe that the dominant contribution to the sum in the denominator comes from the term associated with λ_1. Since all other terms involve $|\lambda_i / \lambda_1|^k$ with $|\lambda_i| < |\lambda_1|$, they decay exponentially as $k \to \infty$. The leading term corresponding to $i = 1$ remains bounded away from zero, so the entire summation is bounded below by a constant $c > 0$, which ensures the inequality holds. Moreover, the $(2,1)$-block of M and the $(1,2)$-block of L converge to the left and right eigenvectors, respectively, with convergence order $\mathcal{O}(1)$. Hence, the proof is complete. $\square$

In the following discussion, based on Theorem 5.7, we observe that the $(1,1)$-block of M and the $(2,2)$-block of L may fail to converge. This behavior poses a challenge in numerical computation, as unbounded growth in these blocks can compromise stability and accuracy. To address this issue, we propose an adjustment strategy inspired by the power method. At each iteration, we measure the maximum absolute entry in the $(1,1)$-block of M or the $(2,2)$-block of L, and apply rescaling to control their growth. This normalization step ensures that these blocks remain numerically well-conditioned throughout the iteration process.

Before presenting the main doubling algorithm, we introduce a preprocessing procedure that transforms a given matrix pair into the standard form. This is achieved by extracting the relevant block structure, aligning identity submatrices, and constructing a permutation matrix to reorder the rows and columns. The transformation is then performed via the solution of linear systems. The resulting matrix pair satisfies the standard form required for applying the doubling algorithm. We summarize this procedure and incorporate the adjustment strategy into the iterative process to form the complete **doubling algorithm**, as described in the following steps.

Algorithm 4 Transform Matrix Pair (A, B) into Standard Form

Input: Matrices $A, B \in \mathbb{R}^{n\times n}$
Output: A matrix pair (M, L) in standard form
Form matrix $C \leftarrow [A(:, n) \mid B(:, 1{:}n{-}1)]$
Construct permutation matrix

$$K \leftarrow \begin{bmatrix} \mathbf{0}_{(n-1)\times 1} & I_{n-1} \\ 1 & \mathbf{0}_{1\times(n-1)} \end{bmatrix}$$

Solve linear systems $X_A \leftarrow C\backslash A, \quad X_B \leftarrow C\backslash B$
Compute $M \leftarrow K \cdot X_A$
Compute $L \leftarrow K \cdot X_B$
return M, L

Algorithm 5 Normalized Doubling Algorithm with Tracking of Submatrices and Eigenvectors

Input: A matrix $A \in \mathbb{R}^{n\times n}$, maximum iteration count `MaxStep`
Output: Norm sequences $\|M_{11}^{(t)}\|$, $\|L_{22}^{(t)}\|$; eigenvector sequences $v^{(t)}$, $w^{(t)\top}$; final eigenvectors
Obtain (M, L) by applying the standard form transformation to (A, I_n) using Algorithm 4
Initialize sequences:

$$\|M_{11}^{(1)}\| \leftarrow \|M(1{:}n{-}1,\ 1{:}n{-}1)\|, \qquad \|L_{22}^{(1)}\| \leftarrow |L(n, n)|$$
$$v^{(1)} \leftarrow \begin{bmatrix} L(1{:}n{-}1,\ n) \\ -1 \end{bmatrix}, \qquad w^{(1)\top} \leftarrow \begin{bmatrix} M(n,\ 1{:}n{-}1) & 1 \end{bmatrix}$$

for $t = 2$ to `MaxStep` **do**
  $\ell \leftarrow \|L(n,\ n)\|_\infty$
  Normalize by the infinity norm: $L \leftarrow L/\ell$
  Compute null space $\texttt{tmp} \leftarrow \texttt{null}([L^\top,\ -M^\top])$
  Obtain $L_1 \leftarrow (\texttt{tmp}(1{:}n, :))^\top, \quad M_1 \leftarrow (\texttt{tmp}(n{+}1{:}2n, :))^\top$
  Update $(M^{(t)}, L^{(t)})$ by transforming $(L_1 M, M_1 L)$ via Algorithm 4
  Record

$$\|M_{11}^{(t)}\| \leftarrow \|M^{(t)}(1{:}n{-}1,\ 1{:}n{-}1)\|, \qquad \|L_{22}^{(t)}\| \leftarrow |L^{(t)}(n, n)|$$
$$v^{(t)} \leftarrow \begin{bmatrix} L^{(t)}(1{:}n{-}1,\ n) \\ -1 \end{bmatrix}, \qquad w^{(t)\top} \leftarrow \begin{bmatrix} M^{(t)}(n,\ 1{:}n{-}1) & 1 \end{bmatrix}$$

end for
Set final eigenvectors

$$v \leftarrow v^{(\texttt{MaxStep})}, \quad w^\top \leftarrow w^{(\texttt{MaxStep})\top}$$

return sequences $\{\|M_{11}^{(t)}\|\}$, $\{\|L_{22}^{(t)}\|\}$, $\{v^{(t)}\}$, $\{w^{(t)\top}\}$, and final v, $w^\top$

In the second experiment, we collect the sequences $\|M_{11}^{(t)}\|$ and $\|L_{22}^{(t)}\|$, the eigenvector sequences $v^{(t)}$ and $w^{(t)\top}$, and the final vectors v and $w^\top$, which approximate the left and right eigenvectors corresponding to the dominant eigenvalue. The goal of this experiment is to verify Theorem 5.6, which asserts that

$$w^\top = \begin{bmatrix} M_{21} & 1 \end{bmatrix}, \quad v = \begin{bmatrix} L_{12} \\ -1 \end{bmatrix}$$

converge to the left and right eigenvectors, respectively. We therefore examine whether the errors between the approximated and true eigenvectors decrease as the number of iterations increases. Moreover, according to Theorem 5.7, the submatrices M_{11} and L_{22} may not converge numerically. We investigate whether appropriate adjustments can lead to their convergence. By observing the sequences $\|M_{11}^{(t)}\|$ and $\|L_{22}^{(t)}\|$, we aim to determine whether their magnitudes tend to stabilize or at least do not diverge as iterations progress. In particular, we also monitor the convergence trend by checking whether the difference between sequential terms, such as $\|M_{11}^{(t+1)} - M_{11}^{(t)}\|$, decreases over time.

In this experiment, we follow a procedure similar to that of the first experiment. To ensure that the eigenvalues and eigenvectors are real, we generate a real matrix $R \in \mathbb{R}^{n\times n}$, and construct a symmetric positive semidefinite matrix A by setting

$$A = R^\top R.$$

Since A is symmetric, all of its eigenvalues and eigenvectors are guaranteed to be real. Moreover, because A is positive semidefinite, all of its eigenvalues are nonnegative. We then apply Algorithm 5 to the matrix A, and evaluate the following aspects:

- whether the errors between the computed left and right eigenvectors and the true eigenvectors decrease and converge to zero as the iteration proceeds,
- whether the norms of the submatrices M_{11} and L_{22} converge under the normalization scheme,
- whether the variation in submatrix norms between sequential iterations gradually decreases.

In this experiment, we fix the maximum number of iterations to 20 in order to observe the convergence behavior within a predefined computational cost. To better illustrate the results, we use plots to visualize the variation of the eigenvectors and the norms of the submatrices throughout the iterations.

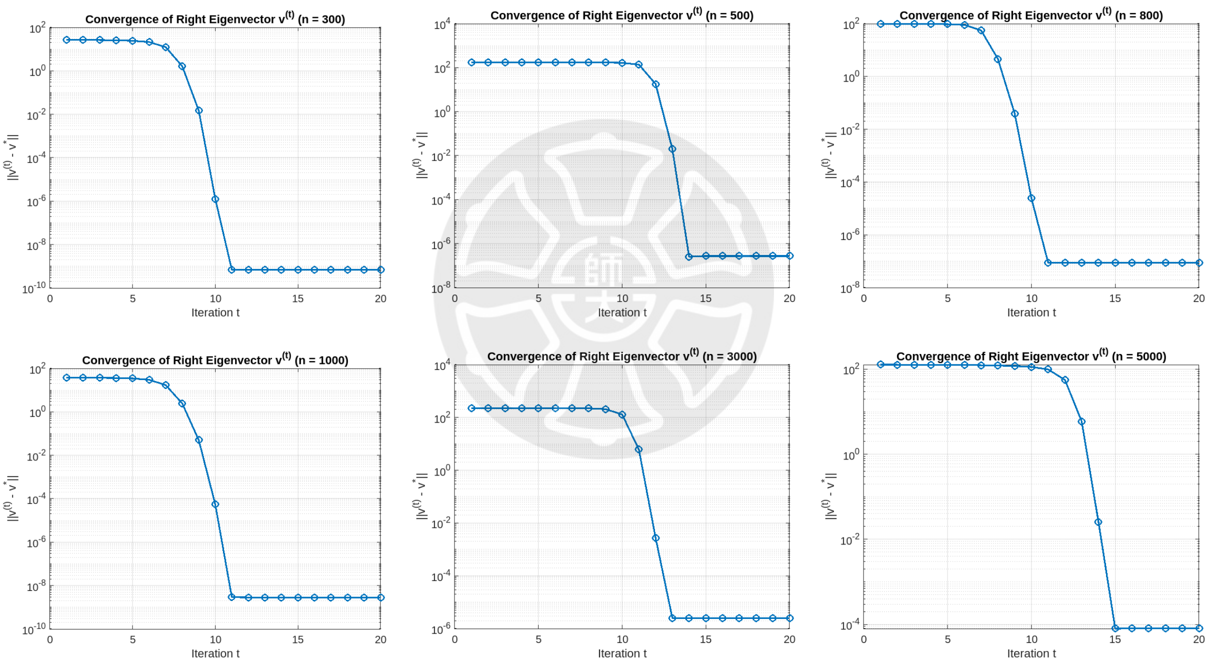

Figure 1: Convergence of the right eigenvector approximation $v^{(t)}$ relative to the true eigenvector for different matrix sizes

In Figure 1, we compare the true right eigenvector with the sequence of approximated right eigenvectors $v^{(t)}$. It can be observed that the sequence eventually converges to the true eigenvector, although the convergence is not monotonic at the beginning. In most cases, the approximation remains relatively unstable during the initial iterations and only starts to decrease significantly after around ten steps. Nevertheless, the overall trend confirms that the computed eigenvectors converge to the true right eigenvector.

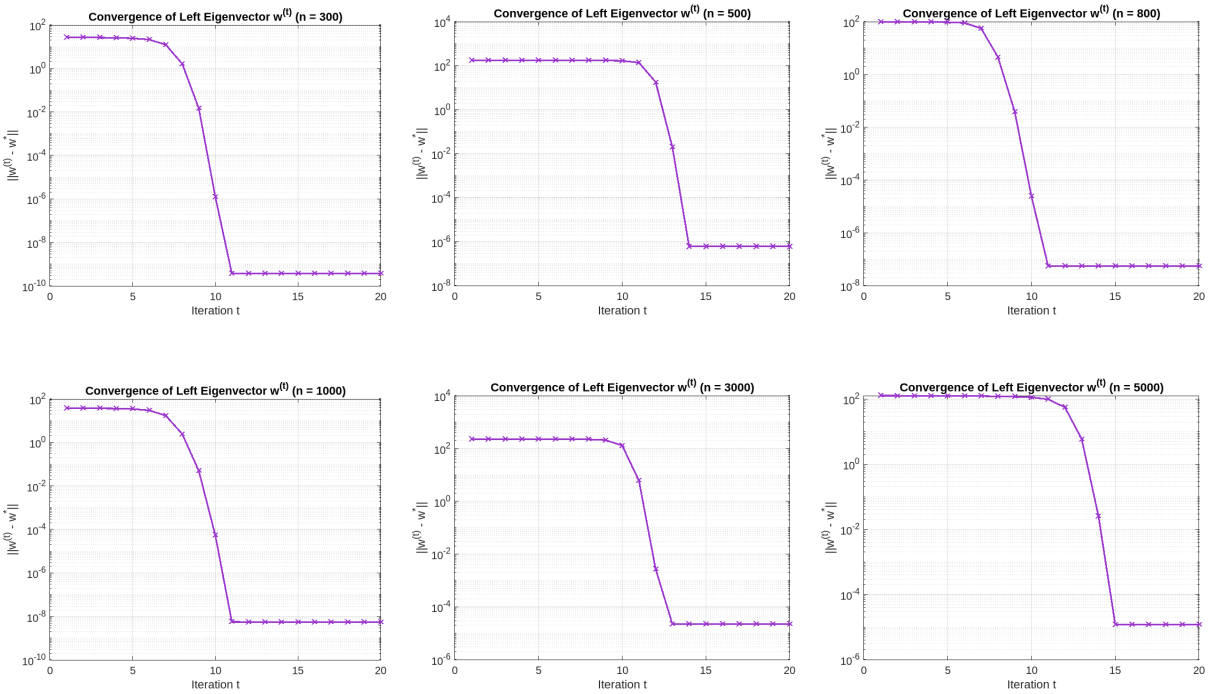

Figure 2: Convergence of the left eigenvector approximation $w^{(t)}$ relative to the true eigenvector for different matrix sizes

Figure 2 illustrates the convergence behavior of the approximated left eigenvectors $w^{(t)}$ toward the true left eigenvector. While the approximations ultimately converge, the convergence process is not smooth in the early stages. In the early iterations, the error changes irregularly and only begins to decrease steadily after around ten steps. Despite this initial instability, the results indicate that the sequence of approximations reliably converges to the correct left eigenvector over time.

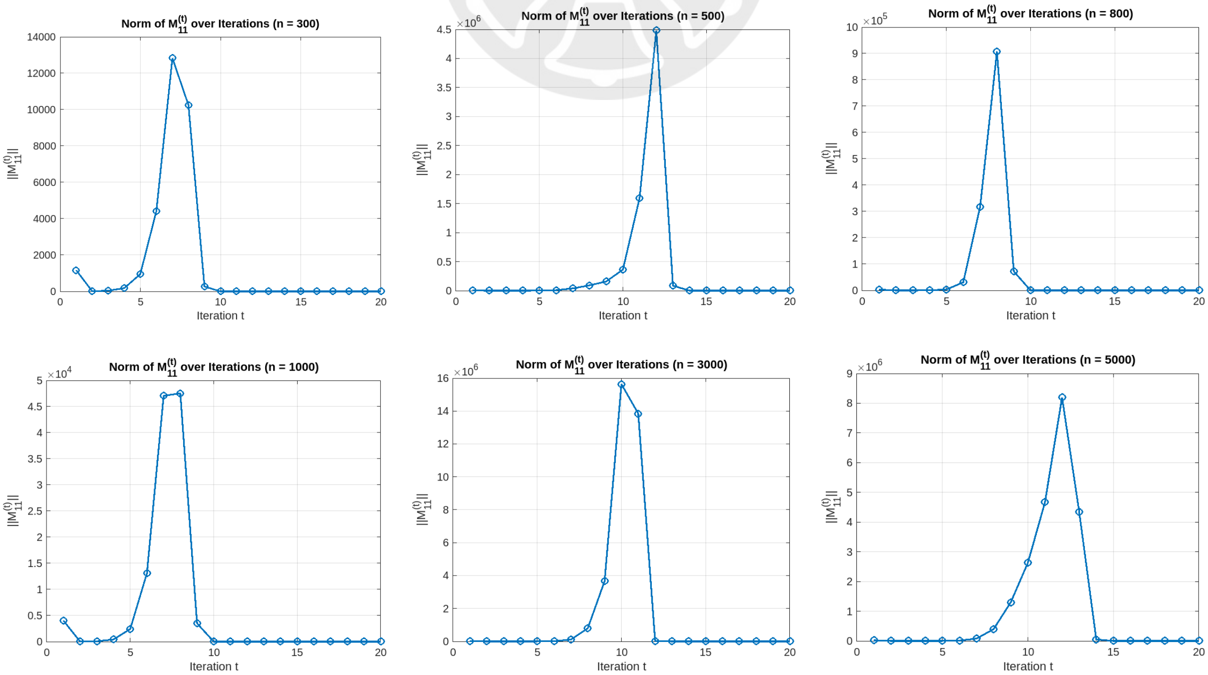

Figure 3: Evolution of $\|M_{11}^{(t)}\|$ over iterations for various matrix sizes

Figure 3 shows the behavior of the norm of $M_{11}^{(t)}$. Initially, the norm increases rather than decreases, and reaches a peak before starting to decline toward zero. The purpose of this experiment is to examine whether the normalization using ℓ can effectively control the growth of M_{11}. From the plots of different matrix sizes, we observe that the control mechanism remains stable once a sufficient number of iterations is reached. Moreover, it prevents numerical divergence during the computation process.

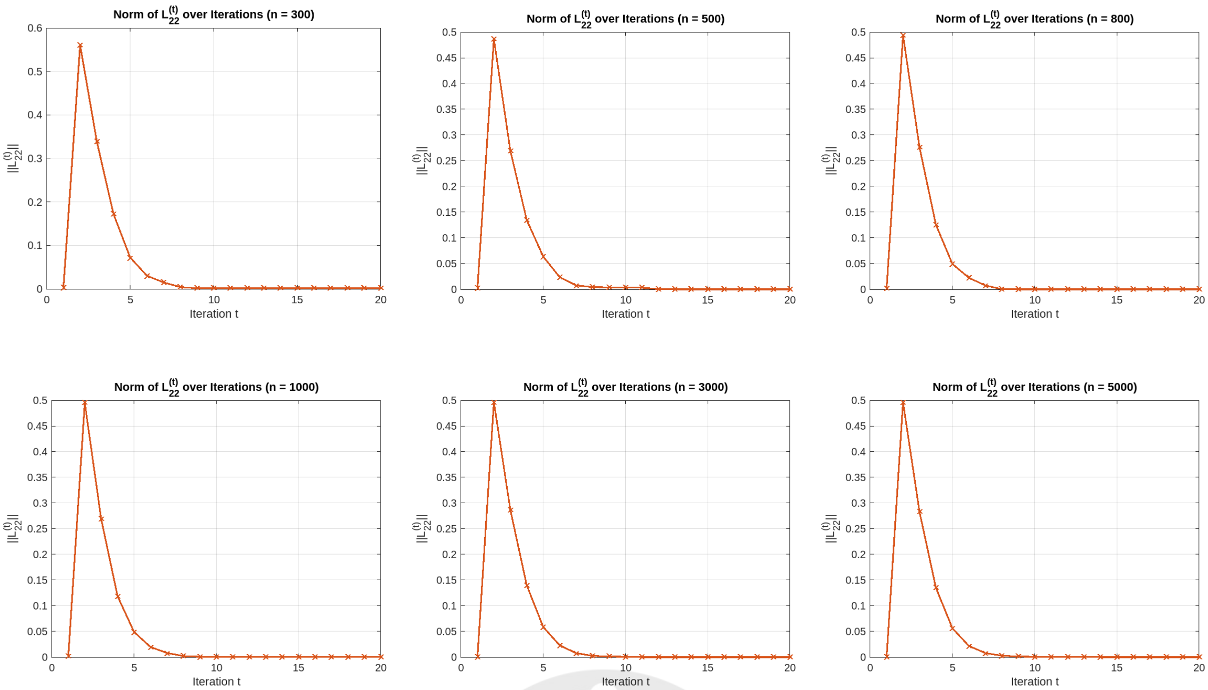

Figure 4: Evolution of $\|L_{22}^{(t)}\|$ over iterations for various matrix sizes

Figure 4 shows the behavior of the norm of $L_{22}^{(t)}$. The norm reaches its maximum at the second iteration, forming a peak in the graph, and then decreases steadily as the iteration count increases. From this figure, we observe that the convergence curves are smooth and consistent across different matrix sizes. This indicates that the adjustment mechanism successfully stabilizes the evolution of L_{22}, regardless of the matrix size.

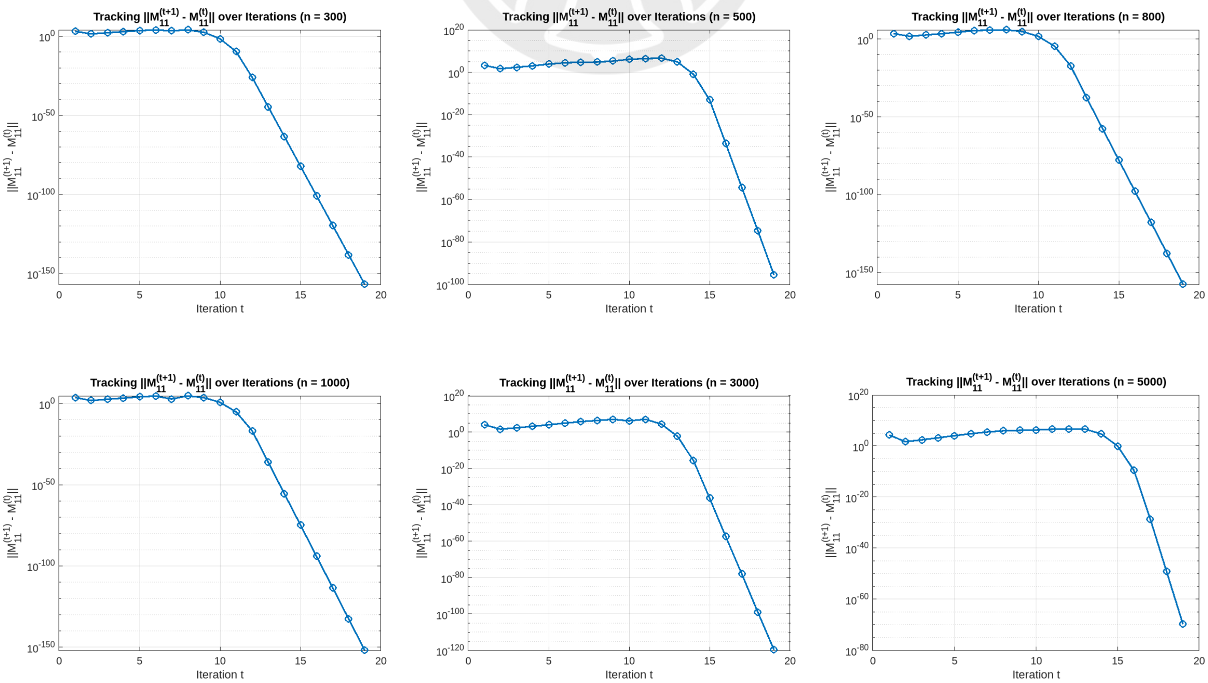

Figure 5: Log-scale tracking of $\|M_{11}^{(t+1)} - M_{11}^{(t)}\|$ for various matrix sizes

In Figure 5, the use of a log-scale makes it difficult to observe noticeable differences during the first ten iterations. However, as the number of iterations increases, the difference between consecutive values decreases rapidly. This trend provides clear evidence of the convergence of the norm of $M_{11}^{(t)}$. The nearly linear decay observed in the log-scale plot suggests that the convergence of $\|M_{11}^{(t)}\|$ is exponential.

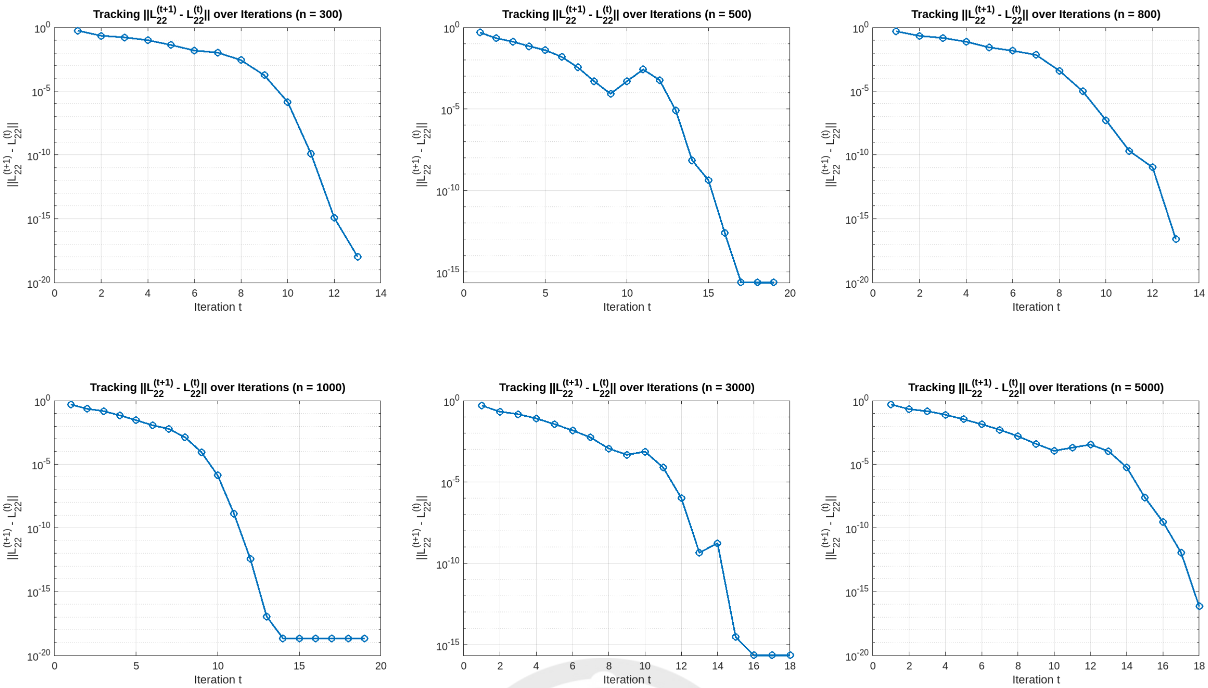

Figure 6: Log-scale tracking of $\|L_{22}^{(t+1)} - L_{22}^{(t)}\|$ for various matrix sizes

In Figure 6, the convergence behavior of $\|L_{22}^{(t+1)} - L_{22}^{(t)}\|$ is less regular compared to that of $\|M_{11}^{(t+1)} - M_{11}^{(t)}\|$ in the log-scale plot. Although the differences do not decrease monotonically, the overall trend is still downward as the iteration count increases. This suggests that, despite some fluctuations, the adjustment mechanism effectively stabilizes the iteration process, allowing the matrix sequence to converge.

From Figures 1 to 6, we can summarize the results of this experiment as follows. It is evident that the normalization factor ℓ plays a crucial role in controlling the convergence behavior of the matrix sequence. Among all components, L_{22} is the first to exhibit convergence, forming a near-perfect decreasing curve starting from iteration 2. However, not all terms converge simultaneously. This is due to the structure of the doubling algorithm, which involves solving a null space and multiplying matrices in each iteration. When we normalize L by $\ell = \|L_{22}\|$, the matrix M is effectively scaled by ℓ as well. As a result, although L_{22} decreases quickly, the corresponding effect on M_{11} is delayed. In fact, M_{11} may initially grow in magnitude before eventually stabilizing. Once L_{22} becomes sufficiently small and approaches convergence, the influence of ℓ on M_{11} decreases, allowing M_{11} to begin its own convergence process. This explains why most other components start to converge around the 10th iteration. Finally, we observe that the convergence of M_{11} is not only stable but also exhibits quadratic convergence. In the following discussion, we will apply numerical techniques to demonstrate the quadratic convergence behavior of M_{11}. To verify the quadratic convergence of M_{11}, we establish the following convergence rate estimation.

Theorem 5.8 (Convergence Rate Estimation). *Suppose the sequence $\{\|M_{11}^{(t)}\|\}_{t\in\mathbb{N}}$ converges to zero, and satisfies the asymptotic relation*

$$\|M_{11}^{(t+1)}\| = c\,\|M_{11}^{(t)}\|^q + o\left(\|M_{11}^{(t)}\|^q\right) \quad \text{as } t \to \infty, \tag{5.7}$$

for some constants $c > 0$ and $q > 0$. Then, for sufficiently large t, the convergence order p can be approximated by

$$q \approx \frac{\log \|M_{11}^{(t+1)}\| - \log \|M_{11}^{(t)}\|}{\log \|M_{11}^{(t)}\| - \log \|M_{11}^{(t-1)}\|}. \tag{5.8}$$

Proof. By the approximation assumption in equation (5.7), we have

$$\|M_{11}^{(t+1)}\| = c\,\|M_{11}^{(t)}\|^q, \quad \|M_{11}^{(t)}\| = c\,\|M_{11}^{(t-1)}\|^q.$$

Dividing the first equation by the second, we obtain

$$\frac{\|M_{11}^{(t+1)}\|}{\|M_{11}^{(t)}\|} = \left(\frac{\|M_{11}^{(t)}\|}{\|M_{11}^{(t-1)}\|} \right)^q .$$

Taking the natural logarithm of both sides yields

$$\log \left(\frac{\|M_{11}^{(t+1)}\|}{\|M_{11}^{(t)}\|} \right) = q \log \left(\frac{\|M_{11}^{(t)}\|}{\|M_{11}^{(t-1)}\|} \right) .$$

Solving for q gives the convergence rate estimate stated in equation (5.8). □

Thus, in the following verification, we use equation (5.8) to confirm the correctness of the convergence rate estimation. To demonstrate this verification, we present a figure that illustrates the computed convergence rate q at each iteration. The results support our hypothesis that M_{11} exhibits quadratic convergence. The observed values of q consistently approach 2 as the iterations proceed, indicating second-order convergence. The experimental setup is the same as in the second experiment. We begin the analysis from $n = 3$, due to the requirements of the convergence rate formula, which involves three consecutive iterations. The figure also records the values of $\|M_{11}^{(t)}\|$ for different matrix sizes and illustrates how the estimated convergence rate varies over iterations. In this analysis, we restrict our attention to the range $q \in [0, 4]$, as it captures the behavior for assessing the convergence order.

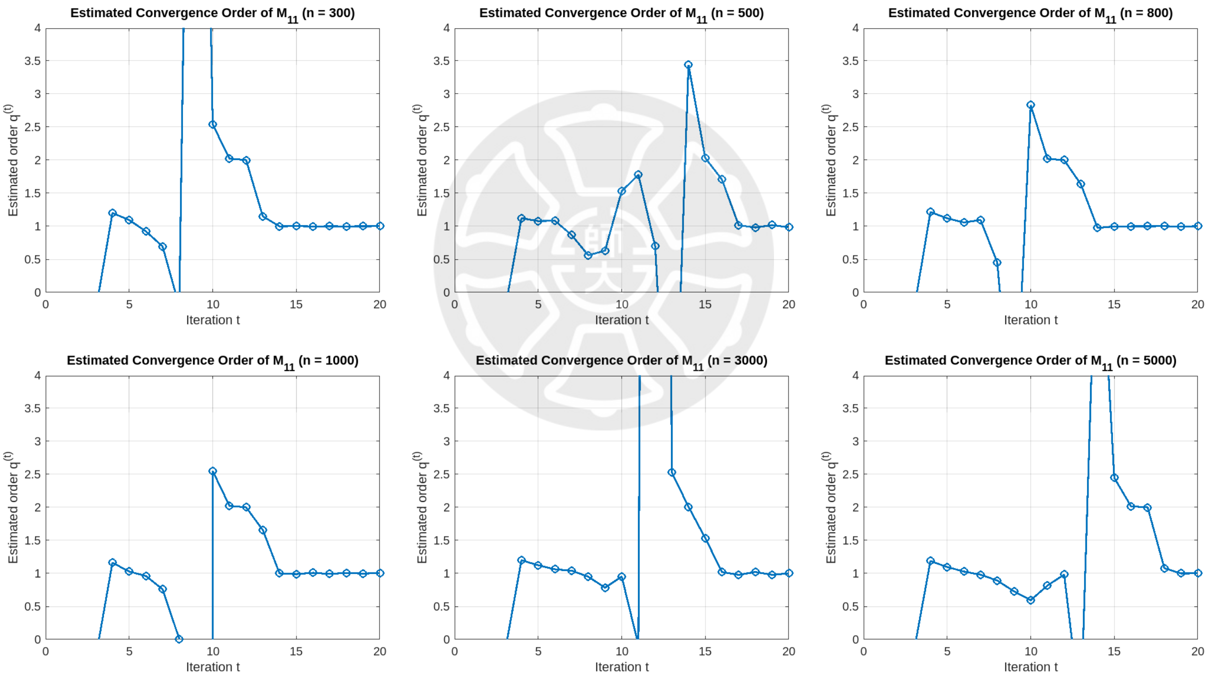

Figure 7: Estimated Convergence Order of M_{11} for Different Matrix Sizes

In Figure 7, we observe that the convergence order of M_{11} increases rapidly. Although the convergence is not evident during the initial iterations, it becomes apparent around iteration 10. From that point onward, the estimated order remains close to 2, confirming the second-order convergence behavior of M_{11}.

To conclude this section, we provide a remark that will be relevant to the subsequent discussion in the next section on the Adaptive Partition Algorithm. This remark is motivated by the convergence analysis in Theorem 5.7, and its proof follows a similar approach.

Remark. *Let $A \in \mathbb{R}^{n\times n}$ be a diagonalizable matrix whose eigenvalues satisfy*

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_p| > |\lambda_{p+1}| \geq \cdots \geq |\lambda_n|.$$

Then, as $k \to \infty$, the matrix pair $[A^k \mid I_n]$ asymptotically satisfies

$$\left[A^k \,\middle|\, I_n \right] \sim \left[\begin{array}{cc|cc} \mathcal{O}(|\lambda_{p+1}|^k) & 0 & I_{n-p} & \mathcal{O}(1) \\ \mathcal{O}(1) & I_p & 0 & \mathcal{O}(|\lambda_p|^{-k}) \end{array} \right],$$

under a block partitioning conforming to submatrices of sizes $(n-p) \times (n-p)$, $(n-p) \times p$, $p \times (n-p)$, and $p \times p$, respectively.

In this section, we have introduced the doubling algorithm by first presenting the standard form and specifying the conditions under which a given matrix pair can be transformed into it. We then applied the doubling algorithm to irreducible and aperiodic stochastic matrices. By partitioning the matrix pair into $(n-1) \times (n-1)$, $(n-1) \times 1$, $1 \times (n-1)$, and 1×1 blocks, we observed that the $(1,1)$-block of M converges to zero, while the $(2,1)$-block of M and the $(1,2)$-block of L converge to the left and right eigenvectors, respectively. These updates are performed in a blockwise manner, and the same iteration strategy can be naturally extended to general matrices beyond the stochastic case.

To further validate the algorithm, we conducted a numerical comparison with the power method. This comparison focused on computation time, the number of iterations required for convergence, and the approximation error of the dominant eigenvalue.

Following this, we extended the analysis to general matrices and observed similar convergence behavior in the $(2,1)$ and $(1,2)$ blocks. We then analyzed the convergence order of the partitioned matrix pair. In a second experiment, we tracked the evolution of the left and right eigenvectors, the norms of submatrices, and their differences across sequential iterations. These observations suggested that the $(1,1)$-block of M exhibits quadratic convergence. To confirm this, we performed a convergence rate estimation for M_{11} and verified its quadratic convergence behavior.

Finally, we conclude with an external remark, which will serve as a foundation for the discussion in the next section.

6 Adaptive Partition Algorithm

In this section, we present a modified version of the doubling algorithm that incorporates an adaptive block partitioning strategy aimed at improving convergence. Unlike the method described in the previous section, this approach dynamically adjusts the partition based on the observed behavior of the $(1,1)$-block of M and the $(2,2)$-block of L, particularly in response to potential divergence. The primary goal remains the extraction of the dominant eigenspace; however, the key innovation lies in monitoring the evolution of selected matrix blocks during the iteration process and applying structural adjustments whenever instability is detected.

This design is inspired by the observation in the previous section, which emphasized the importance of tracking the convergence behavior of submatrices such as M_{11} and L_{22}. Building on that insight, we develop a two-part algorithmic framework.

The first component is a transformation routine that, given any matrix pair and a specified partition index, converts the pair into an extended standard form. This form ensures compatibility with the doubling iteration while preserving the underlying structural properties necessary for convergence.

The second component governs the adaptive adjustment of the partition index based on the behavior of the iteration. When the norm of the selected matrix blocks becomes excessively large—indicating potential numerical divergence—the algorithm increases the partition index and recomputes the extended standard form accordingly.

If the error remains moderate, meaning it is neither diverging nor yet converged, the algorithm proceeds with the doubling iteration as usual, while continuing to monitor the behavior in subsequent steps. This allows for progressive refinement without premature termination or unnecessary structural changes.

Finally, if the monitored error falls below a predefined threshold, the iteration is deemed to have converged. At this point, the process terminates, and the current partition index is retained as the final configuration.

Through this adaptive mechanism, the algorithm becomes more robust and effective across a wider range of input matrices. It maintains structural consistency while dynamically responding to instability during the iterative process.

Algorithm 6 Transform Matrix Pair (A, B, p) into Standard Form

---

Input: Matrices $A, B \in \mathbb{R}^{n\times n}$, and a partition index p
Output: A matrix pair (M, L) in extended standard form
1: Let $n \leftarrow \text{size}(A, 1)$
2: Construct matrix $C \leftarrow \big[A(:,\, n{-}p{+}1{:}n) \,\big|\, B(:,\, 1{:}n{-}p)\big]$
3: Construct permutation matrix

$$K \leftarrow \begin{bmatrix} 0 & I_{n-p} \\ I_p & 0 \end{bmatrix}$$

4: Solve linear systems $X_A \leftarrow C\backslash A, \quad X_B \leftarrow C\backslash B$
5: Compute $M \leftarrow K \cdot X_A$
6: Compute $L \leftarrow K \cdot X_B$
7: **return** (M, L)

Algorithm 7 Adaptive Partition Algorithm with Norm Tracking

Input: A matrix $A \in \mathbb{R}^{n\times n}$
Output: Norm sequences $\|M_{11}^{(t)}\|$, $\|L_{22}^{(t)}\|$, partition indices $p^{(t)}$, and total iteration count
1: Set parameters: tol_{stop}, tol_{grow}, max_iter
2: Initialize: $p \leftarrow 1$, $\max_p \leftarrow n-1$, Iter $\leftarrow 0$, converged $\leftarrow$ false
3: Compute initial matrix pair (M, L) via Algorithm 6 with input (A, I_n, p)
4: Compute initial norms: $\|M_{11}\|$, $\|L_{22}\|$
5: **while** Iter $<$ max_iter **do**
6: Record current norms and partition index

if $\|M_{11}\| < \text{tol}_{\text{stop}}$ **and** $\|L_{22}\| < \text{tol}_{\text{stop}}$ **then**
  converged $\leftarrow$ true, **break**
else if $\|M_{11}\| > \text{tol}_{\text{grow}}$ **or** $\|L_{22}\| > \text{tol}_{\text{grow}}$ **then**
  if $p \geq \max_p$ **then**
    Terminate with warning: maximum partition index reached
  else
    Update: $p \leftarrow p + 1$
    Recompute (M, L) via Algorithm 6 with updated p
    Compute norms from current M, L
  end if
else
  Iter $\leftarrow$ Iter $+ 1$
  Compute null space $\texttt{tmp} \leftarrow \texttt{null}([L^\top,\ -M^\top])$
  Obtain $L_1 \leftarrow (\texttt{tmp}(1{:}n, :))^\top, \quad M_1 \leftarrow (\texttt{tmp}(n{+}1{:}2n, :))^\top$
  $M_2 \leftarrow L_1 M, \quad L_2 \leftarrow M_1 L$
  Compute updated pair (M, L) via Algorithm 6 with input (M_2, L_2, p)
  Compute norms from current M, L
end if
end while
if not converged **then**
  Warn: maximum iterations reached
end if

Based on Algorithm 7 and the remark presented in the previous section, we observe that the eigenvalue $\lambda = 1$ plays a critical role in the partitioning strategy. When the eigenvalues are ordered such that $|\lambda_p| < 1 < |\lambda_{p+1}|$, placing the eigenvalue exactly at the boundary defined by the partitioning of the matrix allows both the $(1, 1)$-block of M and the $(2, 2)$-block of L to converge to zero during the doubling process.

This observation motivates the design of our third experiment. We construct two test cases to investigate the relationship between the partition index and the convergence behavior:

- The first case involves a diagonalizable matrix of size 7, with four eigenvalues strictly inside the unit disk and three outside.
- The second case uses a diagonalizable matrix of size 10, with three eigenvalues inside the unit disk and seven outside.

For each possible partition index $p = 1, 2, \ldots, n-1$, we apply the adaptive doubling algorithm (Algorithm 7) with the following parameters: $\texttt{tol\_stop} = 10^{-10}$, $\texttt{tol\_grow} = 10^4$, and $\texttt{max\_iter} = 200$. During each run, we record the following:

- the partition index used,
- the number of iterations required to reach convergence,
- the norm sequences of the submatrices M_{11} and L_{22} over all iterations.

This setup enables us to evaluate the effectiveness of each partition index and to analyze how the leading and trailing subblocks evolve under the doubling process. The results for the first case are presented in the following figure, which shows the evolution of $\|M_{11}\|$ and $\|L_{22}\|$ over the adaptive iteration process. At each step, the algorithm performs either a doubling update or an adjustment of the partition index, depending on the observed norm behavior. This setup allows us to visualize how the algorithm progresses toward convergence under dynamically changing structural configurations.

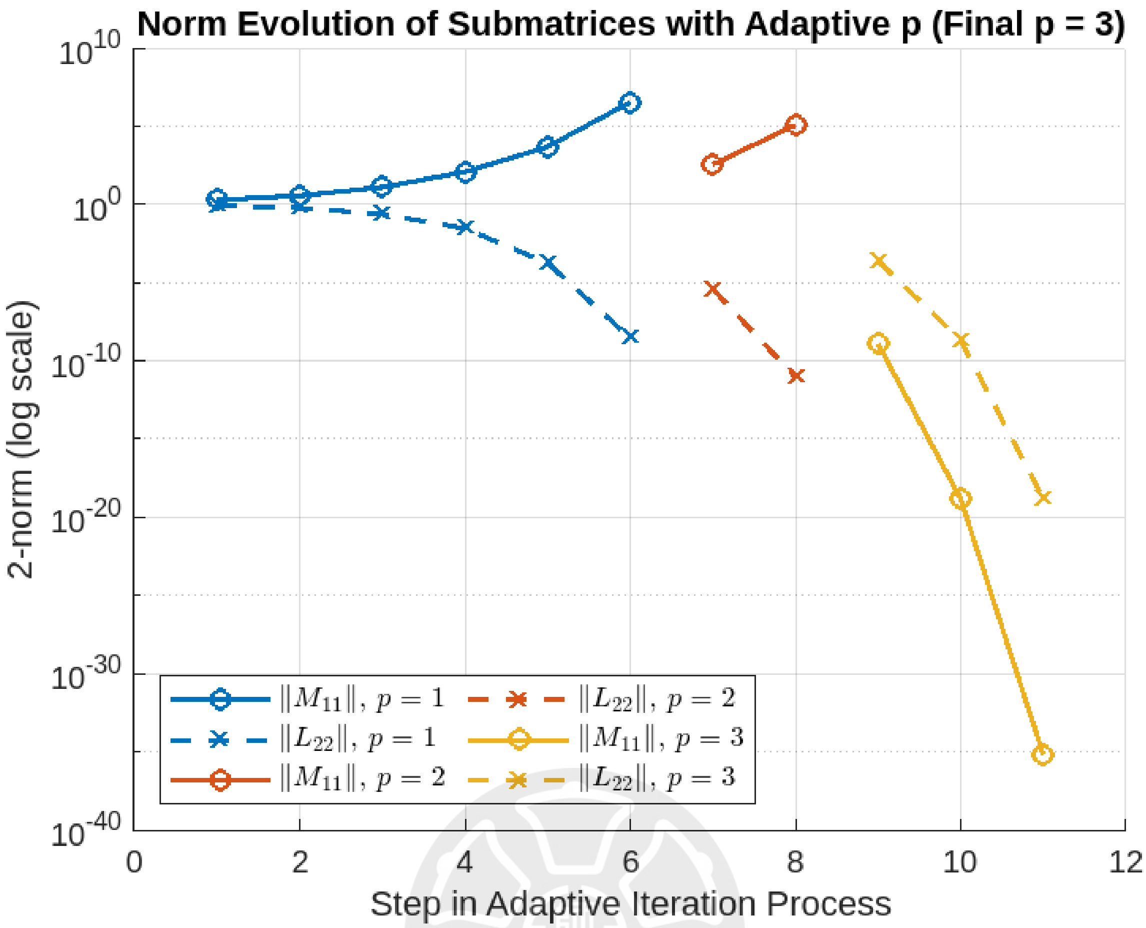

Figure 8: Trajectories of $\|M_{11}\|$ and $\|L_{22}\|$ over the adaptive iteration process. Different colors indicate intervals during which the partition index p remains constant. The final partition index reached is $p = 3$.

As shown in Figure 8, when the partition index is set to $p = 1$, the norm of M_{11} exhibits a divergent trend, while the norm of L_{22} appears to converge. After six doubling iterations, the norm of M_{11} exceeds the predefined growth tolerance, which in this experiment is set to 10^4. This threshold is adjustable depending on the desired sensitivity to divergence.

Once the norm exceeds the growth threshold, the algorithm increments the partition index by one and reconfigures the matrix pair into the standard form accordingly. It then reevaluates the norm under the new configuration. If the updated norm falls below the threshold, the algorithm proceeds with the doubling iteration; otherwise, the partition is adjusted again. This cycle of norm evaluation, partition adjustment, and conditional continuation is repeated whenever the norm of M_{11} becomes excessively large.

Eventually, when the partition index reaches $p = 3$, both $\|M_{11}\|$ and $\|L_{22}\|$ demonstrate simultaneous convergence. This behavior confirms the earlier observation: with four eigenvalues strictly inside the unit disk and three outside, the optimal partition occurs at $p = 3$, effectively separating the spectrum and promoting convergence. This experiment thus supports both the theoretical intuition and the practical effectiveness of the adaptive partitioning strategy.

It is important to note that the iteration steps in the adaptive process do not directly correspond to the number of doubling iterations performed. Each adaptive step involves a check of the submatrix norms and may result either in a doubling update or in an adjustment of the partition index. Therefore, not all adaptive steps represent an actual doubling operation.

In fact, the total number of true doubling steps can be determined by subtracting the number of partition adjustments from the total number of adaptive iterations. That is,

$$\text{Doubling Steps} = \text{Adaptive Steps} - p,$$

where p is the final partition index used. This distinction is crucial when analyzing convergence behavior, as it allows us to more accurately assess the efficiency of the doubling procedure itself.

The results for the second case are presented in the following figure, which illustrates the norm evolution of $\|M_{11}\|$ and $\|L_{22}\|$ across the adaptive iteration. This example further validates the effectiveness of the algorithm under a more unbalanced spectral distribution, where the majority of eigenvalues lie outside the unit disk. Compared to the first case, the norm trajectories in this setting are more irregular due to the dominance of eigenvalues outside the unit circle.

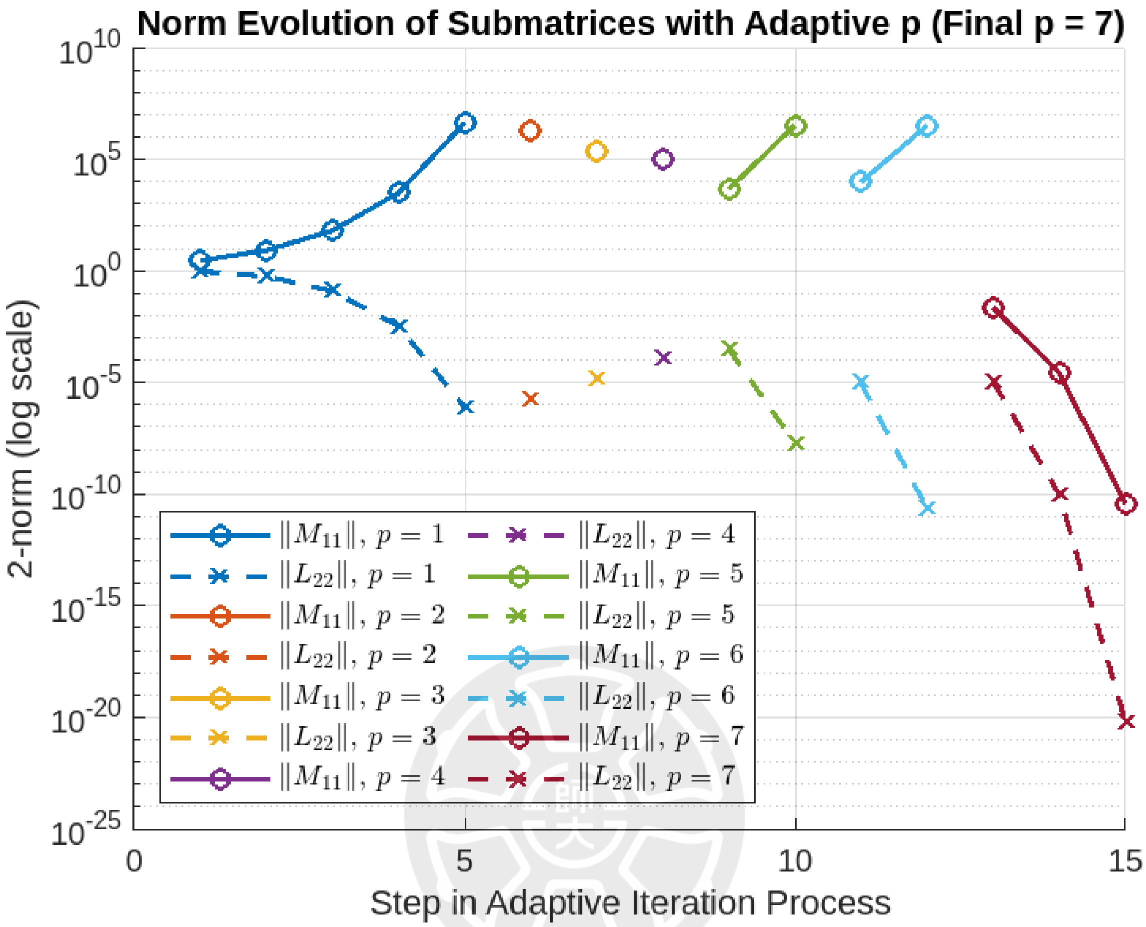

Figure 9: Trajectories of $\|M_{11}\|$ and $\|L_{22}\|$ over the adaptive iteration process. Different colors indicate intervals during which the partition index p remains constant. The final partition index reached is $p = 7$.

In Figure 9, we observe results that are consistent with those of Case 1, further confirming the validity of the proposed partitioning strategy. During iteration steps 5 through 9, the partition index is incrementally adjusted, indicating that the initial choices were suboptimal and required correction. This adjustment process demonstrates the algorithm's ability to detect and respond to improper partitions dynamically. In contrast, if the partition index is poorly chosen and remains unchanged, it may result in only one of the subblocks—either M_{11} or L_{22}—converging while the other diverges. Therefore, the identification of a structurally stable partition is central to the success of the adaptive doubling approach presented in this section.

In the following discussion of this thesis, we shift our focus to the analysis of the dominant eigenspace. Specifically, we aim to understand how the iterative process of the doubling algorithm reveals the structure of this eigenspace through the convergence behavior of certain matrix blocks.

In particular, we observe that when the $(1,1)$-block of M and the $(2,2)$-block of L converge to zero, the remaining nonzero blocks isolate the dominant left and right eigenspaces. This observation forms the basis for characterizing the limiting behavior of the algorithm. We begin by presenting the main theorem that formalizes this relationship and underpins our subsequent analysis of the dominant eigenspace.

Theorem 6.1. *Let (A, I) be a matrix pair, and suppose an adaptive partition index p is identified such that*

$$|\lambda_p| > 1 > |\lambda_{p+1}|,$$

where $\{\lambda_i\}$ are the eigenvalues of A. Then, under the standard form associated with this partition, the system can be represented as

$$\left[\begin{array}{cc|cc} M_{11} & 0 & I_{n-p} & L_{12} \\ M_{21} & I_p & 0 & L_{22} \end{array}\right].$$

As the doubling iteration proceeds, the convergence condition implied by the partition ensures that $M_{11}^* \to 0$ and $L_{22}^* \to 0$. Consequently, the system asymptotically reduces to:

$$\left[\begin{array}{cc|cc} 0 & 0 & I_{n-p} & L_{12}^* \\ M_{21}^* & I_p & 0 & 0 \end{array}\right]. \tag{6.1}$$

From the structure of (6.1) *and the theory of generalized eigenvalues and eigenvectors, we conclude that the dominant eigenspaces can be explicitly characterized as follows. The matrix*

$$\begin{bmatrix} L_{12}^* \\ -I_p \end{bmatrix}$$

*spans the dominant **right** eigenspace, while the matrix*

$$\begin{bmatrix} M_{21}^* & I_p \end{bmatrix}$$

*spans the dominant **left** eigenspace.*

Proof. By equation (6.1), and using the theory of generalized eigenvalues and eigenvectors, we derive the following relation

$$\begin{bmatrix} 0 & 0 \\ M_{21}^* & I_p \end{bmatrix} \begin{bmatrix} V_1 & W_1 \\ V_2 & W_2 \end{bmatrix} = \begin{bmatrix} I_{n-p} & L_{12}^* \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 & W_1 \\ V_2 & W_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix},$$

$$\begin{bmatrix} 0 & 0 \\ M_{21}^* & I_p \end{bmatrix} \begin{bmatrix} V_1 & W_1 \\ V_2 & W_2 \end{bmatrix} \begin{bmatrix} \Lambda_1^{-1} & 0 \\ 0 & I_p \end{bmatrix} = \begin{bmatrix} I_{n-p} & L_{12}^* \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 & W_1 \\ V_2 & W_2 \end{bmatrix} \begin{bmatrix} I_{n-p} & 0 \\ 0 & \Lambda_2 \end{bmatrix},$$

where Λ_1 contains the eigenvalues with modulus greater than one, and Λ_2 contains those with modulus less than one. As the iteration proceeds, $\Lambda_1^{-1} \to 0$ and $\Lambda_2 \to 0$, implying that the dominant and non-dominant components decay in a complementary manner under the transformation. So the equation could be

$$\begin{bmatrix} 0 & 0 \\ M_{21}^* & I_p \end{bmatrix} \begin{bmatrix} 0 & W_1 \\ 0 & W_2 \end{bmatrix} = \begin{bmatrix} I_{n-p} & L_{12}^* \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 & 0 \\ V_2 & 0 \end{bmatrix}.$$

By computing the $(1,1)$-block of the matrix identity, we obtain the following equations

$$\begin{bmatrix} I_{n-p} & L_{12}^* \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = 0$$

These relations imply that

$$\begin{bmatrix} L_{12}^* \\ -I_p \end{bmatrix}$$

spans the dominant **right** eigenspace. Similarly, by observing the $(2,2)$-block of the matrix identity, we obtain

$$\begin{bmatrix} M_{21}^* & I_p \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = 0,$$

which implies that

$$\begin{bmatrix} M_{21}^* & I_p \end{bmatrix}$$

spans the dominant **left** eigenspace. □

In this section, we have presented the complete formulation of the adaptive partition algorithm and designed two representative matrix cases to evaluate its effectiveness. The key idea is to identify an optimal partition such that the $(1,1)$-block of M and the $(2,2)$-block of L both converge to zero. This is achieved by monitoring the norms of these subblocks during each iteration. Depending on their behavior, the algorithm either adjusts the partition index or proceeds with the doubling process, continuing this cycle until convergence is observed in both blocks. Once this condition is met, the resulting matrix pair becomes stable and structure-revealing, allowing us to extract the dominant eigenspace corresponding to the set of eigenvalues with maximal modulus. This outcome aligns precisely with the central objective of our thesis.

In the next section, we summarize the main findings of this work and outline potential directions for future research, thereby bringing the thesis to its conclusion.

7 Conclusion and Future Work

In this section, we summarize the key contributions of this thesis and outline directions for future research.

This thesis investigated the doubling algorithm and its application to the computation of dominant eigenspaces. We began by introducing the standard form for matrix pairs and provided conditions for its existence through a transformation into this form.

The first case we considered was that of an irreducible and aperiodic stochastic matrix. We applied the doubling process to this class of matrices and presented a theorem stating that the $(1, 1)$-block of M converges to zero, the $(2, 1)$-block of M combines with the scalar 1 to form a convergent row vector, and the $(1, 2)$-block of L combines with -1 to form a convergent column vector. This structural result provides a precise characterization of the limiting behavior of the algorithm. These block formulations significantly improve computational efficiency and enable faster implementation in practical settings. By leveraging this structure, we provided a complete description of the doubling algorithm specifically tailored to irreducible and aperiodic stochastic matrices. Furthermore, we compared the performance of the doubling algorithm against the classical power method. Numerical experiments were conducted to validate the theoretical results, demonstrating both the correctness and the computational advantages of our proposed approach.

In the second case, we extended the framework to general matrix pairs beyond irreducible and aperiodic stochastic matrices. We proved that certain structural properties are preserved: specifically, the $(2, 1)$-block of M combined with the scalar 1 forms a convergent row vector, and the $(1, 2)$-block of L combined with -1 forms a convergent column vector. We then analyzed the convergence behavior of the $(1, 1)$-block of M and the $(2, 2)$-block of L, and proposed an adjustment strategy to control their convergence more effectively. Numerical experiments were conducted to observe the convergence of the previously mentioned row and column vectors, as well as to verify that the proposed adjustment can successfully guide the behavior of both M_{11} and L_{22}. To further support the structure-based analysis, we considered the behavior of the doubling iteration $[A^k \mid I_n]$ under the transformation to the standard form. It can be shown that

$$\left[A^k \middle| I_n\right] \sim \left[\begin{array}{cc|cc} \mathcal{O}(|\lambda_2|^k) & 0 & I & \mathcal{O}(1) \\ \mathcal{O}(1) & 1 & 0 & \mathcal{O}(|\lambda_1|^{-k}) \end{array}\right],$$

where λ_1 and λ_2 denote the dominant and subdominant eigenvalues, respectively. This asymptotic form illustrates how dominant and non-dominant eigenspace components decay or grow during iteration, aligning with the observed block behavior in the doubling process. Furthermore, we computed the empirical convergence order of the $(1, 1)$-block of M and observed that it consistently approaches 2, indicating that the doubling algorithm achieves quadratic convergence in this block.

Furthermore, we extended the analysis of the doubling iteration to the more general asymptotic form

$$\left[A^k \middle| I_n\right] \sim \left[\begin{array}{cc|cc} \mathcal{O}(|\lambda_{p+1}|^k) & 0 & I_{n-p} & \mathcal{O}(1) \\ \mathcal{O}(1) & I_p & 0 & \mathcal{O}(|\lambda_p|^{-k}) \end{array}\right],$$

which reflects the spectral separation with $|\lambda_p| < 1 < |\lambda_{p+1}|$. Building on this structure, we proposed a novel adaptive partitioning algorithm. Unlike traditional approaches that rely on the norm of submatrices to guide convergence, our method dynamically selects the partition index by monitoring the norms of the subblocks M_{11} and L_{22}. When the norm of either block exhibits divergent behavior, the algorithm increases the partition index and reconfigures the matrix structure; otherwise, it continues the doubling process. This adaptive mechanism is guided by the asymptotic structure that emerges during iteration. This adaptive partitioning strategy identifies the index p such that the $(1, 1)$-block of M and the $(2, 2)$-block of L converge simultaneously, serving as a practical indicator of an optimal partition. We validated the effectiveness of this approach through numerical experiments. Moreover, under the correct partitioning, we were able to explicitly extract both the left and right dominant eigenspaces, confirming the theoretical implications of the algorithm.

It is important to note that the limiting form derived in our theorem assumes that the matrix is diagonalizable. We do not consider the case of non-diagonalizable matrices, as this would require dealing with the Jordan canonical form, which introduces additional complications and is beyond the scope of this thesis. One computational concern is that the doubling algorithm involves repeated inversion of submatrices. Although these submatrices are typically smaller than the original matrix, the cost of matrix inversion can still be significant compared to simpler iterative methods such as the power method. Moreover, in the case of general matrices, the adjustment strategy in the doubling algorithm cannot rely on the convergence of both the $(1, 1)$-block of M and the $(2, 2)$-block of L simultaneously. Since we cannot guarantee that both blocks converge at the same time, one block may diverge while the other converges, making the adjustment unreliable under certain partitions. Finally, in the current version of the adaptive partition algorithm, the partition index is gradually adjusted during the iteration process. The first index that results in simultaneous convergence of the monitored subblocks is selected as the final partition. However, the algorithm does not incorporate other strategies—such as predictive heuristics or spectral estimates—to proactively guide the selection of the best partition.

Future research may explore the following directions:

- Investigate alternative formulations of the doubling algorithm that avoid explicit matrix inversion. Eliminating the need for submatrix inversion could significantly improve computational efficiency, particularly in large-scale problems.

- Analyze the behavior of the doubling iteration when multiple dominant eigenvalues are present. Specifically, when two or more eigenvalues share the largest modulus while the remaining ones are strictly smaller, it remains unclear whether the current doubling scheme ensures stable convergence. Further study is needed to understand potential issues and propose solutions in such cases.

- Address the scenario where $|\lambda_p| = |\lambda_{p+1}| = 1$. This situation challenges the partitioning strategy, as the spectrum cannot be clearly divided into dominant and non-dominant components. Future work could aim to develop heuristics or refined criteria to handle such borderline cases.

- Improve the efficiency of the adaptive partition algorithm by incorporating techniques such as binary search. Instead of incrementally increasing the partition index, a more efficient search strategy could reduce computational cost while still identifying an optimal or near-optimal partition.

A MATLAB Code

Listing A.1: Power Method for Dominant Eigenvector

```
function [x, time, iter] = power_method_simple(A, x0, tol, max_iter)
% POWER_METHOD_SIMPLE Basic power method to compute dominant eigenvector
% Inputs:
% A : square matrix
% x0 : initial vector (non-zero)
% tol : stopping tolerance
% max_iter : maximum number of iterations
% Outputs:
% x : approximate dominant eigenvector
% time : computation time
% iter : number of iterations

    x = x0 / norm(x0); % Normalize initial vector
    iter = 0;
    tic;
    while iter < max_iter
        y = A * x;
        x_new = y / norm(y);
        if norm(x_new - x) < tol
            break;
        end
        x = x_new;
        iter = iter + 1;
    end
    time = toc;
end
```

This listing provides MATLAB implementations for Algorithm 1. Listing A.1 implements the classical power method, which iteratively approximates the dominant eigenvector of a matrix by repeatedly multiplying with the matrix and normalizing. The convergence behavior depends on the ratio $|\lambda_2/\lambda_1|$, where λ_1 is the dominant eigenvalue and λ_2 is the second-largest in magnitude.

Listing A.2: Doubling Algorithm for Irreducible and Aperiodic Stochastic Matrices with Transform Matrix Pair (S,I_n) into Standard Form

```
function [M_21, L_12, iter, time] = da_iteration(H, p)
% DA_ITERATION performs the DA update on matrix H
% Inputs:
% H : a row-stochastic matrix of size n x n
% p : block size (typically 1)
% Outputs:
% M_21 : final left eigenvector block from DA
% L_12 : final right eigenvector block from DA
% iter : number of iterations performed
% time : total elapsed time

    n = size(H, 1);
```

```
    % Partition H
    M_11 = H(1:n-p, 1:n-p);
    M_12 = H(1:n-p, n-p+1:n);
    M_21 = H(n-p+1:n, 1:n-p);
    M_22 = H(n-p+1:n, n-p+1:n);

    % Partition identity matrix
    L_11 = eye(n-p);
    L_12 = zeros(n-p, p);
    L_21 = zeros(p, n-p);
    L_22 = eye(p);

    % Transform the original matrix to standard form
    M_11_new = M_11 - M_12 * (M_22 \ M_21);
    M_12_new = L_12;
    M_21_new = M_22 \ M_21;
    M_22_new = L_22;
    L_11_new = L_11;
    L_12_new = -M_12 / M_22;
    L_21_new = L_21;
    L_22_new = L_22 / M_22;

    % Update the original matrix
    M_11 = M_11_new;
    M_12 = M_12_new;
    M_21 = M_21_new;
    M_22 = M_22_new;
    L_11 = L_11_new;
    L_12 = L_12_new;
    L_21 = L_21_new;
    L_22 = L_22_new;

    iter = 0;
    tic
    % DA iterations
    while any(abs(M_11) > 1e-15, 'all')
        % Transform the standard form to square (DA)
        Q = L_11 + L_12 * M_21 / (M_22 - M_21 * L_12);
        M_11_new = M_11 * Q * M_11;
        M_21_new = L_22 * M_21 * Q * M_11 + M_21;
        L_12_new = L_12 + M_11 * Q * L_12 * L_22;
        L_22_new = (L_22^2) / (M_22 - M_21 * L_12);

        % Update the matrix
        M_11 = M_11_new;
        M_21 = M_21_new;
        L_12 = L_12_new;
        L_22 = L_22_new;

        iter = iter + 1;
    end
    time = toc;
end
```

Listing A.2 presents a MATLAB implementation of the doubling algorithm for irreducible and aperiodic stochastic matrices, corresponding to Algorithm 2 and Algorithm 3. The procedure begins by transforming the matrix pair (S, I_n) into the standard form described earlier in the thesis. It then applies a series of doubling iterations to compute the left and right dominant eigenvectors.

Listing A.3: Comparison of the Power Method and Doubling Algorithm for Computing the Dominant Eigenvector of Irreducible and Aperiodic Stochastic Matrices.

```
% MAIN_COMPARISON Compare DA and Power Method
clear; clc;
rng(42);

num = 500; % Matrix size: 500, 1000, 3000, 5000, 8000, 10000
R = rand(num); % Random real matrix
A = (R + R') / 2; % Make it symmetric
A = A ./ sum(A, 2); % Make it row-stochastic

% === Power Method ===
x0 = rand(num, 1);
tol = 1e-15;
max_iter = 1000;
[a_power, time_power, iter_power] = power_method_simple(A, x0, tol, max_iter);

% Normalize to match eigenvector form (last entry = -1)
a_power = -a_power / a_power(end);

% === Doubling Algorithm (DA) ===
[M_21, L_12, iter_da, time_da] = da_iteration(A, 1);
a_da = [L_12; -1];

% === True Eigenvectors from eig() ===
[V, D, W] = eig(A); % V: right eigenvectors, W: left eigenvectors'
a_true = -V(:,1) / V(end,1); % Normalize right eigenvector
v = W'; % W is left eigenvectors', so transpose
b_true = v(1,:) / v(1,end); % Normalize left eigenvector
b_da = [M_21 1]; % Approximate left eigenvector from DA

% === Error Comparison ===
err_power = norm(a_true - a_power);
err_da = norm(a_true - a_da);
err_left_da = norm(b_true - b_da);

% === Output Results ===
fprintf("Matrix size: %d\n", num);
fprintf("Power Method: time = %.4f s, iter = %d, error = %.2e\n", ...
       time_power, iter_power, err_power);
fprintf("DA : time = %.4f s, iter = %d, error = %.2e\n", ...
       time_da, iter_da, err_da);
fprintf("DA (left eig vec error) = %.2e\n", err_left_da);
```

Listing A.3 compares the performance of the doubling algorithm and the power method on a synthetically generated irreducible and aperiodic stochastic matrix. The results include computation time, iteration counts, and numerical errors with respect to the true eigenvectors computed by MATLAB's built-in `eig` function.

Listing A.4: Transformation of Matrix Pair (A, B) into Standard Form

```
% This function transforms a matrix pair (A, B) into the following standard form:
% M = [* ((n-1)x(n-1)) 0 ((n-1)x1); * (1x(n-1)) 1 (1x1)]
% L = [I ((n-1)x(n-1)) * ((n-1)x1); 0 (1x(n-1)) * (1x1)]

function [M, L] = SForm1(A, B)
    % This function computes two matrices M and L based on inputs A and B.
    % A, B : square matrices of size n x n
```

```
    n = size(A, 1); % Automatically determine n from the size of A

    % Create matrix C by combining the last column of A and the first n-1 columns of B
    C = [A(:, n), B(:, 1:(n-1))];

    % Define permutation matrix K
    K = [zeros(n-1,1) eye(n-1); 1 zeros(1,n-1)];

    % Compute M and L
    M = K * (C \ A);
    L = K * (C \ B);
end
```

Listing A.4 implements the standard form transformation described in Algorithm 4, which converts a matrix pair (A, B) into a structured pair (M, L) suitable. The resulting matrices satisfy the following form:

$$M = \begin{bmatrix} * & 0 \\ * & 1 \end{bmatrix}, \quad L = \begin{bmatrix} I & * \\ 0 & * \end{bmatrix},$$

where $*$ denotes unspecified matrix blocks of appropriate size, and the final column of M and the final row of L are structurally simplified. This transformation is achieved by constructing a new basis using a matrix C composed of the last column of A and the first $n-1$ columns of B. The permutation matrix K reorders the basis to yield the desired block structure. This form facilitates further analysis, such as the application of the doubling algorithm or the identification of dominant eigenspaces.

Listing A.5: Normalized Doubling Iteration with Norm Tracking and Eigenvector Approximation

```
function [M11_norms, L22_norms, V_seq, W_seq, v_final, w_final] =
    NormalizedDoublingTrack(A, MaxStep)
% NormalizedDoublingTrack -
% Track submatrices and eigenvectors during normalized doubling
%
% Inputs:
% A - Symmetric matrix A of size n x n (real-valued)
% MaxStep - Maximum number of iterations
%
% Outputs:
% M11_norms - 1 x MaxStep array of M11 norms over iterations
% L22_norms - 1 x MaxStep array of L22 norms over iterations
% V_seq - n x MaxStep matrix, each column is v(t)
% W_seq - MaxStep x n matrix, each row is w(t)
% v_final - Final right eigenvector approximation
% w_final - Final left eigenvector approximation

    n = size(A, 1);
    [M, L] = SForm1(A, eye(n)); % Apply standard form transform (A, I)

    M11_norms = zeros(1, MaxStep);
    L22_norms = zeros(1, MaxStep);
    V_seq = zeros(n, MaxStep);
    W_seq = zeros(MaxStep, n);

    % Initialize tracking
    M11_norms(1) = norm(M(1:n-1, 1:n-1));
    L22_norms(1) = abs(L(n, n));
    V_seq(:, 1) = [L(1:n-1, n); -1];
    W_seq(1, :) = [M(n, 1:n-1), 1];

    for t = 2:MaxStep
        % Normalize
```

```
        l = abs(L(n, n));
        L = L / l;

        % Null space calculation
        tmp = null([L' -M']);
        L1 = tmp(1:n, :)';
        M1 = tmp(n+1:end, :)';

        % Doubling and transforming to standard form
        M2 = L1 * M;
        L2 = M1 * L;
        [M, L] = SForm1(M2, L2);

        % Record tracking info
        M11_norms(t) = norm(M(1:n-1, 1:n-1));
        L22_norms(t) = abs(L(n, n));
        V_seq(:, t) = [L(1:n-1, n); -1];
        W_seq(t, :) = [M(n, 1:n-1), 1];
    end

    % Final eigenvectors
    v_final = V_seq(:, MaxStep);
    w_final = W_seq(MaxStep, :);
end
```

Listing A.5 presents the MATLAB implementation of Algorithm 5, which performs a doubling iteration on a matrix $A \in \mathbb{R}^{n\times n}$ after transforming it into standard form. The algorithm tracks the convergence behavior of the $(1,1)$-block of matrix M and the $(2,2)$-block of matrix L, while simultaneously approximating the dominant left and right eigenvectors. At each iteration, the algorithm normalizes the system using the (n,n) entry of L, applies a basis transformation via the null space of the augmented matrix $[L^\top \quad -M^\top]$, and updates the matrices through the standard form transformation. The norms $\|M_{11}^{(t)}\|$ and $|L_{22}^{(t)}|$ are recorded over time, along with the evolving eigenvector approximations $v^{(t)}$ and $w^{(t)}$. This implementation provides valuable insights into the convergence pattern and eigenvector behavior of the doubling algorithm under normalization, especially in the presence of dominant eigenvalues.

Listing A.6: Evaluation of Convergence Behavior in the Normalized Doubling Algorithm

```
n = 300; % 500, 800, 1000, 3000, 5000
rng(42);
R = randn(n);
A = R' * R;
MaxStep = 20;

[M11_norms, L22_norms, V_seq, W_seq, v_final, w_final] = NormalizedDoublingTrack(A,
    MaxStep);

% === True Eigenvectors ===
[V, D, W] = eig(A);
[~, idx] = max(max(abs(D)));
v_true = -V(:,idx) / V(end,idx);
v = W';
w_true = v(idx,:) / v(idx,end);

% === Compute Errors ===
v_errors = vecnorm(V_seq - v_true, 2, 1); % Column-wise error for v
w_errors = vecnorm(W_seq - w_true, 2, 2); % Row-wise error for w

% === Plot v^{(t)} error ===
figure;
```

```
semilogy(1:MaxStep, v_errors, '-o', 'LineWidth', 1.5);
xlabel('Iteration t');
ylabel('||v^{(t)} - v^*||');
title(sprintf('Convergence of Right Eigenvector v^{(t)} (n = %d)', n));
grid on;

% === Plot w^{(t)} error ===
figure;
semilogy(1:MaxStep, w_errors, '-x', 'LineWidth', 1.5, 'Color', [0.6 0.2 0.8]);
xlabel('Iteration t');
ylabel('||w^{(t)} - w^*||');
title(sprintf('Convergence of Left Eigenvector w^{(t)} (n = %d)', n));
grid on;

% === Plot M_{11}^{(t)} norm ===
figure;
plot(1:MaxStep, M11_norms, '-o', 'LineWidth', 1.5);
xlabel('Iteration t');
ylabel('||M_{11}^{(t)}||');
title(sprintf('Norm of M_{11}^{(t)} over Iterations (n = %d)', n));
grid on;

% === Plot L_{22}^{(t)} norm ===
figure;
plot(1:MaxStep, L22_norms, '-x', 'LineWidth', 1.5, 'Color', [0.85 0.33 0.1]);
xlabel('Iteration t');
ylabel('||L_{22}^{(t)}||');
title(sprintf('Norm of L_{22}^{(t)} over Iterations (n = %d)', n));
grid on;

% === Delta Norm of M_{11} ===
delta_M11_norms = zeros(1, MaxStep - 1);
for t = 1:MaxStep - 1
    delta = M11_norms(t+1) - M11_norms(t);
    delta_M11_norms(t) = norm(delta);
end

figure;
semilogy(1:MaxStep-1, delta_M11_norms, '-o', 'LineWidth', 1.5);
xlabel('Iteration t');
ylabel('||M_{11}^{(t+1)} - M_{11}^{(t)}||');
title(sprintf('Tracking ||M_{11}^{(t+1)} - M_{11}^{(t)}|| (n = %d)', n));
grid on;

% === Delta Norm of L_{22} ===
delta_L22_norms = zeros(1, MaxStep - 1);
for t = 1:MaxStep - 1
    delta = L22_norms(t+1) - L22_norms(t);
    delta_L22_norms(t) = norm(delta);
end

figure;
semilogy(1:MaxStep-1, delta_L22_norms, '-o', 'LineWidth', 1.5);
xlabel('Iteration t');
ylabel('||L_{22}^{(t+1)} - L_{22}^{(t)}||');
title(sprintf('Tracking ||L_{22}^{(t+1)} - L_{22}^{(t)}|| (n = %d)', n));
grid on;
```

```
% === Estimate Convergence Order q of M_{11} ===
MaxStep = length(M11_norms);
q_est = zeros(1, MaxStep - 2); % Start from t = 3

for t = 3:MaxStep
    num = log(M11_norms(t)) - log(M11_norms(t-1));
    den = log(M11_norms(t-1)) - log(M11_norms(t-2));
    q_est(t-2) = num / den;
end

% === Display and Plot Estimated Order ===
disp('Estimated convergence order q:');
disp(q_est');

figure;
plot(3:MaxStep, q_est, 'o-', 'LineWidth', 1.5);
xlabel('Iteration t');
ylabel('Estimated order q^{(t)}');
title(sprintf('Estimated Convergence Order of M_{11} (n = %d)', n));
ylim([0 4]);
grid on;
```

Listing A.6 implements the second experiment, which evaluates the convergence behavior of the normalized doubling algorithm when applied to a symmetric positive semi-definite matrix $A \in \mathbb{R}^{n\times n}$. The matrix is constructed as $A = R^\top R$, where R is a random Gaussian matrix, ensuring that A is symmetric and positive semi-definite. The experiment tracks several key quantities across a fixed number of iterations:

- The norms of the $(1,1)$-block of M, denoted $\|M_{11}^{(t)}\|$, and the $(2,2)$-block of L, denoted $\|L_{22}^{(t)}\|$.
- The right and left eigenvector approximations $v^{(t)}$ and $w^{(t)}$ at each iteration, compared to the true dominant eigenvectors v^* and w^*.
- The convergence error $\|v^{(t)} - v^*\|$ and $\|w^{(t)} - w^*\|$, plotted in logarithmic scale.
- The differences between sequential iterates $\|M_{11}^{(t+1)} - M_{11}^{(t)}\|$ and $\|L_{22}^{(t+1)} - L_{22}^{(t)}\|$, to assess convergence smoothness.
- The estimated convergence order of $M_{11}^{(t)}$ computed using the standard log-ratio formula:

$$q^{(t)} = \frac{\log\|M_{11}^{(t)}\| - \log\|M_{11}^{(t-1)}\|}{\log\|M_{11}^{(t-1)}\| - \log\|M_{11}^{(t-2)}\|}.$$

Several plots are generated to visualize the convergence trends and estimate the convergence rate. This empirical evaluation confirms the expected near-quadratic convergence behavior of the normalized doubling process for structured matrices like the one used in this test case.

Listing A.7: Transformation of Matrix Pair (A, B, p) into Standard Form

```
function [M, L] = SForm_expand(A, B, p)
% SForm_expand - Transform matrix pair (A, B, p) into block standard form that gives
% M = [* ((n-p)x(n-p)) 0 ((n-p)xp); * (px(n-p)) I (pxp)]
% L = [I ((n-p)x(n-p)) * ((n-p)xp); 0 (px(n-p)) * (pxp)]
%
% Inputs:
% A, B - n x n matrices
% p - partition index (number of trailing rows/cols to move)
%
% Output:
% M, L - transformed matrices

    n = size(A, 1);
```

```
    % Combine trailing columns of A and leading columns of B into C
    % C: n x n, full-rank block to build new basis
    C = [A(:, (n-p+1):n), B(:, 1:(n-p))];

    % Permutation matrix K: move bottom p rows to the bottom in the new basis
    K = [zeros(n - p, p), eye(n - p); eye(p), zeros(p, n - p)];

    % Transform A and B using the new basis defined by C and permutation K
    M = K * (C \ A);
    L = K * (C \ B);
end
```

Listing A.7 implements the standard form transformation described in Algorithm 6, which converts a matrix pair (A, B, p) into a structured pair (M, L) suitable. The resulting matrices satisfy the following form:

$$M = \begin{bmatrix} * & 0 \\ * & I_p \end{bmatrix}, \quad L = \begin{bmatrix} I_{n-p} & * \\ 0 & * \end{bmatrix},$$

where $*$ denotes unspecified matrix blocks of appropriate size, and the final column of M and the final row of L are structurally simplified. This transformation is accomplished by constructing a new basis using a matrix C, which consists of the last p columns of A and the first $n-p$ columns of B. A permutation matrix K is then applied to reorder the basis, producing the desired block structure. This resulting form enables subsequent procedures, such as applying the adaptive partition algorithm or extracting the dominant eigenspace.

Listing A.8: Adaptive Partitioning for Doubling Algorithm with Norm Tracking

```
function [m11_norms, l22_norms, p_seq, Iter] = adaptive_partition_da_trace(A)
% adaptive_partition_da_trace - Adaptive doubling algorithm tracking M11, L22 norms
    and partition index.
% Stops when both norms are less than 1e-10.
%
% Input:
% A - n x n real matrix
%
% Output:
% m11_norms - 1 x Iter array of ||M11||_2 values
% l22_norms - 1 x Iter array of ||L22||_2 values
% p_seq - 1 x Iter array of partition indices used
% Iter - Total number of iterations

    tol_stop = 1e-10;
    tol_grow = 1e+4;
    max_iter = 200;

    n = size(A, 1);
    p = 1;
    max_p = n - 1;

    % Initialize
    [M, L] = SForm_expand(A, eye(n), p);
    m11_norm = norm(M(1:(n-p),1:(n-p)),2);
    l22_norm = norm(L((n-p+1):n,(n-p+1):n),2);

    m11_norms = [];
    l22_norms = [];
    p_seq = [];

    Iter = 0;
    converged = false;
```

```
    while Iter < max_iter

        % Record norms and partition index
        m11_norms(end+1) = m11_norm;
        l22_norms(end+1) = l22_norm;
        p_seq(end+1) = p;

        if (m11_norm < tol_stop) && (l22_norm < tol_stop)
            converged = true;
            break;

        elseif (m11_norm > tol_grow) || (l22_norm > tol_grow)
            if p >= max_p
                warning('p reached maximum (%d), forced to stop adjustment.', max_p);
                break;
            end
            p = p + 1;
            [M, L] = SForm_expand(M, L, p);
            m11_norm = norm(M(1:(n-p),1:(n-p)),2);
            l22_norm = norm(L((n-p+1):n,(n-p+1):n),2);

        else
            Iter = Iter + 1;
            tmp = null([L' -M']);
            L1 = tmp(1:n, :)';
            M1 = tmp(n+1:end, :)';

            M2 = L1 * M;
            L2 = M1 * L;
            [M, L] = SForm_expand(M2, L2, p);

            m11_norm = norm(M(1:(n-p),1:(n-p)),2);
            l22_norm = norm(L((n-p+1):n,(n-p+1):n),2);

        end
    end

    if ~converged
        warning('Maximum number of iterations (%d) reached. Algorithm did not converge.
            ', max_iter);
    end
end
```

Listing A.8 provides the MATLAB implementation of the Adaptive Partition Algorithm with norm tracking, corresponding to Algorithm 7. This function evaluates the convergence behavior of the $(1, 1)$-block of M and the $(2, 2)$-block of L. At each iteration, it either adjusts the partition index or applies a doubling update, depending on the observed norm behavior. The norms $\|M_{11}\|$ and $\|L_{22}\|$ are recorded throughout the process until both converge below a specified tolerance (10^{-10}). The algorithm returns the complete norm sequences, the number of iterations required for convergence, and the final partition index at which both target blocks satisfy the convergence criterion.

Listing A.9: Numerical Experiment for Adaptive Partition Algorithm Using a Diagonalizable Matrix with Prescribed Eigenvalues

```
% Case 1: 4 eigenvalues inside the unit disk, 3 outside
% eigvals_inside = [0.1, 0.3, 0.5, 0.7];
% eigvals_outside = [1.2, 1.5, 2.0];

% Case 2: 3 eigenvalues inside the unit disk, 7 outside
```

```
% eigvals_inside = [0.3, 0.5, 0.9];
% eigvals_outside = [1.2, 1.5, 1.8, 2.0, 2.3, 2.5, 3.0];
eigvals = [eigvals_inside, eigvals_outside];

% Create diagonal matrix and construct a diagonalizable matrix A
D = diag(eigvals);
n = length(D);
[Q, ~] = qr(randn(n)); % Random orthogonal matrix
A = Q * D / Q; % Diagonalizable matrix A

% Run the adaptive doubling algorithm
[m11_norms, l22_norms, p_seq, Iter] = adaptive_partition_da_trace(A);

% === Plot Settings ===
colors = lines(10); % Up to 10 unique partition indices
unique_p = unique(p_seq);

figure; hold on;

for i = 1:length(unique_p)
    p_val = unique_p(i);
    idx = find(p_seq == p_val);

    % Split into continuous segments for consistent coloring
    split_idx = [1, find(diff(idx) > 1) + 1, length(idx) + 1];

    for j = 1:(length(split_idx) - 1)
        range = idx(split_idx(j):split_idx(j + 1) - 1);

        plot(range, m11_norms(range), '-o', ...
             'Color', colors(i,:), ...
             'DisplayName', sprintf('$\\|M_{11}\\|$, $p = %d$', p_val), ...
             'LineWidth', 1.5);

        plot(range, l22_norms(range), '--x', ...
             'Color', colors(i,:), ...
             'DisplayName', sprintf('$\\|L_{22}\\|$, $p = %d$', p_val), ...
             'LineWidth', 1.5);
    end
end

% Labels and legend
xlabel('Step in Adaptive Iteration Process');
ylabel('2-norm (log scale)');
set(gca, 'YScale', 'log');

final_p = p_seq(end);
title(sprintf('Norm Evolution of Submatrices with Adaptive p (Final p = %d)', final_p)
    );

legend('Interpreter', 'latex', 'NumColumns', 2, 'Location', 'southwest');
legend show;
grid on;
```

Listing A.9 presents a MATLAB script for testing the Adaptive Partition Algorithm on a diagonalizable matrix constructed with prescribed eigenvalues. Two cases are considered: one with a mix of eigenvalues strictly inside and outside the unit disk. The matrix is constructed via similarity transformation using a random orthogonal matrix to ensure diagonalizability. For each partition index p, the script applies the algorithm and plots the norm trajectories of the $(1,1)$-block of M and the $(2,2)$-block of L across iterations, providing a visual representation of the convergence behavior and the variation of the partition index throughout the adaptive process.

References

[1] G. Ammar and V. Mehrmann. "On Hamiltonian and symplectic Hessenberg forms". In: *Linear Algebra Appl.* 149 (1991), pp. 55–72. ISSN: 0024-3795. DOI: https://doi.org/10.1016/0024-3795(91)90325-Q. URL: https://www.sciencedirect.com/science/article/pii/002437959190325Q.

[2] S. L. Brunton and J. N. Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. 2nd ed. Cambridge University Press, 2022. ISBN: 978-1009098489.

[3] E. K.-W. Chu, H.-Y. Fan, and W.-W. Lin. "A structure-preserving doubling algorithm for continuous-time algebraic Riccati equations". In: *Linear Algebra Appl.* 396 (2005), pp. 55–80. ISSN: 0024-3795. DOI: https://doi.org/10.1016/j.laa.2004.10.010. URL: https://www.sciencedirect.com/science/article/pii/S0024379504004434.

[4] S. H. Friedberg, A. J. Insel, and L. E. Spence. *Linear Algebra*. 4th ed. Pearson, 2002. ISBN: 978-0130084514.

[5] R. A. Horn and C. R. Johnson. *Matrix Analysis*. 2nd ed. Cambridge University Press, 2012. ISBN: 978-0521548236.

[6] T.-M. Huang. *PageRank*. Lecture slides. Class lecture, National Taiwan Normal University. 2024.

[7] T.-M. Huang and W.-W. Lin. "Power and Inverse Power Methods". Lecture notes, National Taiwan Normal University. 2017.

[8] W.-W. Lin and S.-F. Xu. "Convergence Analysis of Structure-Preserving Doubling Algorithms for Riccati-Type Matrix Equations". In: *SIAM J. Matrix Anal. Appl.* 28.1 (2006), pp. 26–39. DOI: 10.1137/040617650. eprint: https://doi.org/10.1137/040617650. URL: https://doi.org/10.1137/040617650.

[9] V. Mehrmann. "A step toward a unified treatment of continuous and discrete time control problems". In: *Linear Algebra Appl.* 241–243 (1996). Proc. 4th Conf. Int. Linear Algebra Soc., pp. 749–779. ISSN: 0024-3795. DOI: https://doi.org/10.1016/0024-3795(95)00257-X. URL: https://www.sciencedirect.com/science/article/pii/002437959500257X.

[10] E. Seneta. *Non-negative Matrices and Markov Chains*. 2nd ed. Springer, 2008. ISBN: 978-0387510415.

[11] G. Strang. *Linear Algebra and Learning from Data*. 1st ed. Wellesley-Cambridge Press, 2019. ISBN: 978-0692196380.