

UNIVERSIDAD NACIONAL DE ASUNCIÓN

Facultad de Ingeniería Ingeniería Electrónica



**Desarrollo de un sistema de control para el algoritmo
GMRES(m) de resolución de sistemas lineales grandes
dispersos**

Gustavo Enrique Espínola Mena

San Lorenzo - Paraguay

2019

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Miembros del Consejo Directivo

Consejeros Titulares

Prof. Dr. Ing. Rubén Alcides López Santacruz (Decano)
Prof. Ing. Primo Antonio Cano Coscia (Vice-Decano)
Prof. MSc. Ing. Rubén Darío Zárate Rojas (Docente)
Prof. Ing. Ana María Brun de Lesme (Docente)
Prof. Ing. Francisco Manuel Arrom Irún (Docente)
Prof. Ing. Ariel Jorge Martínez Marin (Docente)
Prof. Ing. Héctor Vladimir Arce Toledo (Docente)
Prof. Ing. Helio Ricardo Moral Granada (Docente)
Ing. Martín Wilobaldo Balmaceda Pérez (Graduado)
Ing. Tania Anahí Vargas Agüero (Graduada)
Ing. Ninoska Natalia Candia Villamayor (Graduada)
Est. Sebastián Daniel Olivella Noguera (Estudiante)
Est. María Paula Oviedo Velázquez (Estudiante)
Est. Ulises Manuel Leónido González Galeano (Estudiante)

Consejeros Suplentes

Prof. Ing. Fulgencio Antonio Aquino Duarte (Docente)
Ing. Fausto Ariel Alfonso Duarte (Graduado)
Ing. José Luis González Osorio (Graduado)
Est. Federico Daniel Peralta Samaniego (Estudiante)
Est. Antonic Josué Wuyk Borja (Estudiante)
Est. Sofía Vierci Codas (Estudiante)

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

"Resolver un problema es hacer un descubrimiento. Un gran problema significa un gran descubrimiento, pero hay una partícula de descubrimiento en la solución de cualquier problema. El suyo puede ser modesto, pero si pone a prueba la curiosidad que induce a poner en juego las facultades inventivas, y si lo resuelve por medios propios, puede experimentar la tensión y el encanto del descubrimiento y el goce del triunfo."

George Pólya (1887-1985), matemático húngaro.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Agradecimientos institucionales

Al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el soporte al Proyecto de Investigación 14-INV-186: «Control de algoritmos bloque iterativos basados en el Subespacio de Krylov - CABIBESKRY».

Al Centro de Investigación en Matemática (CIMA), por el apoyo logístico para la presentación de los resultados de este Trabajo en el Congreso de Matemática Aplicada y Computacional (CNMAC) en Brasil.

Al Laboratorio de Computación Científica y Aplicada (LCCA) de la Facultad Politécnica - UNA, por el acceso a las instalaciones y el apoyo a la iniciación científica de los alumnos de grado.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Agradecimientos

A los Asesores: Dr. Christian Schaeerer, Dra. Miki Saito y MSc. Juan Carlos Cabral, por los consejos precisos para superar los obstáculos encontrados.

A toda la Comunidad de la Facultad de Ingeniería, en las sedes de San Lorenzo y CITEC - Luque, especialmente a aquellas personas de quienes he recibido un constante incentivo para alcanzar esta meta.

A mis amigos, con quienes he compartido con gusto la vida universitaria.

A Marina, por haberme escuchado y apoyado en toda esta etapa.

Agradecimiento especial a mi Familia. Definitivamente no hubiera podido llegar aquí sin ellos.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Resumen

Se diseña un sistema de control para el Algoritmo del Residuo Mínimo Generalizado con reinicios o GMRES(m) para la resolución de sistemas lineales grandes dispersos. El método propuesto combina dos estrategias: modifica la dimensión m del subespacio de búsqueda de Krylov si detecta estancamiento y lo enriquece con vectores de información de ciclos anteriores. La propuesta se aplica a la resolución de problemas de ingeniería y su rendimiento es puesto en comparación frente a otros métodos iterativos similares de la literatura científica basados en GMRES(m) con parámetros fijos. Los resultados numéricos para las matrices seleccionadas sugieren que este método podría, para algunos problemas, superar el estancamiento e incluso mejorar el rendimiento -en tiempo de ejecución y cantidad de almacenamiento en memoria- causado por un método estándar con parámetro fijo.

Palabras claves: *método iterativo, GMRES, esparcimiento electromagnético.*

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Índice general

1 Introducción	6
1.1 Motivación	6
1.2 Sistemas de ecuaciones lineales	8
1.3 Problemática	9
1.3.1 Proyecto CABIBESKRY - CONACYT	11
1.4 Originalidad y Relevancia	11
1.5 Objetivo General	12
1.6 Objetivos Específicos	13
1.7 Alcance y limitaciones	13
1.8 Organización del Trabajo	13
2 Sistemas lineales y GMRES	15
2.1 Métodos computacionales para resolución de sistemas lineales: métodos directos e iterativos	15
2.2 Matrices dispersas	16
2.3 Métodos iterativos para matrices dispersas	19
2.4 Métricas para métodos iterativos: error y residuo	21
2.5 Métodos de proyección	22
2.6 Métodos de subespacio de Krylov	23
2.7 Por qué utilizar un subespacio de Krylov	24
2.8 GMRES	25
2.8.1 Método de Arnoldi	26
2.8.2 Minimización de la norma residual	29
2.8.3 Convergencia del GMRES	30
2.9 GMRES con reinicio: GMRES(m)	30
2.10 Ventajas y desventajas del método con reinicio	32

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

3 Diseño del sistema de control para GMRES	34
3.1 Estrategias para mejorar GMRES(m)	34
3.2 Modelado del método iterativo como sistema de control	35
3.3 Formulación de control	36
3.4 GMRES(m) con parámetro m controlado	36
3.5 Propuestas de enriquecimiento de subespacio	38
3.5.1 LGMRES(m, l)	38
3.5.2 GMRES-E(m, d)	39
3.6 Propuesta del sistema de control para el GMRES(m)	39
4 Esparcimiento electromagnético	42
4.1 Electromagnetismo computacional	42
4.2 Esparcimiento electromagnético	43
4.3 Ecuaciones de Maxwell	44
4.4 Ecuaciones de Maxwell en el espacio vacío	46
4.5 Ecuación de onda para el espacio libre	47
4.6 Ecuación de Helmholtz	47
4.7 Problema de esparcimiento	48
4.8 Aspectos de diseño	51
5 Experimentos numéricos	53
5.1 Configuración inicial de los parámetros	54
5.2 Resultados y discusión	55
6 Conclusiones y trabajos futuros	61
APÉNDICE A Presentaciones nacionales e internacionales	63
APÉNDICE B Código fuente	74

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Índice de figuras

Figura 1.1: Etapas de la solución de un problema de las ciencias e ingenierías que ha sido descrito con ecuaciones diferenciales.	7
Figura 1.2: Curvas de (a) convergencia rápida y (b) convergencia lenta -estancamiento- para el método GMRES(m) con $m = 30$.	11
Figura 2.1: Problema de almacenamiento de una matriz densa y solución con una matriz equivalente dispersa.	18
Figura 2.2: Ejemplo de malla de elementos finitos 2D (izq.) y estructura de matriz dispersa asociada.	20
Figura 2.3: Interpretación de la condición de ortogonalidad.	23
Figura 2.4: Estructura de cada ciclo de reinicio del método GMRES(m).	31
Figura 3.1: Método iterativo para sistemas lineales, representado como sistema de control de lazo cerrado.	35
Figura 3.2: Diagrama de bloques para GMRES(m).	37
Figura 3.3: Diagrama de bloques para GMRES(m) adaptativo.	37
Figura 3.4: Diagrama de bloques para el sistema de control propuesto.	41
Figura 4.1: Descripción del problema de esparcimiento electromagnético.	44
Figura 4.2: Geometría del problema de esparcimiento electromagnético sobre una cavidad.	49
Figura 4.3: Geometría de la discretización en una malla rectangular.	50
Figura 4.4: Estructura del sistema lineal $Au = f$.	52
Figura 4.5: Discretización errónea de una onda debido al <i>aliasing</i> .	52

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Figura 5.1: Comparación de curvas de convergencia entre dos métodos adaptativos. Norma residual relativa ($\ r_j\ /\ r_0\ $), dimensión del subespacio de búsqueda (m_j para GMRES(m_j) y s_j para A-LGMRES-E($m_j, 1, 3$)) y norma de la variable de control $\ y_j\ $, para las matrices (a) cavity02 y (b) cavity04.	58
Figura 5.2: Curvas de convergencia para problemas cavity01 a cavity05: norma residual relativa vs. cantidad de ciclos.	59
Figura 5.3: Curvas de convergencia para problemas cavity06 y cavity07: norma residual relativa vs. cantidad de ciclos.	60
Figura A.1: Resumen sometido al CNMAC 2017, pág. 1.	64
Figura A.2: Resumen sometido al CNMAC 2017, pág. 2.	65
Figura A.3: Póster presentado al CNMAC 2017.	66
Figura A.4: Certificado de participación en el CNMAC 2017.	67
Figura A.5: Resumen sometido al CNMAC 2018, pág. 1.	68
Figura A.6: Resumen sometido al CNMAC 2018, pág. 2.	69
Figura A.7: Póster presentado al CNMAC 2018.	70
Figura A.8: Certificado de participación en el CNMAC 2018.	71
Figura A.9: Póster presentado en el II Encuentro de Investigadores de la SCP.	72
Figura A.10: Certificado de participación en el II Encuentro de Investigadores.	73

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Índice de tablas

Tabla 2.1: Cuadro comparativo entre métodos directos e iterativos	17
Tabla 5.1: Matrices resultantes de la discretización del problema de esparcimiento electromagnético	54
Tabla 5.2: Métricas para las matrices seleccionadas y métodos iterati- vos: tiempo de ejecución en segundos y cantidad de ciclos de ejecucion.	56
Tabla 5.3: Resultados para A-LGMRES-E($m_j, 3, 1$). Valores máximos, mínimos y medios de $\ y_j\ $ para cada problema de la Tabla 1 y cantidad de ciclos requeridos para que $\ r_j\ /\ r_0\ < 10^{-6}$ o cantidad máxima de 2000 ciclos.	57

CAPÍTULO 1

Introducción

1.1. Motivación

La resolución de problemas de las ciencias e ingenierías consiste de varias etapas. El modelado matemático de problemas en ingeniería a menudo pasa por un proceso de resolución numérica que valora la solución aproximada del modelo. Las etapas del proceso de solución son descritas en la figura 1.1. La solución numérica de un problema con formulación en ecuaciones diferenciales pasa por una discretización, es decir, la resolución del problema para un conjunto discreto y finito de puntos del dominio del problema [1]. El proceso de linealización hace posible analizar el comportamiento de un modelo no lineal de gran complejidad alrededor de un punto del dominio del problema utilizando solamente funciones lineales. Entonces, el conocimiento de los sistemas lineales es un requisito esencial para el análisis posterior de tales problemas no lineales que las ciencias e ingenierías enfrentan en la actualidad.

El estudio de los sistemas lineales generados en el proceso, que se resuelven usando métodos computacionales, deriva en la rama de la Matemática conocida como Álgebra Lineal Numérica. A continuación se presentan los siguientes problemas clásicos [3]:

- **Sistemas lineales de ecuaciones:** Resolver $Ax = b$. Aquí A es una matriz de orden $n \times n$, con elementos reales o complejos, b es un vector columna dado con n elementos, y x es el vector columna que deseamos calcular,

PROBLEMA DE LAS CIENCIAS EN INGENIERÍAS



Ecuaciones integro-diferenciales

- Problema de dominio infinito
- Existencia y unicidad de la solución analítica



Problema aproximado: discretización

- Aproximación de dominio finito
- Convergencia a la solución analítica



Problema del álgebra lineal: computación

- Linealización
- Computación por producto matriz-vector

Figura 1.1: Etapas de la solución de un problema de las ciencias e ingenierías que ha sido descrito con ecuaciones diferenciales. **Fuente:** [2].

también con una cantidad n de incógnitas.

- **Problemas de mínimos cuadrados:** Calcular el vector x que minimiza la norma $\|Ax-b\|_2$. Aquí, A es una matriz de orden $m \times n$, x y b son vectores de n y m elementos respectivamente, $\|y\|_2 = \sqrt{\sum |y_i|^2}$ es la norma euclídea del vector y .
- **Problemas de autovalores y autovectores:** Dada una matriz A de orden $m \times m$, hallar un escalar λ llamado *autovalor* y un vector v no nulo de orden $n \times 1$ llamado *autovector*, tales que $Av = \lambda v$.
- **Problemas de descomposición en valores singulares de matrices:** Dada una matriz A de orden $m \times n$, hallar un escalar λ y un vector v no nulo de orden $n \times 1$ tal que $A^T A v = \lambda v$.

Estos problemas son objeto de constante investigación debido a las siguen-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

tes causas [4]:

- las aplicaciones emergentes cada vez más complejas requieren cálculos eficientes con matrices cada vez más grandes, para las cuales los algoritmos existentes son muy lentos;
- las arquitecturas de los computadores evolucionan continuamente, lo que obliga a crear o modificar algoritmos para buscar la máxima eficiencia en cada una de ellas;
- los requerimientos de precisión son cada vez mayores;
- siguen apareciendo constantemente nuevas clases de matrices estructuradas para las cuales se debe intentar desarrollar algoritmos más específicos y/o precisos que los existentes.

La idea central, entonces, del estudio del Álgebra Lineal Numérica es «aprovechar la estructura de las clases de matrices que aparecen en los problemas a fin de desarrollar algoritmos específicos para dichas matrices» [4].

1.2. Sistemas de ecuaciones lineales

Los sistemas de ecuaciones lineales son de particular interés para la resolución de diferentes problemas científicos y tecnológicos. Estas ecuaciones son de la forma

$$Ax = b, \quad (1.1)$$

donde $A \in \mathbb{C}^{n \times n}$ es la matriz de coeficientes, $x \in \mathbb{C}^n$ es el vector columna que contiene las incógnitas y $b \in \mathbb{C}^n$ es el vector columna de los términos independientes.

Al resolver un sistema de ecuaciones lineales $Ax = b$ proveniente de diferentes aplicaciones de las ciencias e ingenierías, se asume que la matriz de los

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

coeficientes A es una matriz no singular, ya que se espera que el vector incógnita x devuelva los valores numéricos de las magnitudes físicas del problema de ingeniería en estudio. Por lo tanto, el método computacional que resuelva el sistema lineal debe devolver una solución similar a la solución exacta del sistema $x^* = A^{-1}b$. Sin embargo, en la práctica se presentan problemas computacionales cuando A guarda características que la hacen «casi singular». Entonces, se busca un método que calcule un vector próximo a x^* sin calcular explícitamente la matriz inversa.

1.3. Problemática

Este trabajo busca implementar un método numérico de resolución de sistemas lineales grandes dispersos en problemas de ingeniería. En el proceso de diseño de estos métodos, se debe tener en cuenta los siguientes requisitos:

- **Tiempo de ejecución.** En principio el problema de un sistema lineal de ecuaciones podría resolverse con el algoritmo de eliminación gaussiana, el método más sencillo y directo a la hora de aplicar métodos computacionales. Sin embargo, el costo en tiempo de ejecución para este algoritmo es demasiado alto cuando n , el tamaño del sistema lineal, es muy grande. Esto se debe a que el tiempo de ejecución de los métodos numéricos es directamente proporcional a la cantidad de operaciones aritméticas (sumas, restas, multiplicaciones, divisiones) que cada método numérico necesita, que a su vez depende de la dimensión y estructura de la matriz de coeficientes. Para la eliminación gaussiana, por ejemplo, se requiere aproximadamente $2n^3/3$ operaciones aritméticas [5]. Este conteo de operaciones es un buen estimador de la velocidad a la que el tiempo de ejecución crece a medida que el tamaño del problema crece.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

- **Rellenado y almacenamiento en memoria.** Otro método numérico habitualmente elegido es el de factorización LU, por presentar la ventaja de reducción de operaciones frente al método de eliminación gaussiana. Sin embargo, el método requiere la modificación de la matriz inicial, en particular, propicia el **rellenado** (*fill-in*, en inglés) de matrices dispersas (es decir, matrices cuya cantidad de elementos no nulos es muy pequeña en comparación con su cantidad de elementos nulos). Este fenómeno produce un aumento considerable en la cantidad de **memoria requerida** para almacenar las matrices resultantes. Así, una matriz dispersa podría cumplir los requisitos de memoria antes de la aplicación del método numérico, pero no durante su ejecución.
- **Convergencia y estancamiento.** Los métodos numéricos iterativos fueron diseñados para el ahorro de los recursos computacionales antes citados (tiempo de ejecución y memoria requerida) porque no modifican el tamaño de la matriz de coeficientes, pero en general no se asegura la **convergencia**, aquí definida como la característica según la cual el método elegido llega a la solución del problema en tiempo finito para todos los tipos de problemas. El fenómeno contrario se conoce como **estancamiento**: la imposibilidad de alcanzar la solución del problema en tiempo finito. En la figura 1.2 se observa un problema de estancamiento. Una medida habitual consiste en tomar la norma del residuo $\|r\| = \|b - Ax\|$ en cada iteración del método. Si se presenta estancamiento, $\|r\|$ no disminuye pese al aumento en la cantidad de iteraciones del método.

En resumen, la idea central del trabajo es desarrollar un método iterativo para resolución de sistemas lineales que almacene la menor cantidad posible de elementos de la matriz de coeficientes, se ejecute en menor tiempo y sea lo suficientemente robusto para garantizar la solución de todos los problemas en

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

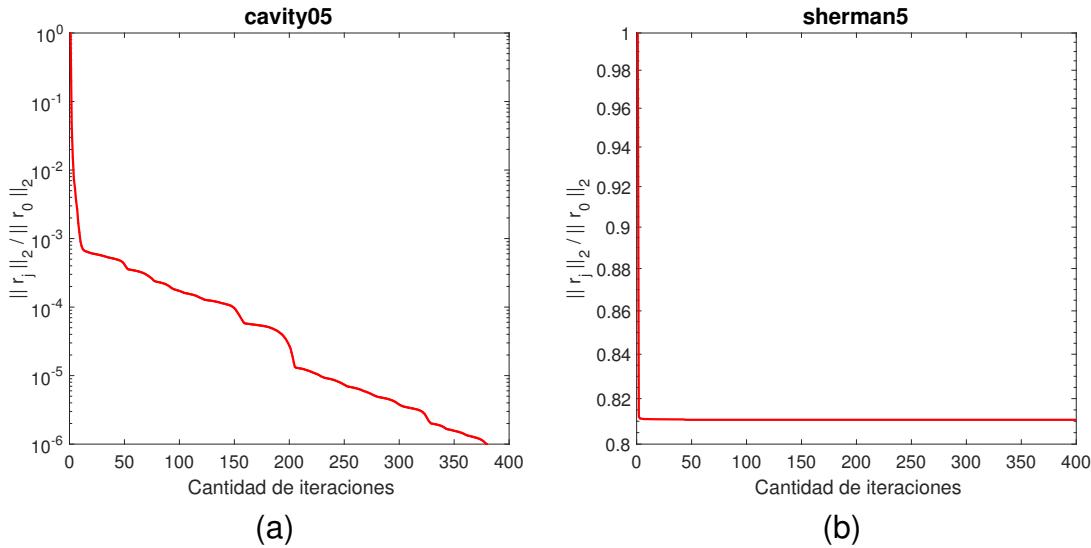


Figura 1.2: Curvas de (a) convergencia rápida y (b) convergencia lenta - estancamiento- para el método GMRES(m) con $m = 30$. **Fuente:** [6].

comparación con otros métodos de la literatura científica.

1.3.1. Proyecto CABIBESKRY - CONACYT

A fin de investigar los problemas recién citados, este trabajo formó parte del Proyecto de Investigación «Control de Algoritmos Bloque Iterativos Basados En Subespacio de KRYlov» (CABIBESKRY), ejecutado por el Centro de Investigación en Matemática (CIMA), con soporte financiero del Consejo Nacional de Ciencia y Tecnología (CONACYT). El Objetivo del Proyecto fue: «proponer una ley de control para métodos iterativos basados en el Sub-Espacio de Krylov [...]» [7]. Dos tipos de algoritmos son considerados: 1) los que presentan mejor eficiencia para sistemas simétricos positivos definidos y 2) los que presentan mejor eficiencia para sistemas generales, posiblemente indefinidos y mal condicionados. En este trabajo nos centraremos en estos últimos.

1.4. Originalidad y Relevancia

Este trabajo evalúa la convergencia del algoritmo RESiduo Mínimo Generalizado con reinicios, o GMRES(m) (*Generalized Minimal RESidual* en inglés) [8]

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

de resolución de sistemas lineales grandes dispersos, mediante la aplicación de métodos de la Teoría de Control, cuyas ideas han sido ampliamente experimentadas para sistemas físicos, pero no suficientemente reconocidas en el ámbito del desarrollo de software. En el artículo de A. Bhaya y E. Kaszkurevicz [9] se explica que un método iterativo puede ser representado como un sistema de control realimentado, con una planta y un controlador. La planta es el objeto a ser controlado, de la forma

$$x_k = x_{k-1} + u_k \quad (1.2)$$

donde x_k es la aproximación a la solución del sistema lineal en la iteración k , igual a la suma entre la aproximación de la iteración anterior x_{k-1} más una variable controlada u_k . La elección y modificación del algoritmo corresponde con la elección del controlador.

Como punto de partida, los conceptos desarrollados en los trabajos «Una estrategia inspirada en control para variar el parámetro de reinicio m de GMRES(m)» de Cuevas [10] y «Estrategia de control para GMRES(m) aumentado con vectores armónicos de Ritz» de Cabral [11] son explorados, con la novedad de la combinación de los métodos allí descritos. Además, la resolución óptima del sistema lineal subyacente al problema elegido de esparcimiento de ondas propone una alternativa a los trabajos previos «Formulación de Control para el Problema de Esparcimiento de Helmholtz» de I. Ortiz [12] y «Algoritmo Parareal en el estudio de Ondas Electromagnéticas - 2D» de E. De Los Santos y J. Gavilán [13].

1.5. Objetivo General

Desarrollar un sistema de control con los valores óptimos de los parámetros del algoritmo Residuo Mínimo Generalizado con reinicios o GMRES(m).

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

1.6. Objetivos Específicos

- Estudiar métodos iterativos para sistemas lineales grandes y dispersos.
- Analizar las ventajas y desventajas del método iterativo elegido.
- Analizar los diferentes mecanismos de control del algoritmo GMRES(m).
- Desarrollar el sistema de control para el algoritmo GMRES(m) en un lenguaje de programación.
- Ejecutar el algoritmo GMRES(m) bajo parámetros de control modificados.
- Medir los tiempos de CPU y los números de iteraciones del algoritmo y comparar con métodos del Estado del Arte.

1.7. Alcance y limitaciones

Se busca probar los métodos elegidos con matrices de un problema de esparcimiento electromagnético [14], para las cuales se ha comprobado que poseen problemas de estancamiento o convergencia lenta. Las matrices generadas son dispersas, no hermitianas.

1.8. Organización del Trabajo

En el Capítulo 2 se presentan las características más importantes de los métodos numéricos de resolución de sistemas de ecuaciones lineales, de los métodos de subespacio de Krylov en general, y del método elegido GMRES en particular. Posteriormente, se hace una descripción detallada de la modificación de GMRES con reinicio, denominada GMRES(m). Se describen las estrategias para mejorar el GMRES(m), luego se presenta la estrategia de control que permite acelerar la convergencia. Se presenta la metodología propuesta en este trabajo y los diferentes métodos que permitirán compararla.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Si bien el método es transversal a cualquier problema de ingeniería, una aplicación en Electromagnetismo fue elegida para el testeo del método, la descripción se puede leer en el Capítulo 4. Se enuncia un problema de esparcimiento de onda electromagnética descrito en el artículo científico «*GMRES with Adaptive Deflated Restarting and its performance on an EM cavity problem*» de K. Du [14], y se explica, paso a paso, su discretización como sistema lineal. Los resultados experimentales son presentados en el Capítulo 5, el cual expone las métricas recolectadas sobre los sistemas lineales del problema de esparcimiento electromagnético.

Las conclusiones y trabajos futuros se presentan en el Capítulo 6. Finalmente, se incluyen Apéndices con las presentaciones del trabajo en eventos de iniciación científica nacionales e internacionales en el marco del Proyecto CABIBESKRY.

CAPÍTULO 2

Sistemas lineales y GMRES

2.1. Métodos computacionales para resolución de sistemas lineales: métodos directos e iterativos

Los sistemas lineales $Ax = b$ pueden ser resueltos con **métodos directos** o **métodos iterativos**. Los **métodos directos** son aquellos tales que, suponiendo que no existan errores de aritmética finita, permiten calcular la solución exacta del sistema en una cantidad finita de operaciones [15], la mayoría de las cuales modifican la matriz A original: por ejemplo, el reordenamiento de los elementos de la matriz A original (ej., la eliminación gaussiana) o la descomposición en factores de A (ej., LU o QR). Las técnicas directas, en general, poseen una cantidad de operaciones fija desde el comienzo: no se puede detener su ejecución hasta obtener la solución del sistema. Si el tamaño de la matriz es muy grande, tal tiempo de ejecución es del orden del cubo del tamaño de la matriz [16]. Como otra desventaja, los métodos directos presentan un grado de paralelismo bajo lo que limita su escalabilidad en sistemas paralelos [17]. Además, los métodos directos no son apropiados para resolver sistemas lineales con una cantidad muy grande de ecuaciones, en especial si la matriz de coeficientes resultante tiene la mayor parte de sus elementos iguales a cero, porque necesitan que toda la información de la matriz sea almacenada, y esta ocupa grandes cantidades de espacio en memoria.

En comparación, los **métodos iterativos** son aquellos tales que generan una

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

sucesión de vectores $\{x_i\}, i = 0, 1, 2, \dots$ que se aproximan a la solución del sistema en cada iteración del método [18]. En otras palabras, la iteración consiste en realizar una cantidad finita de cálculos aritméticos basados en un vector inicial x_0 , hasta devolver una solución aproximada x_1 , que a su vez es dato para estimar la siguiente aproximación a la solución x_2 , y así sucesivamente. Es posible determinar la ejecución de un método iterativo y así encontrar una solución aproximada al sistema lineal. Estos métodos no modifican la estructura de la matriz dada, ya que su principal operación es el producto de matriz por vector [17]. Como desventaja, los métodos iterativos no son exactos: como una sucesión de vectores puede tener infinitos términos, el diseño de un método iterativo debe incluir un criterio de detención para garantizar que se ejecute en tiempo finito, habitualmente cuando el enésimo vector de aproximación x_n devuelve un error de cómputo menor que un umbral de tolerancia, para el cual los errores de discretización y de redondeo de la aritmética finita son despreciables. Los métodos iterativos son habitualmente la única solución para ecuaciones no lineales, pero también son muy utilizados para sistemas lineales con una cantidad muy grande de incógnitas (a veces en el orden de millones), ya que los métodos directos pueden ser muy costosos incluso para la computadora más potente disponible. Otra ventaja de los métodos iterativos es que se puede aprovechar la estructura dispersa de la matriz A, cuya característica veremos a continuación.

Finalmente, un resumen de las características de ambos métodos se presenta en la tabla 2.1.

2.2. Matrices dispersas

No siempre las matrices asociadas a un sistema de ecuaciones lineales son matrices completamente llenas de valores numéricos significativos: en muchas ocasiones una gran parte de sus elementos son iguales a cero, por lo tanto estos elementos no tienen un papel muy relevante en los métodos numéricos ya que

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Tabla 2.1: Cuadro comparativo entre métodos directos e iterativos

Métodos directos	Métodos iterativos
Solución exacta	Solución aproximada
Exactitud depende de la computadora	Exactitud depende del método
Costo elevado en memoria y tiempo CPU	Costo reducido en memoria y (posiblemente) tiempo CPU
Difícil de implementar en sistemas heterogéneos y/o paralelos	Orientados a sistemas heterogéneos y/o paralelos

Fuente: Elaboración propia.

el resultado de operar cualquier número con un cero se conoce de antemano. Un buen tratamiento de estas matrices permite ahorrar espacio en memoria y tiempo de ejecución en los métodos. Siempre que sea posible utilizar métodos alternativos de almacenamiento y de solución para explotar la estructura de una matriz, se podrá mejorar el rendimiento de un método para resolver sistemas lineales [19].

Debido a ello, las matrices de los coeficientes son habitualmente clasificadas en matrices dispersas o densas. Una **matriz dispersa** (*sparse*, en inglés) es una matriz, habitualmente de gran tamaño« que permite técnicas especiales para tomar ventaja del gran número de elementos iguales a cero» [20]. En contraste, las matrices llenas de elementos no nulos se llaman **matrices densas**. Un parámetro descriptivo muy importante de una matriz dispersa A es la cantidad de elementos no nulos, a obtener en Matlab con el comando *nnz(A)* (*number of non-zero elements*, en inglés).

Cuando se almacenan y operan matrices dispersas en una computadora, es beneficioso y a menudo necesario utilizar algoritmos especializados y estructuras de datos que aprovechen la estructura dispersa de la matriz. Las operaciones que usan estructuras y algoritmos de matriz densa estándar son lentas e inefficientes cuando se aplican a matrices dispersas grandes ya que el procesamiento y la memoria se desperdician en los ceros. Las consideraciones de memoria pa-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

ra matrices dispersas radican en que los elementos no nulos son más fáciles de guardar en una estructura de datos apropiada, por tanto requieren un almacenamiento significativamente menor. La figura 2.1 muestra las limitaciones de tratar de almacenar todos los elementos de una matriz pese a contener una cantidad mínima de elementos no nulos, ej., una matriz tridiagonal, y la solución al problema con su matriz equivalente dispersa que almacena sólo los elementos no nulos: mientras que una matriz densa necesitaría 7450 GB para ser almacenada, la matriz dispersa requiere unos 55 MB.

```
>> n = 5;
>> A1 = full(gallery('tridiag', n, 1, -2, 1))

A1 =

-2     1     0     0     0
 1    -2     1     0     0
 0     1    -2     1     0
 0     0     1    -2     1
 0     0     0     1    -2

>> n = 1000000;
>> A2 = full(gallery('tridiag', n, 1, -2, 1))
Error using full
Requested 1000000x1000000 (7450.6GB) array exceeds maximum
array size preference. Creation of arrays greater than this
limit may take a long time and cause MATLAB to become
unresponsive. See array size limit or preference panel for more
information.

>> A = sparse(gallery('tridiag', n, 1, -2, 1));
>> whos A
  Name            Size           Bytes  Class       Attributes
  A            1000000x1000000      55999976  double    sparse
```

Figura 2.1: Problema de almacenamiento de una matriz densa y solución con una matriz equivalente dispersa. **Fuente:** Elaboración propia.

La solución para representar y trabajar con matrices dispersas es utilizar una estructura de datos alternativa para representar los datos dispersos. Los elementos nulos se pueden ignorar y solo los datos distintos de cero en la matriz dispersa deben almacenarse o aplicarse al método. Existen algunas estructuras de

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

datos que pueden ser utilizadas para almacenar de manera eficiente una matriz dispersa; a continuación se enumeran tres ejemplos comunes [21; 22; 23].

- Diccionario de llaves. Se usa un diccionario donde un índice de fila y columna se asigna a un valor.
- Lista de listas. Cada fila de la matriz se almacena como una lista, con cada sublistas que contiene el índice de la columna y el valor.
- Lista de coordenadas. Se almacena una lista de estructuras, donde cada estructura contiene las coordenadas del elemento (el índice de fila y el índice de columna) y el valor.

También hay estructuras de datos que son más adecuadas para realizar operaciones eficientes; dos ejemplos de uso común se enumeran a continuación:

- Fila dispersa comprimida o CSR (*compressed sparse row*). La matriz dispersa se representa utilizando tres matrices unidimensionales para los valores distintos de cero, las extensiones de las filas y los índices de columna.
- Columna dispersa comprimida o CSC (*compressed sparse column*). Igual que el método de la fila dispersa comprimida, excepto que los índices de la columna se comprimen y se leen primero antes que los índices de la fila.

Las estructuras de fila o columna dispersa comprimida se utilizan a menudo para representar matrices dispersas dada la eficiencia en el acceso a memoria y la multiplicación de matrices. Por defecto, MATLAB utiliza la estructura CSC.

2.3. Métodos iterativos para matrices dispersas

La idea natural de tomar ventaja de la gran cantidad de ceros de una matriz dispersa fue generada al intentar la resolución de sistemas provenientes de

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

la ingeniería. En muchos problemas de sistemas eléctricos de potencia, cálculo de estructuras, discretización de ecuaciones diferenciales, dimensionamiento de redes en general, etc., es una característica común la existencia de una gran cantidad de elementos que interactúan apenas con una cantidad limitada de ellos entre sí. Un ejemplo se puede observar en la figura 2.2: se muestra la estructura de una matriz dispersa asociada con el problema de la retícula de elementos finitos. Así, los sistemas lineales provenientes de aplicaciones de ciencias e ingenierías son generalmente dispersas y de estructura predecible.

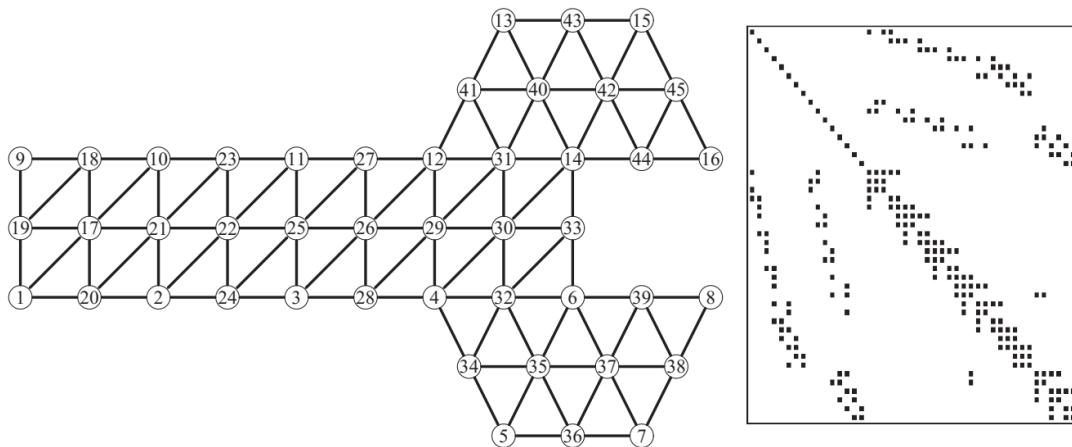


Figura 2.2: Ejemplo de malla de elementos finitos 2D (izq.) y estructura de matriz dispersa asociada. **Fuente:** [21].

Uno de los métodos directos más utilizados para sistemas lineales es la eliminación gaussiana. Este método requiere, para matrices densas, el almacenamiento de los n^2 elementos de la matriz de coeficientes y aproximadamente $2n^3/3$ operaciones aritméticas para todo el proceso [16]. Por el contrario, la multiplicación matriz-vector puede tomar ventaja de la estructura ordenada de la matriz dispersa. Esto se debe a que, si una matriz de orden $n \times n$ cuenta con sólo k elementos no nulos por fila ($k \ll n$), entonces la multiplicación matriz-vector requiere solamente el almacenamiento de kn elementos de la matriz de coeficientes, más kn operaciones de coma flotante, cantidad significativamente menor que las $2n^2$ operaciones que requeriría la multiplicación de una matriz densa por un vector.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

En conclusión, si la multiplicación matriz-vector se puede realizar rápidamente, y si se pueden almacenar los elementos no nulos de manera bien estructurada, entonces los métodos iterativos pueden ser más rápidos y requerir menos almacenamiento que la eliminación gaussiana u otros métodos para resolver $Ax = b$.

2.4. Métricas para métodos iterativos: error y residuo

A continuación se definen las siguientes notaciones para métodos iterativos de resolución de sistemas de ecuaciones lineales dentro de una iteración k :

- La solución aproximada del sistema: x_k .
- La solución exacta del sistema, en el caso de que A sea invertible: $x^* = A^{-1}b$.
- El error, vector diferencia entre la solución exacta y la solución aproximada:
$$e_k = A^{-1}b - x_k.$$
- El residuo, vector diferencia entre los dos miembros de la igualdad 1.1:
$$r_k = b - Ax_k.$$

Cuando se utiliza un sistema lineal es «resuelto» usando métodos computacionales con aritmética de precisión finita, el implementador del método debe ser consciente de no haber encontrado la solución exacta en ninguna iteración. Entonces, es importante tener una idea bien acabada de cuál es la mejor solución aproximada x_k a fin de implementar un criterio para detener la ejecución del algoritmo de resolución. Como el cálculo de la matriz inversa A^{-1} generalmente es prohibitivo, no se puede calcular la solución exacta, por lo tanto el error e_k se desconoce. Sin embargo, como existe una relación de dependencia entre el error

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

e_k y el residuo r_k :

$$r_k = b - Ax_k = A(A^{-1}b - x_k) = Ae_k \Rightarrow \|r_k\| \leq \|A\|\|e_k\|, \quad (2.1)$$

donde $\|A\|$ es una norma matricial inducida, resulta que la norma residual $\|r_k\|$ puede ser utilizada para estimar error:

$$\|e_k\| \leq \epsilon_0 \Rightarrow \|r_k\| \leq \|A\|\|e_k\| \leq \|A\|\epsilon_0 = \epsilon \quad (2.2)$$

Esta última afirmación, $\|r_k\| \leq \epsilon$, es utilizada en los métodos computacionales iterativos como criterio de detención a fin de que el algoritmo se ejecute en tiempo finito al hacer que la norma residual sea inferior a un umbral de tolerancia $\epsilon > 0$. El siguiente paso en el diseño de un método iterativo consiste en generar una sucesión decreciente de normas residuales.

2.5. Métodos de proyección

El método de proyección para resolver sistemas de ecuaciones lineales es un método iterativo que extrae una solución x_k de un subespacio de \mathbb{C}^n . Sea \mathcal{K} este **subespacio de búsqueda** que contiene a las soluciones aproximadas, a la par que garantiza la reducción de la norma residual.

En general, se hace que el residuo r_k sea ortogonal a un conjunto finito de vectores linealmente independientes. Esto define el **subespacio de restricciones** \mathcal{L} como el conjunto de combinaciones lineales de tales vectores. La condición de ortogonalidad impuesta sobre el nuevo resíduo $r_k = r_0 - A\delta$ se ilustra en la figura 2.3, donde puede observarse que $\|r_k\| \leq \|r_0\|$. Las restricciones se conocen como condiciones de Petrov-Galerkin [21].

En el ejemplo anterior se describe un paso básico de proyección. En general, se requiere una sucesión de tales proyecciones, con un par de subespacios diferente en cada iteración, a fin de crear una sucesión monótona decreciente de la

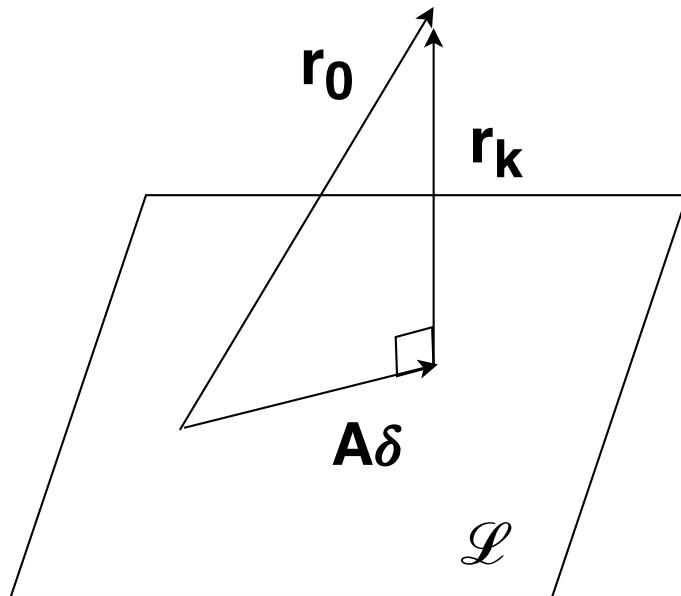


Figura 2.3: Interpretación de la condición de ortogonalidad. **Fuente:** [21]

norma residual.

2.6. Métodos de subespacio de Krylov

Las observaciones vistas en la sección 2.2 derivan en la pregunta de si es posible, con cierto grado de aproximación, diseñar un método iterativo que aproveche las propiedades de las matrices dispersas, la multiplicación matriz-vector y la adición de una cantidad de reglas de control que sea mucho menor que la cantidad de operaciones de un método directo. Si el diseño cumple estos requisitos, el método iterativo resultante será superior al método directo en términos de tiempo de ejecución y memoria requerida.

Un método que cumple con estas características es el **método de subespacio de Krylov**¹, que busca soluciones aproximadas $x_k, k = 1, 2, \dots$ en cada iteración del método, de acuerdo con la expresión:

$$x_k = x_0 + z_k, z_k \in \mathcal{K}^{(k)}(A, v) := \text{span}\{v, Av, A^2v, \dots, A^{k-1}v\}, \quad (2.3)$$

¹Alekséi Nikoláyevich Krylov (1863-1945), ingeniero naval y matemático ruso.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

donde $\mathcal{K}^{(k)}(A, v)$ es el subespacio de Krylov de dimensión k asociado a la matriz A y al vector v [11]. En la Teoría de Control, este subespacio se conoce como «subespacio de controlabilidad» [16; 24].

Si el subespacio 2.3 no contiene una buena solución aproximada para cualquier valor de k suficientemente pequeño, o si una solución aproximada no puede ser computada fácilmente, entonces se requiere la modificación del método original para obtener otro subespacio de Krylov con mejores propiedades.

2.7. Por qué utilizar un subespacio de Krylov

La solución de un sistema lineal no singular pertenece a un subespacio de Krylov cuya dimensión es el grado del polinomio mínimo² de la matriz A . Por lo tanto, si el polinomio mínimo de A tiene bajo grado entonces el subespacio a construir puede también ser pequeño. En este caso el método de subespacio de Krylov tiene la oportunidad de alcanzar una convergencia rápida [25].

Sea A de orden $n \times n$ y sean $\lambda_1, \lambda_2, \dots, \lambda_k$ los autovalores no repetidos. Sea m_j el índice de λ_j , el tamaño del mayor bloque de Jordan asociado a λ_j . Su polinomio mínimo es:

$$q(t) = \prod_{j=1}^k (t - \lambda_j)^{m_j} \quad (2.4)$$

y su grado correspondiente es $m = \sum_{j=1}^k m_j$.

Efectuando los productos de 2.4 y usando los coeficientes α_j obtenidos, podemos escribir:

$$q(t) = \sum_{j=1}^m \alpha_j t^j \quad (2.5)$$

Por el Teorema de Cayley-Hamilton, A satisface $q(A) = 0$, así:

$$-\alpha_0 I = \alpha_1 A + \alpha_2 A^2 + \cdots + \alpha_m A^m$$

²El polinomio mínimo $q(t)$ de A es un polinomio mónico con grado mínimo que se construye a partir de los autovalores de A .

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

$$\begin{aligned} -\alpha_0 I &= A(\alpha_1 + \alpha_2 A + \cdots + \alpha_m A^{m-1}) \\ -A^{-1}\alpha_0 I &= A^{-1}A(\alpha_1 + \alpha_2 A + \cdots + \alpha_m A^{m-1}) \\ A^{-1} &= -\frac{1}{\alpha_0} \sum_{j=0}^{m-1} \alpha_{j+1} A^j \end{aligned} \quad (2.6)$$

Multiplicando 2.6 por el vector b del sistema obtenemos:

$$A^{-1}b = -\frac{1}{\alpha_0} \sum_{j=0}^{m-1} \alpha_{j+1} A^j b, \quad (2.7)$$

es decir, la solución del sistema $x^* = A^{-1}b$ está como combinación lineal de los vectores del subespacio de Krylov.

Cabe aclarar que si la matriz A es singular y el sistema lineal tiene soluciones, estas podrían no estar en el subespacio de Krylov [25], pero este fenómeno no entra en estudio en este Trabajo.

2.8. GMRES

El método **GMRES** (de *Generalized Minimal RESidual*, o Residuo Mínimo Generalizado) fue presentado por Y. Saad y M. Schultz en 1986 [8]. Este método es una generalización del algoritmo MINRES (de *MINimal RESidual*), presentado por C. Paige y M. Saunders para resolver sistemas lineales cuya matriz es simétrica (hermética si la matriz es compleja). El algoritmo GMRES resuelve de manera más apropiada los sistemas lineales no simétricos (no hermíticos en el caso de sistemas con entradas complejas).

A continuación, se describe el método GMRES: para facilitar la notación se asume que la matriz A y los vectores implicados en el método tienen componentes reales, sin embargo los resultados son generalizables para matrices complejas. Al inicio del método, se define una aproximación inicial de la solución, x_0 ; un **residuo inicial** $r_0 = b - Ax_0$; y el subespacio de Krylov asociado a la matriz A y

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

al vector r_0 :

$$\mathcal{K}^{(k)}(A, r_0) = \text{span}\{r_0, Ar_0, A^2r_0, \dots, A^{k-1}r_0\}, \quad (2.8)$$

en el cual se busca la solución aproximada x_k . A menos que se diga lo contrario, el subespacio de Krylov se denominará simplemente \mathcal{K} . En la k -ésima iteración, el método GMRES devuelve una solución de la forma:

$$x_k = x_0 + z_k, \quad (2.9)$$

donde z_k es la combinación lineal de los vectores base del subespacio \mathcal{K} . Para generar z_k , el método construye una base ortogonal de \mathcal{K} a través del algoritmo de Gram-Schmidt modificado, cuya variante aplicada en este trabajo es conocida como Método de Arnoldi [5]. Se define V_k , una matriz de orden $n \times k$ cuyos vectores columna contienen esa base. La solución aproximada al final de la k -ésima iteración se calcula mediante la relación:

$$x_k = x_0 + V_k y_k \quad (2.10)$$

donde y_k es un vector que contiene los coeficientes de la combinación lineal de los vectores columna de V_k . El método GMRES está diseñado de modo tal que se logra minimizar la norma residual:

$$\|r_k\| = \min_{x_k \in x_0 + \mathcal{K}^{(k)}(A, r_0)} \|b - Ax_k\|. \quad (2.11)$$

2.8.1. Método de Arnoldi

El Método de Arnoldi³ es un algoritmo utilizado para extraer información de una matriz A cuando sólo se cuenta con una subrutina para calcular multiplicaciones matriz-vector [3; 10]. Para explicar su funcionamiento para matrices

³Walter E. Arnoldi (1917 - 1995), ingeniero estadounidense.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

reales (pero ampliable a matrices complejas), se define $K = [y_1, y_2, \dots, y_k]$, una matriz de orden $n \times k$ donde $y_1 = r_0$, $y_2 = Ar_0 = Ay_1$, $y_3 = A^2r_0 = Ay_2$, ..., $y_k = A^{k-1}r_0 = Ay_{k-1}$ son vectores columna que pertenecen al subespacio de Krylov relativo a la matriz A y al vector r_0 . Al resolver el producto de matrices AK se obtiene:

$$AK = [Ay_1, Ay_2, \dots, Ay_k] = [y_2, y_3, \dots, A^ky_1]. \quad (2.12)$$

Se puede observar que las $k - 1$ columnas a contar por la izquierda de la matriz AK son idénticas a las $k - 1$ columnas a contar por la derecha de la matriz K . Suponiendo que la matriz K sea invertible (para calcular el vector $c = -K^{-1}A^ky_1$) y recordando la propiedad del Álgebra Lineal, de que la multiplicación de una matriz M por un vector unitario canónico e_i devuelve la i -ésima columna de la matriz M [20], se tiene la relación:

$$AK = K[e_2, e_3, \dots, e_k, -c] = KC, \text{ o:} \quad (2.13)$$

$$K^{-1}AK = C = \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_1 \\ 1 & 0 & \cdots & 0 & -c_2 \\ 0 & 1 & \cdots & \vdots & -c_3 \\ \vdots & 0 & \cdots & \vdots & \vdots \\ \vdots & \vdots & \cdots & 0 & \vdots \\ \vdots & \vdots & \cdots & 1 & -c_k \end{bmatrix} \quad (2.14)$$

donde C es una matriz especial llamada **matriz de Hessenberg**. Ahora, se reemplaza K con una matriz ortogonal V tal que los k primeros vectores columna de K y V generen el mismo subespacio de Krylov. El primer vector columna es el residuo inicial normalizado: $v_1 = r_0/\beta$, donde $\beta = \|r_0\|$. En contraste con la matriz K , que podría ser difícil de invertir, la matriz V estará mejor condicionada.

Sea $K = VW$, donde V es matriz ortogonal y W es matriz triangular. Enton-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

ces:

$$K^{-1}AK = (W^{-1}V^{-1})A(VW) = C. \quad (2.15)$$

Por propiedad de matriz ortogonal, $V^{-1} = V^*$ [20] (*), luego:

$$V^*AV = WCW^{-1} = H. \quad (2.16)$$

Ya que W y W^{-1} son ambas matrices triangulares superiores y C es una matriz de Hessenberg, se puede afirmar que $H = WCW^{-1}$ también es una matriz de Hessenberg. Para el método GMRES, se define $V = [v_1, v_2, \dots, v_k]$ donde v_i es el i -ésimo vector columna de V . Como $V^TAV = H$, por la propiedad (*) se tiene $AV = VH$, de ahí:

$$Av_j = \sum_{i=1}^{j+1} h_{i,j} v_i. \quad (2.17)$$

Como los vectores v_i son ortonormales, podemos multiplicar ambos lados de la igualdad 2.17 por v_m^T para obtener:

$$v_m^T A q_j = \sum_{i=1}^{j+1} h_{i,j} v_m^T v_i = h_{m,j} \text{ para } 1 \leq m \leq j. \quad (2.18)$$

y así finalmente obtener:

$$v_{j+1} = \frac{Av_j - \sum_{i=1}^j h_{i,j} v_i}{h_{j+1,j}} \quad (2.19)$$

El Método de Arnoldi puede ser resumido como sigue:

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T, \text{ o también: } AV_k = V_{k+1} \tilde{H}_k \quad (2.20)$$

donde:

$$\tilde{H}_k = \begin{bmatrix} & H_k \\ 0, \dots, 0, h_{k+1,k} \end{bmatrix} \quad (2.21)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

es una matriz rectangular de orden $(k + 1) \times k$ compuesta por la matriz H_k , y una $(k + 1)$ -ésima fila cuyo único elemento no nulo es $h_{k+1,k}$. Al multiplicar por la izquierda de ambos miembros de la ecuación 2.20 por $V_k^T = V_k^{-1}$ resulta:

$$V_k^T A V_k = V_k^T V_k H_k + V_k^T h_{k+1,k} v_{k+1} e_k^T, \text{ o:} \quad (2.22a)$$

$$V_k^* A V_k = H_k \quad (2.22b)$$

donde el término $V_k^* h_{k+1,k} v_{k+1} e_k^T$ se anula porque las columnas de V_k forman una base ortonormal.

2.8.2. Minimización de la norma residual

Sea $z_k = V_k y_k$ el vector combinación lineal de los vectores V_k , base del subespacio de Krylov. Para conocer las propiedades de y_k :

$$\begin{aligned} r_k &= b - Ax_k = b - A(x_0 + z_k) = (b - Ax_0) - Az_k \\ &= r_0 - Az_k = r_0 - AV_k y_k = (\text{por Arnoldi:}) \\ &= r_0 - V_{k+1} \tilde{H}_k y_k = \beta v_1 - V_{k+1} \tilde{H}_k y_k \\ &= \beta V_{k+1} e_1 - V_{k+1} \tilde{H}_k y_k = V_{k+1} (\beta e_1 - \tilde{H}_k y_k). \end{aligned} \quad (2.23)$$

Finalmente y como la norma matricial inducida $\|V_{k+1}\| = 1$ por ser V_{k+1} una matriz ortonormal, el problema de minimización de la norma residual descrito en la ecuación 2.11 se modifica:

$$\|r_k\| = \min_{x_k \in x_0 + \mathcal{K}_k(A, r_0)} \|\beta e_1 - \tilde{H}_k y_k\|, \quad (2.24)$$

problema de mínimos cuadrados. El problema 2.24 es más fácil de resolver que el 2.11 porque se requiere solamente una matriz de orden $(k + 1) \times k$, apropiado siempre que el valor de k no sea muy elevado.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Algoritmo GMRES

```

1: procedure GMRES( $A, b$ ) ▷ Resolver el sistema  $Ax = b$ 
2:   Entrada: Elegir  $x_0$ , calcular  $r_0 = b - Ax_0$ ,  $\beta = \|r_0\|$  y  $v_1 = r_0/\beta$ .
3:   Salida: Solución del sistema lineal  $Ax = b$ .
4:   for  $j = 1, 2, \dots, k$  do
5:     Calcular  $h_{ij} = \langle Av_j, v_i \rangle$  para  $i = 1, 2, \dots, j$ 
6:     Calcular  $\tilde{v}_{j+1} = Av_j - \sum_{i=1}^j h_{ij}v_i$ 
7:     Calcular  $h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$ 
8:     Calcular  $v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$ 
9:   end for
10:  Hallar el  $y_k$  = vector  $y$  que minimice  $\|\beta e_1 - \tilde{H}_k y\|$ 
11:  Formar la solución  $x_k = x_0 + V_k y_k$ 

```

2.8.3. Convergencia del GMRES

Una importante propiedad del método GMRES es que las normas residuales de cada iteración forman una sucesión monótona no creciente. Esto es:

$$\|r_j\| \leq \|r_{j-1}\| \tag{2.25}$$

En aritmética exacta, el método GMRES clásico (basado en el método de Arnoldi) para matrices no simétricas/no hermíticas converge a la solución exacta de $Ax = b$ cuando calcula una base V_j con tantos vectores columnas como la dimensión del sistema lineal mismo, es decir, en la iteración $j = n$ [8]. En la práctica, se desea evitar tener que trabajar con la dimensión máxima del subespacio de \mathbb{C}^n . El único caso de interés es aquel en el cual GMRES alcanza la convergencia en una cantidad de iteraciones m mucho menor que la dimensión del problema ($m \ll n$).

2.9. GMRES con reinicio: GMRES(m)

El método GMRES se vuelve computacionalmente costoso en cada iteración del mismo. A medida que la dimensión del subespacio de búsqueda crece, también aumenta la cantidad de multiplicaciones matriz-vector y la necesidad de al-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

macenamiento. Por ese motivo, Saad y Schultz [8] propusieron una variación de GMRES con reinicio, el GMRES(m). En el método, se define un **ciclo de ejecución** (ver Figura 2.4), dentro del cual el subespacio de Krylov alcanza dimensión m , se almacena la solución estimada y el residuo calculados hasta ese instante, y se utiliza esta información para el nuevo ciclo. Cada ciclo contiene m iteraciones, y cada iteración consiste en el cálculo de los vectores ortogonales vía Método de Arnoldi. Es decir, en el j -ésimo ciclo, se utiliza el residuo r_{j-1} para construir m vectores de una base ortonormal de $\mathcal{K}^{(m)}(A, r_{j-1})$.

Iter. 1	Iter. 2	Iter. 3	...	Iter. m	Iter. 1	Iter. 2	Iter. 3	...	Iter. m	Iter. 1	Iter. 2	Iter. 3	...	Iter. m		Iter. 1	Iter. 2	Iter. 3	...	Iter. m
$v_1 = \frac{r_0}{\ r_0\ }$	v_2	v_3	...	v_m	$v_1 = \frac{r_1}{\ r_1\ }$	v_2	v_3	...	v_m	$v_1 = \frac{r_2}{\ r_2\ }$	v_2	v_3	...	v_m	...	$v_1 = \frac{r_{j-1}}{\ r_{j-1}\ }$	v_2	v_3	...	v_m
Ciclo 1 $\kappa^{(m)}(A, r_0)$					Ciclo 2 $\kappa^{(m)}(A, r_1)$					Ciclo 3 $\kappa^{(m)}(A, r_2)$					Ciclo j $\kappa^{(m)}(A, r_{j-1})$					

Figura 2.4: Estructura de cada ciclo de reinicio del método GMRES(m). **Fuente:** Elaboración propia.

Al final del ciclo, se devuelve un vector $z_j = V_m y_m$ tal que

$$x_j = x_{j-1} + z_j, \quad (2.26)$$

de tal manera a minimizar la norma euclíadiana del residuo:

$$\|r_j\|_2 = \min_{x_j \in x_{j-1} + \mathcal{K}^{(m)}(A, r_{j-1})} \|b - Ax_j\|. \quad (2.27)$$

La expresión 2.27 es similar a la obtenida en 2.11 para GMRES; en particular, GMRES(m) considera que el valor máximo de k en cada ciclo de ejecución es m . Si el criterio de convergencia no es satisfecho se ejecuta un nuevo ciclo, en caso contrario el método llega a su fin y se considera x_j como la solución del sistema lineal.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Algoritmo GMRES(m)

```
1: procedure GMRES( $A, b, m, \epsilon$ )                                ▷ Resolver el sistema  $Ax = b$ 
2:   Entrada: Elegir  $x_0$ , calcular  $r_0 = b - Ax_0$  y  $v_1 = r_0/\|r_0\|$ .
3:   Salida:  $x_j$ : solución aproximada del sistema lineal  $Ax = b$ .
4:   for  $j = 1, 2, \dots, m$  do
5:     Calcular  $h_{ij} = \langle Av_j, v_i \rangle$  para  $i = 1, 2, \dots, j$ 
6:     Calcular  $\tilde{v}_{j+1} = Av_j - \sum_{i=1}^j h_{ij}v_i$ 
7:     Calcular  $h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$ 
8:     Calcular  $v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$ 
9:   end for
10:  Hallar  $y_j$  = vector  $y$  que minimice  $\|\beta e_1 - \tilde{H}_k y\|$ 
11:  Formar la solución  $x_j = x_0 + V_j y_j$ 
12:  Reinicio: Calcular  $r_j = b - Ax_j$ 
13:  if  $\|r_j\| < \epsilon$  then
14:    stop
15:  else
16:     $x_0 = x_j$ ;  $v_1 = r_j/\|r_j\|$ , volver al paso 4.
17:  end if
```

2.10. Ventajas y desventajas del método con reinicio

La ventaja que permite el método con reinicio GMRES(m) es la de mantener bajo control los costos del almacenamiento y del tiempo del procesamiento de la base del subespacio de búsqueda. Se generan como máximo m vectores de la base ortonormal del subespacio de Krylov. El valor de reinicio m tiene un valor constante, generalmente $m \ll n$. Actualmente m se elige en base a la experimentación numérica, y no se ha encontrado aún su valor óptimo. Otra propiedad importante radica en que el GMRES(m) mantiene la convergencia ya que la sucesión de las normas residuales obtenidas en ciclos de GMRES(m) sigue siendo no creciente.

Sin embargo, el reinicio del método ralentiza la convergencia del GMRES ya que la información útil relativa a los subespacios de búsqueda generados en los ciclos anteriores (en particular, los vectores ortonormales del Método de Arnoldi) no se preserva con el reinicio. Además, si la elección del parámetro de reinicio m no es apropiada, la convergencia del GMRES(m) no está garantizada, y el mé-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

todo puede desembocar en una convergencia lenta e inclusive estancamiento. Si ocurre estancamiento, una estrategia simple consiste en aumentar m , con el consecuente aumento del subespacio de búsqueda, pero esto no siempre asegura mejor convergencia y hace que el consumo de memoria sea prohibitivo e innecesario [26; 27]. De ahí, se hace necesario incluir otra información relevante del subespacio de búsqueda desecharido del ciclo anterior.

CAPÍTULO 3

Diseño del sistema de control para GMRES

3.1. Estrategias para mejorar GMRES(m)

En la literatura científica se ha buscado varias estrategias para mejorar el método GMRES(m). Una combinación de ellas podría ayudar a una convergencia más rápida. A continuación se describen algunas de ellas:

- **Elección adaptativa del parámetro m :** experimentalmente, se observa que el parámetro m tiene mayor influencia en la rapidez de convergencia y está fuertemente ligado al costo computacional del método, en memoria y tiempo de ejecución. Sin embargo, la mejor manera de seleccionarlo aún no se ha establecido. Varios trabajos de investigación han propuesto diferentes estrategias para elegir este parámetro [28; 29; 30].
- **Enriquecimiento (*augmentation* en inglés) del subespacio:** de acuerdo con ese enfoque, el subespacio de búsqueda es la suma directa de dos subespacios; el primero, formado por vectores que pertenecen al subespacio de Krylov $\mathcal{K}^{(m)}(A, r_{j-1})$, y un segundo conjunto de vectores que guardan información relevante generada en los ciclos anteriores, de modo tal que el subespacio de búsqueda resultante todavía sea de dimensión mucho menor que la dimensión del problema original [27; 31; 32].
- **Precondicionamiento del sistema lineal:** se resuelve el sistema lineal modificado por la izquierda $M^{-1}Ax = M^{-1}b$ o por la derecha $AM^{-1}(Mx) =$

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS

b. La matriz M , llamada **precondicionadora del sistema**, es una matriz no singular del mismo orden que A , tal que el sistema lineal modificado esté mejor condicionado que el sistema lineal original [21]. Un buen precondicionador puede mejorar la convergencia del método iterativo, de manera suficiente para compensar el costo de construir y aplicar el precondicionador [18].

3.2. Modelado del método iterativo como sistema de control

Según este paradigma, descrito por A. Bhaya y E. Kaszkurewiczz [9], es posible asociar un método iterativo para sistemas lineales a un sistema de control de lazo cerrado y de tiempo discreto. A modo de ejemplo, en la figura 3.1 se define $y_k = Ax_k$ como el vector de salida del sistema, el vector r_k representa el residuo o diferencia entre el vector de entrada b y el vector de salida y_k . Así, el problema numérico de resolver el sistema lineal $Ax = b$ equivale a forzar la salida y a que sea igual a la entrada b , mediante un controlador apropiado C . Cuando esto ocurra, el vector de estado x representa la solución deseada del sistema lineal $Ax = b$. La solución única del sistema depende del rango de la matriz de controlabilidad $[b, Ab, A^2b, \dots, A^{m-1}b]$, que contiene a los vectores base del subespacio de Krylov. Elecciones diferentes del controlador C representan diferentes métodos iterativos.

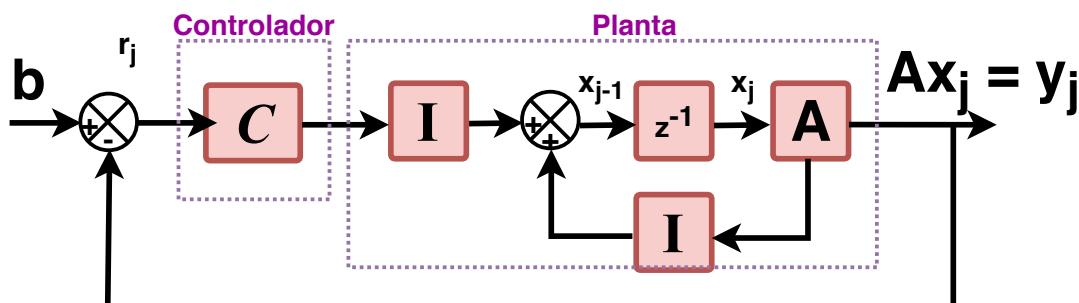


Figura 3.1: Método iterativo para sistemas lineales, representado como sistema de control de lazo cerrado. **Fuente:** [9].

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

3.3. Formulación de control

A fin de formular la resolución iterativa de un sistema lineal $Ax = b$ como un sistema de control, consideramos el residuo $r_j = b - Ax_j$ como el error o desviación [9]. Entonces, resolver el sistema lineal equivale a reducir el residuo r_j a un vector nulo.

Dado un sistema lineal de ecuaciones, en general, un método iterativo puede ser descrito como una sucesión de aproximaciones x_j de la forma:

$$\left\{ \begin{array}{l} z_j = \text{Solver} \\ x_j = x_{j-1} + z_j \\ r_j = b - Ax_j \end{array} \right. \quad (3.1)$$

3.4. GMRES(m) con parámetro m controlado

A continuación se modela el GMRES(m) como un sistema de control realimentado para luego acoplar a ese sistema un controlador para el parámetro m . El objetivo del controlador es cambiar el valor del reinicio en cada llamada a GMRES, el valor que vamos a usar en cada llamada estará dado por un controlador inspirado en un controlador PID (proporcional-integral-derivativo).

En el contexto de GMRES(m), la relación entre el residuo r_{j-1} y z_j está dado por la subrutina GMRES con el parámetro interno fijo m . Las recurrencias asociadas a este modelo realimentado son:

$$\left\{ \begin{array}{l} z_j = \text{GMRES}(m) \\ x_j = x_{j-1} + z_j \\ r_j = b - Ax_j \end{array} \right. \quad (3.2)$$

La figura 3.3 muestra un diagrama de bloques correspondiente a las recurrencias que modelan al GMRES(m).

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS

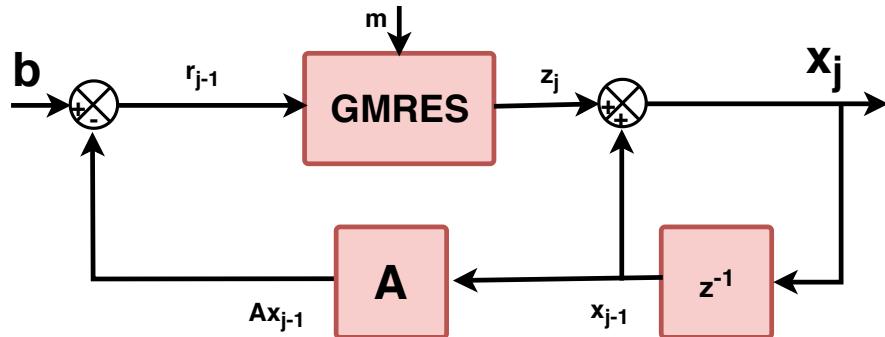


Figura 3.2: Diagrama de bloques para GMRES(m). **Fuente:** [10; 11]

Suponiendo que m es un parámetro fijo, se define m_{j-1} como el parámetro de reinicio de ciclo $j - 1$. Entonces, se obtiene un $z_{j-1} = \text{GMRES}(A, r_{j-1}, m_{j-1})$. Para la llamada a GMRES(m), r_{j-1} se toma como resíduo «inicial» y m_{j-1} denota la dimensión máxima del sub-espacio de Krylov. Si m_{j-1} toma un valor mayor suficiente para que GMRES(m) realice un único ciclo, se tiene el método GMRES sin reinicio.

En este trabajo, se modificará el parámetro m_{j-1} para mejorar el valor de z_{j-1} . Para actualizar m_{j-1} se utiliza una recurrencia similar a la usada para x_j :

$$m_j = m_{j-1} + u_{j-1} \quad (3.3)$$

donde u_{j-1} es la variable de control, que puede ser una constante ($u_{j-1} = \alpha$) o un controlador PID. En este trabajo, se empleará una constante α entera positiva como sinónimo de que se aumentará la dimensión del subespacio de búsqueda si se detecta estancamiento.

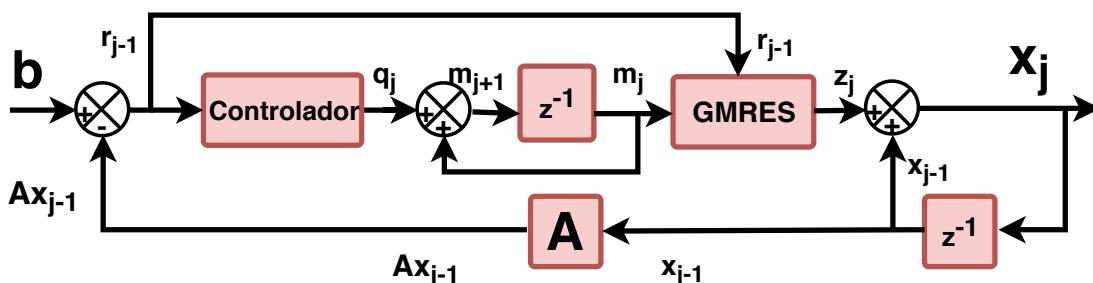


Figura 3.3: Diagrama de bloques para GMRES(m) adaptativo. **Fuente:** [10; 11].

3.5. Propuestas de enriquecimiento de subespacio

La idea central del enriquecimiento consiste en determinar un subespacio de búsqueda de dimensión $s = m + k$:

$$\text{span}\{v_1, v_2, \dots, v_m, \omega_1, \omega_2, \omega_k\} \quad (3.4)$$

donde los primeros m vectores son determinados por el Método de Arnoldi con el vector inicial $v_1 = r_{j-1}/\|r_{j-1}\|$. Los k vectores restantes contienen información relevante guardada de los ciclos anteriores.

3.5.1. LGMRES(m, l)

El método de enriquecimiento propuesto por A. Baker [33] intenta mejorar la convergencia al añadir de l vectores de aproximación de errores de la forma

$$\psi_{j-1} = x_{j-1} - x_{j-2}, \quad (3.5)$$

donde $\psi_0 = 0$. Baker demuestra que los vectores de aproximación de error pertenecen a los subespacios de búsqueda de ciclos anteriores, por lo tanto guardan información relevante a los mismos. El subespacio resultante es:

$$\mathcal{S}_{LGMRES} = \mathcal{K}_j^{(m)}(A, r_{j-1}) \bigcup \text{span}\{\psi_k\}, k = (j-l), \dots, (j-1). \quad (3.6)$$

El Método de Arnoldi queda modificado por la adición de los vectores de aproximación de error. La matriz V_{s+1} es la matriz de orden $n \times (s+1)$ cuyos $m+1$ primeros vectores columna son los vectores de Arnoldi y los demás l vectores columna son resultado de ortogonalizar los vectores de aproximación de error $\psi_k, k = (j-l), \dots, (j-1)$ con respecto a los primeros $m+1$ vectores columna. La matriz W_s de orden $n \times s$ contiene a las primeras m columnas de V_{s+1} , y sus

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

últimas l columnas son los l vectores de aproximación de error. La relación en el ciclo j es

$$AW_s = V_{s+1} \tilde{H}_s \quad (3.7)$$

Es importante aclarar que LGMRES no ayuda a la convergencia cuando un problema sufre estancamiento para algún valor del parámetro de reinicio m .

3.5.2. GMRES-E(m, d)

El método propuesto por R. Morgan [34] enriquece el subespacio de Krylov con d vectores armónicos de Ritz asociados a los valores armónicos de Ritz más pequeños. A modo de definición, un valor armónico de Ritz $\tilde{\lambda}_i$ con vector armónico de Ritz asociado $\varphi_i = W g_i$, $g_i \in \mathbb{C}^{s \times 1}$, relativo al subespacio $A\mathcal{K}_{j-1}^{(s)}(A, r_{j-1})$ satisface:

$$(A\varphi_i - \tilde{\lambda}\varphi_i) \perp A\mathcal{K}^{(s)}(A, r_{j-1}) \iff (AW_s)^H (AW_s g_i - \tilde{\lambda}_i W_s g_i) = 0. \quad (3.8)$$

El problema de autovalores reducido $A\varphi_i \approx \tilde{\lambda}\varphi_i$ se transforma así en un problema de autovalores generalizado:

$$W_s^H A^H A W_s g_i = \tilde{\lambda}_i W_s^H A^H W_s g_i \quad (3.9)$$

donde W_s es una matriz de orden $n \times s$ ($s = m + d$), cuyos primeros m vectores columna son los vectores de Arnoldi y cuyos últimos d vectores columna corresponden a los autovectores aproximados φ .

3.6. Propuesta del sistema de control para el GMRES(m)

En el capítulo anterior, se ha explicado que el GMRES(m) halla una solución de la forma

$$x_j = x_{j-1} + V_j^{(m)} y_j, \quad (3.10)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Entonces, cuando la norma euclíadiana del vector y_j es muy pequeña, por ej., menor que un umbral de tolerancia δ ocurre:

$$V_j^{(m)} y_j \approx 0 \Rightarrow x_j \approx x_{j-1} \Rightarrow b - Ax_j \approx b - Ax_{j-1} \Rightarrow r_j \approx r_{j-1} \quad (3.11)$$

y ocurre estancamiento. Por lo tanto, una estrategia consiste en modificar el tamaño del subespacio de búsqueda para el siguiente ciclo, con un controlador para el parámetro m . En esta propuesta se ha probado la siguiente recurrencia:

$$m_j = m_{j-1} + u_j; \text{ con } u_j = \begin{cases} \alpha & \text{si } \|y_j\| < \delta \\ 0 & \text{si } \|y_j\| \geq \delta \end{cases}, \quad (3.12)$$

donde δ es un umbral de medida del estancamiento. La estrategia de comutación al controlador se debe a que, si hay estancamiento, los vectores de aproximación al error no ayudan a superarlo. Si no hay estancamiento, no es necesario modificar el parámetro m .

Luego de todo lo descrito, el subespacio de búsqueda queda Enriquecido de la siguiente manera:

$$x_j = x_{j-1} + z_j; \quad z_j \in \mathcal{K}^{(m_j)}(A, r_{j-1}) \cup \mathcal{U}. \quad (3.13)$$

$$\mathcal{U} = \begin{cases} \text{span}\{\psi_{j-l}, \dots, \psi_{j-1}, \varphi_1, \dots, \varphi_d\} & \text{si } \|y_{j-1}\| \geq \delta \\ \text{span}\{\varphi_1, \dots, \varphi_d\} & \text{si } \|y_{j-1}\| < \delta \end{cases}$$

donde:

- los ψ_i son los vectores de aproximación de error [33].
- los φ_i son los vectores armónicos de Ritz asociados a los menores valores

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS

armónicos de Ritz en magnitud [34].

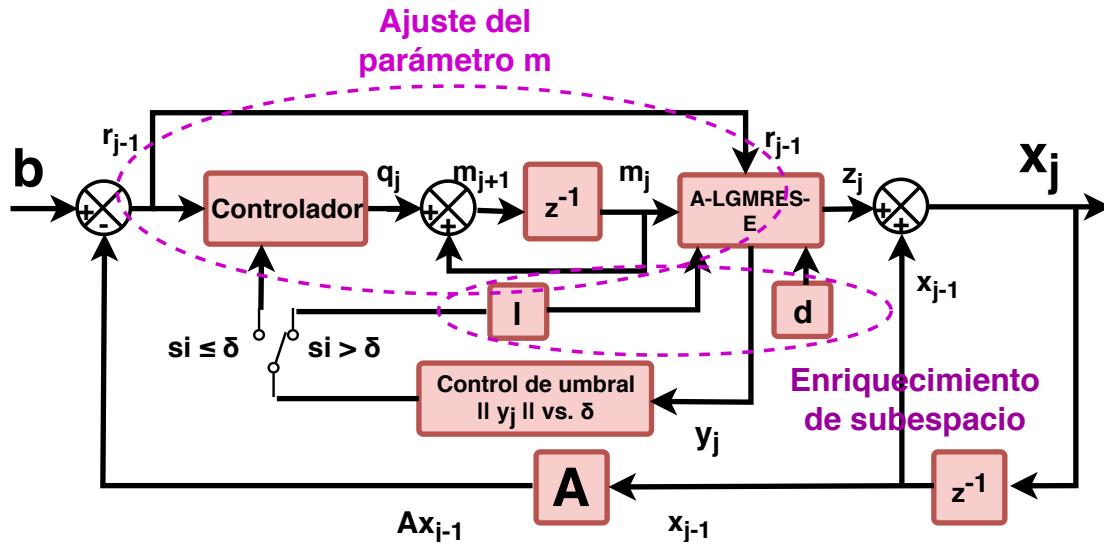


Figura 3.4: Diagrama de bloques para el sistema de control propuesto. **Fuente:** Elaboración propia.

CAPÍTULO 4

Esparcimiento electromagnético

4.1. Electromagnetismo computacional

El estudio de los fenómenos eléctricos y magnéticos y la interacción entre ellos, ha sido una de las ramas de la Física centrales del siglo XX, y su estudio con herramientas computacionales todavía está por alcanzar un buen nivel de madurez durante este siglo XXI. La solución analítica de ecuaciones para problemas electromagnéticos es compleja y, para aplicaciones industriales, generalmente se requieren aproximaciones aún más complejas que sin embargo requieren una solución en el menor tiempo posible. El modelado y simulación por computadora de los fenómenos electromagnéticos se conoce como electromagnetismo computacional (abreviada aquí como EMC) [35].

El EMC es una ciencia multidisciplinaria. Sus disciplinas centrales son la teoría electromagnética y los métodos numéricos, pero para implementaciones industriales, el modelado geométrico y su visualización, la computación y la complejidad algorítmica, todos tienen roles importantes. Las aplicaciones del EMC son múltiples e incluyen: antenas, efectos biológicos de EM, creación de nuevos métodos de diagnóstico y tratamiento médico, diseño y testeo de circuitos de alta velocidad, estudio de la superconductividad, fabricación de dispositivos de microondas, monitoreo ambiental, desarrollo de nuevos materiales, aviónica, comunicaciones, generación y conservación de energía, radares e imágenes, vigilancia y recolección de inteligencia.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

4.2. Esparcimiento electromagnético

El término «**esparcimiento**» (o *scattering*, en inglés) se utiliza a menudo en la literatura científica y técnica para referirse de manera general a los efectos producidos sobre una onda cuando la misma incide sobre un obstáculo [12]. El fenómeno de esparcimiento es estudiado en la Acústica, la Óptica y el Electromagnetismo, en este trabajo nos limitaremos al estudio del esparcimiento electromagnético.

Desde el punto de vista del problema de esparcimiento electromagnético, todos los materiales de construcción de equipos están compuestos por partículas cargadas eléctricamente. Cuando una onda electromagnética incide sobre el material, las cargas excitadas irradian nuevas ondas electromagnéticas *esparcidas* en direcciones arbitrarias, que no necesariamente tienen que coincidir con la dirección inicial de la onda incidente.

De acuerdo al modelo matemático relacionado, las ondas electromagnéticas son descritas por cuatro funciones vectoriales de la posición $x \in \mathbb{R}^3$ y del tiempo $t \in \mathbb{R}$, llamadas **intensidad de campo eléctrico E, intensidad de campo magnético H, desplazamiento eléctrico D e inducción magnética B**. Estos campos electromagnéticos son originados a su vez por distribuciones de cargas eléctricas estáticas o flujos continuos de carga -las corrientes eléctricas-. La distribución de cargas se modela con una función escalar ρ conocida como **densidad de carga**, mientras que la corriente eléctrica se describe con una función vectorial j . Las interacciones entre las funciones de onda electromagnética constituyen las **Ecuaciones de Maxwell**¹.

Los campos D y B pueden ser suprimidos de la formulación matemática mediante leyes constitutivas dependientes del medio considerado. Así, la explicación del fenómeno electromagnético se reduce a la descripción de los campos

¹James Clerk Maxwell (1831-1879), físico-matemático escocés.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

fundamentales \mathbf{E} y \mathbf{H} .

4.3. Ecuaciones de Maxwell

La figura 4.1 muestra una vista esquemática del problema de esparcimiento electromagnético. Un campo electromagnético incidente \mathbf{E}_{inc} interactúa con un sistema de partículas que ocupa un dominio de \mathbb{R}^3 . Las ecuaciones de Maxwell describen en forma diferencial la interacción entre los campos eléctricos y magnéticos, de la siguiente manera:

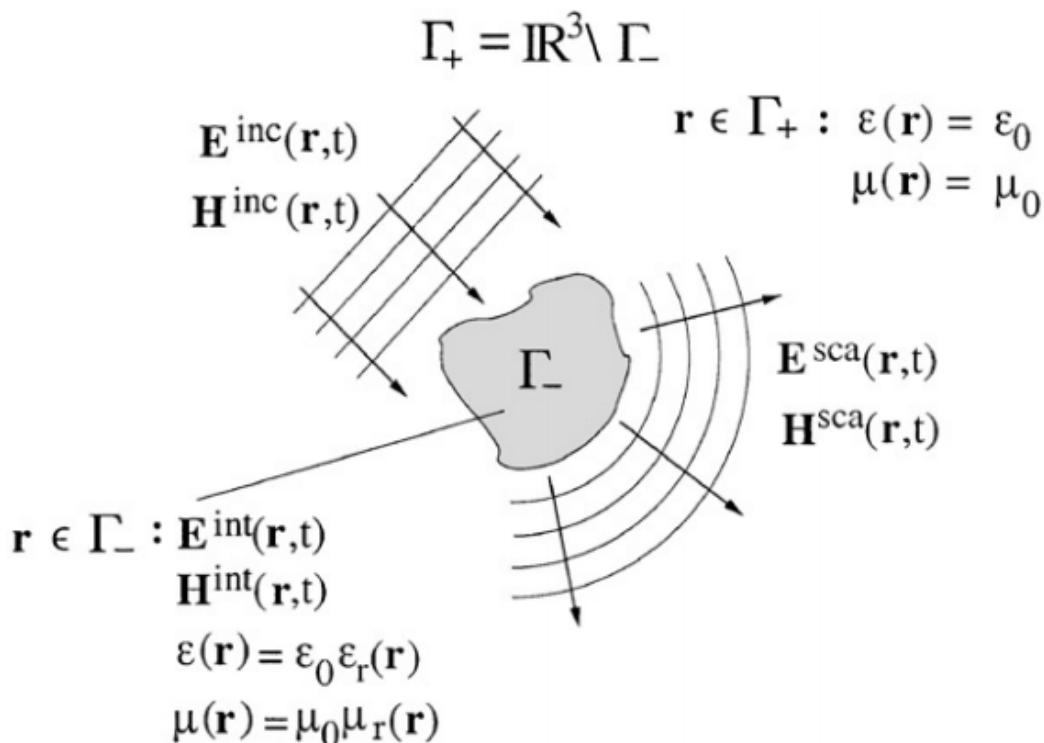


Figura 4.1: Descripción del problema de esparcimiento electromagnético. **Fuente:** [36].

$$\nabla \cdot \mathbf{D} = \rho \quad (\text{Ley Eléctrica de Gauss}) \quad (4.1)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (\text{Ley Magnética de Gauss}) \quad (4.2)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (\text{Ley de Faraday}) \quad (4.3)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

$$\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{j} \quad (\text{Ley de Ampère}) \quad (4.4)$$

donde ∇ es el operador nabla del Análisis Vectorial [37].

Para analizar la interacción de un campo electromagnético con un medio material se introducen dos leyes constitutivas que relacionan los campos \mathbf{D} y \mathbf{B} con \mathbf{E} y \mathbf{H} respectivamente:

$$\mathbf{D} = \mathbf{D}(\mathbf{E}, \mathbf{H}) \quad (4.5)$$

$$\mathbf{B} = \mathbf{B}(\mathbf{E}, \mathbf{H}) \quad (4.6)$$

Las ecuaciones constitutivas generalmente contienen relaciones complicadas. En general, no dependen solamente del carácter molecular de los materiales como la polarización o la magnetización, sino también de las magnitudes macroscópicas como la temperatura o la densidad del material [38]. Además, hay dependencias del tiempo, por ejemplo, la histéresis: los campos medidos en un tiempo τ dependen también de los tiempos $t < \tau$.

Si se ignora los medios ferro-eléctricos y ferro-magnéticos y si los campos son relativamente pequeños, las ecuaciones constitutivas pueden ser modeladas por ecuaciones lineales de la forma:

$$\mathbf{D} = \epsilon \mathbf{E} \quad (4.7)$$

$$\mathbf{B} = \mu \mathbf{H} \quad (4.8)$$

donde ϵ y μ son la permitividad eléctrica y la permeabilidad magnética respectivamente. En medios homogéneos, isótropos y lineales, ϵ y μ se consideran como constantes físicas propias del medio. Para una mejor definición de las funciones para cualquier medio, se recomienda [39].

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Sustituyendo las ecuaciones 4.7 y 4.8 en las ecuaciones 4.1 a 4.4 obtenemos las ecuaciones de Maxwell en términos de los campos fundamentales \mathbf{E} y \mathbf{H} .

$$\nabla \cdot \mathbf{E} = \rho \quad (4.9)$$

$$\nabla \cdot (\mu \mathbf{H}) = 0 \quad (4.10)$$

$$\nabla \times \mathbf{E} = -\frac{\partial(\mu \mathbf{H})}{\partial t} \quad (4.11)$$

$$\nabla \times \mathbf{H} = \frac{\partial(\epsilon \mathbf{E})}{\partial t} + \mathbf{j} \quad (4.12)$$

4.4. Ecuaciones de Maxwell en el espacio vacío

Si se considera ahora una región del espacio en la cual no existe una fuente de corriente ($\mathbf{j}_a = 0$) y tal que la conductividad es nula ($\sigma = 0$) obtenemos:

$$\mathbf{j} = \mathbf{j}_a + \sigma \mathbf{E} = 0 \quad (4.13)$$

Así, las ecuaciones de Maxwell 4.9 a 4.12 se reducen a:

$$\nabla \cdot \mathbf{E} = 0 \quad (4.14)$$

$$\nabla \cdot \mathbf{H} = 0 \quad (4.15)$$

$$\nabla \times \mathbf{E} = -\frac{\partial(\mu \mathbf{H})}{\partial t} \quad (4.16)$$

$$\nabla \times \mathbf{H} = \frac{\partial(\epsilon \mathbf{E})}{\partial t} \quad (4.17)$$

En el vacío, ϵ y μ son escalares constantes y denotados por ϵ_0 y μ_0 . En este trabajo, se medirán fenómenos electromagnéticos sobre aire, un medio aproximadamente igual al vacío.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

4.5. Ecuación de onda para el espacio libre

Tomando el rotacional de la ecuación 4.11 y utilizando la ecuación 4.12 obtenemos:

$$\nabla \times (\nabla \times \mathbf{E}) = -\nabla \times (\mu_0 \frac{\partial \mathbf{H}}{\partial t}) = -\mu_0 \frac{\partial(\nabla \times \mathbf{H})}{\partial t} = -\mu_0 \epsilon_0 \frac{\partial^2 \mathbf{E}}{\partial t^2} \quad (4.18)$$

Luego, considerando la identidad del Álgebra Vectorial: $\nabla \times \nabla \times \mathbf{E} = \nabla(\nabla \cdot \mathbf{E}) - \Delta \mathbf{E}$, donde $\Delta = \nabla^2$ es el operador laplaciano, y utilizando la ecuación 4.14 resulta:

$$\nabla \times \nabla \times \mathbf{E} = \nabla(\nabla \cdot \mathbf{E}) - \Delta \mathbf{E} = -\Delta \mathbf{E}, \quad (4.19)$$

Combinando las ecuaciones 4.18 y 4.19 obtenemos:

$$\Delta \mathbf{E} - \mu_0 \epsilon_0 \frac{\partial^2 \mathbf{E}}{\partial t^2} = 0, \quad (4.20)$$

llamada ecuación de onda para \mathbf{E} , debido a que sus soluciones son campos eléctricos que se propagan en el espacio.

4.6. Ecuación de Helmholtz

Se considera que la solución a la ecuación de onda 4.21 depende armónicamente del tiempo:

$$\mathbf{E} = \mathbf{E}_z = \operatorname{Re}[u(x, y)e^{-j\omega t}] \quad (4.21)$$

donde $\operatorname{Re}[\cdot]$ es la parte real de una cantidad compleja y j (sin negritas) es la unidad imaginaria ($j^2 = -1$). Sustituyendo en la ecuación 4.21 resulta:

$$\Delta[\operatorname{Re}\{u(x, y)e^{-j\omega t}\}] - \frac{1}{c^2} \frac{\partial^2[\operatorname{Re}\{u(x, y)e^{-j\omega t}\}]}{\partial t^2} = 0. \quad (4.22)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Considerando que los operadores ∇ , $\frac{\partial^2}{\partial t^2}$ y Re son lineales, el orden de las operaciones se puede intercambiar:

$$\text{Re} \left\{ e^{-j\omega t} \left[\Delta u + \frac{\omega^2}{c^2} u \right] \right\} = 0, \quad (4.23)$$

donde $u = u(x, y)$ por simplicidad de notación. Como la ecuación se debe satisfacer para todo tiempo t , queda:

$$\Delta u + \frac{\omega^2}{c^2} u = 0 \quad (4.24)$$

La ecuación 4.24 se llama **ecuación de Helmholtz**.

4.7. Problema de esparcimiento

A continuación se considera la reproducción del problema de esparcimiento electromagnético para una cavidad embebida en una superficie conductiva plana [14]. Se asume que el medio y el material son invariantes en la dirección z , ver Figura 4.2. Se considera además que el medio es no magnético y existe una permeabilidad magnética constante: $\mu(x, y) = \mu_0$ en todo el medio. El semiespacio por encima de la superficie conductiva está lleno con un medio homogéneo, lineal e isotrópico con permitividad eléctrica ϵ_0 . La propiedad electromagnética del medio está caracterizada por la permitividad relativa: $\epsilon_r(x, y) = \epsilon(x, y)/\epsilon_0$.

Para la polarización magnética transversa (TM), en la cual el campo magnético es transverso a la dirección invariante y el campo eléctrico tiene la forma $\mathbf{E} = (0, 0, u(x, y))$, las ecuaciones de Maxwell se reducen a una ecuación de Helmholtz:

$$\Delta u + k_0^2 \epsilon_r u = f, \text{ en } \Omega, \quad (4.25a)$$

$$u = 0, \text{ en } S, \quad (4.25b)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

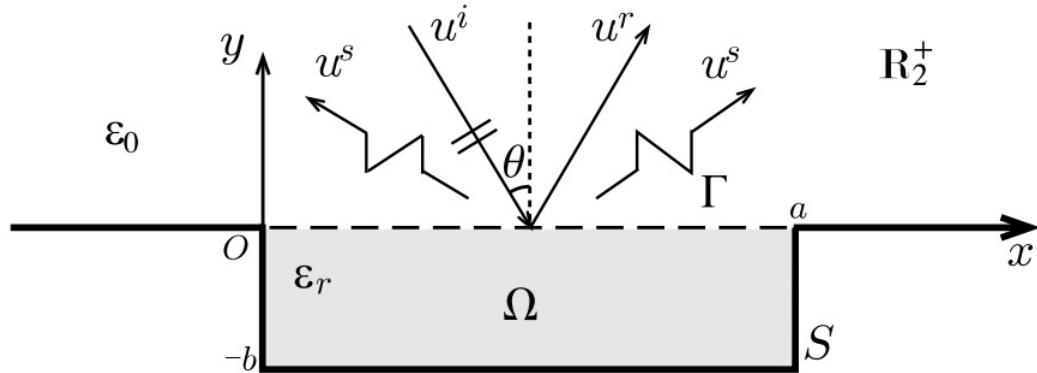


Figura 4.2: Geometría del problema de esparcimiento electromagnético sobre una cavidad. **Fuente:** [14].

$$\partial_n u = T(u) + g, \text{ en } \Gamma, \quad (4.25c)$$

Para resolver el sistema de ecuaciones diferenciales, se utilizará el método de las diferencias finitas espaciales, el cual consiste en aproximar las derivadas en las ecuaciones diferenciales a través de cocientes diferenciales obtenidos a partir de las series de Taylor. Al proceso de reemplazo de los operadores diferenciales por sus respectivos cocientes diferenciales se denomina **discretización**. El error entre la solución numérica y la solución exacta está determinado por el error introducido al reemplazar las derivadas por sus respectivos cocientes diferenciales. Este error es conocido como **error de discretización o de truncamiento**.

Para el problema 4.25 se considera una discretización en diferencias finitas de su dominio:

$$\Omega_h = (x_i, y_j) : i = 0, 1, \dots, M + 1; j = 0, 1, \dots, N + 1 \quad (4.26)$$

una partición uniforme de $\Omega = [0, a] \times [-b, 0]$, donde:

$$h_x = \frac{a}{M + 1}, \quad h_y = \frac{b}{N + 1}, \quad x_i = ih_x, \quad y_j = -b + jh_y \quad (4.27)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

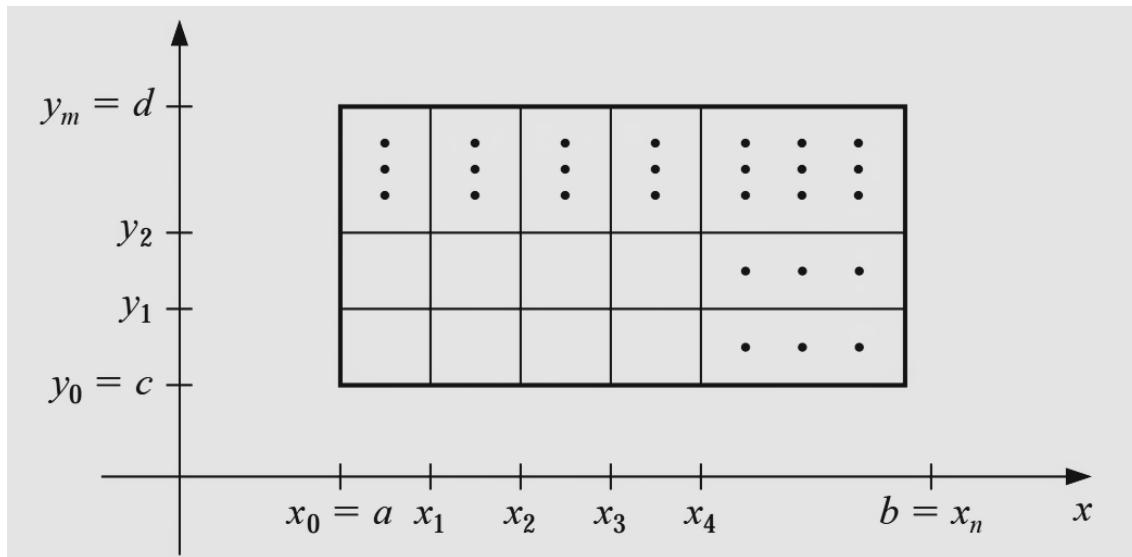


Figura 4.3: Geometría de la discretización en una malla rectangular. Cada vértice de la malla es un punto a medir. **Fuente:** [40].

Las derivadas espaciales son aproximadas a través de diferencias centradas de segundo orden:

$$\frac{\partial^2 f(x, y)}{\partial x^2} = \frac{f(x + h_x, y) - 2f(x, y) + f(x - h_x, y)}{h_x^2} + O(h_x^2) \quad (4.28a)$$

$$\frac{\partial^2 f(x, y)}{\partial y^2} = \frac{f(x, y + h_y) - 2f(x, y) + f(x, y - h_y)}{h_y^2} + O(h_y^2) \quad (4.28b)$$

donde $O(h_x^2)$ y $O(h_y^2)$ son los errores de truncamiento, cuyo valor es fuertemente dependiente de h_x^2 y h_y^2 . Sea $u_{i,j}$ la solución en el punto (x_i, y_j) del dominio discreto. Dicho de otra manera, cada $u_{i,j}$ representa una incógnita del sistema. El sistema lineal de ecuaciones para la ecuación en diferencias finitas 4.25a está dado por

$$\frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h_x^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h_y^2} + k_0^2 \epsilon_r(x_i, y_j) u_{i,j} = f(x_i, y_j) \quad (4.29)$$

para todo $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$. Para la ecuación 4.25b se tiene:

$$u_{0,j} = u_{M+1,j} = u_{i,0} = 0, i = 1, 2, \dots, M, j = 1, 2, \dots, N \quad (4.30)$$

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

La forma discreta de la condición de frontera 4.25c está dada por:

$$\frac{u_{i,N+1} - u_{i,N}}{h_y} = \sum_{l=1}^M g_{il} u_{l,N+1} + g(x_i), \quad i = 1, 2, \dots, M \quad (4.31)$$

donde:

$$g_{il} = g_{il}^{re} + i g_{il}^{im} \quad (4.32a)$$

$$g_{il}^{re} = -t_{il} \frac{k_0 |x_i - x_l|}{2} Y_1(k_0 |x_i - x_l|) \quad (4.32b)$$

$$g_{il}^{im} = \frac{k_0 h_x}{2} \frac{J_1(k_0 |x_i - x_l|)}{|x_i - x_l|} \quad (4.32c)$$

y:

$$t_{il} = \begin{cases} \frac{1}{h_x} (1 - \ln(2)); & \text{si } |i - l| = 1; \\ -\frac{2}{h_x}; & \text{si } i = l; \\ \frac{1}{h_x} \ln \frac{(i-l)^2}{(i-l)^2 - 1}; & \text{caso contrario.} \end{cases} \quad (4.33)$$

Finalmente, queda un sistema de ecuaciones de la forma $Au = f$, donde A , u y f se componen como se muestra en la figura 4.4.

4.8. Aspectos de diseño

A fin de que un método computacional sea aplicable a la resolución del problema de esparcimiento electromagnético, el consenso es que las dimensiones de la malla deben ser del orden de la mitad de longitud de onda esparcida o, de ser posible, mucho menor. Esto se debe al fenómeno de solapamiento (*aliasing*, en inglés), ocurre cuando se muestrea una onda, con longitud de onda corta, y el resultado se malinterpreta como una oscilación con una longitud de onda mucho mayor. Este problema es muy frecuente en la discretización espacial de la onda [41].

Debido a esto, la computación de la solución numérica se vuelve desafiante cuando los sistemas físicos son de gran tamaño comparadas con la longitud

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

$$\mathbf{A} = \begin{bmatrix} A_1 + D_1 & h_y^{-2} I_M & & & \\ h_y^{-2} I_M & A_1 + D_2 & h_y^{-2} I_M & & \\ & h_y^{-2} I_M & A_1 + D_3 & \ddots & \\ & & \ddots & \ddots & h_y^{-2} I_M \\ & & & h_y^{-2} I_M & A_1 + D_N & h_y^{-2} I_M \\ & & & & h_y^{-2} I_M & A_2 \end{bmatrix},$$

$$A_1 = h_x^{-2} \cdot \text{tridiag}[1, -2, 1] - 2h_y^{-2} I_M,$$

$$D_j = k_0^2 \cdot \text{diag}[\varepsilon_r(x_1, y_j), \varepsilon_r(x_2, y_j), \dots, \varepsilon_r(x_M, y_j)],$$

$$G = (g_{il})_{i,l=1}^M,$$

$$A_2 = h_y^{-1} G - h_y^{-2} I_M,$$

$$\mathbf{u} = [u_{1,1}, \dots, u_{M,1}, u_{1,2}, \dots, u_{M,2}, \dots, u_{1,N+1}, \dots, u_{M,N+1}]^T,$$

$$\mathbf{f} = [f_\Omega, f_\Gamma]^T,$$

$$f_\Omega = [f(x_1, y_1), \dots, f(x_M, y_1), f(x_1, y_2), \dots, f(x_M, y_N)],$$

$$f_\Gamma = -h_y^{-1} [g(x_1), g(x_2), \dots, g(x_M)].$$

Figura 4.4: Estructura del sistema lineal $Au = f$. **Fuente:** [14].

de onda de los campos: la malla resultante genera una cantidad muy grande de elementos, y el método numérico se vuelve particularmente inestable para sistemas muy grandes.

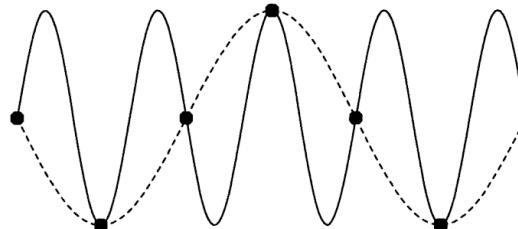


Figura 4.5: Discretización errónea de una onda debido al *aliasing*. **Fuente:** [41].

CAPÍTULO 5

Experimentos numéricos

En este capítulo se presentan las métricas obtenidas al resolver sistemas lineales con los métodos iterativos sugeridos en las secciones anteriores, utilizando parámetros tanto fijos como variables. La resolución de los sistemas lineales fue realizada con el software Matlab®R2016a sobre sistema operativo Windows de 64 bits en una computadora personal con un procesador Intel Core i5-4210U y 8 GB de memoria RAM. Si el método iterativo utilizado alcanza la cantidad máxima de 2000 ciclos y no obtiene una norma residual relativa de 10^{-6} , se afirma que **el método no converge**.

Las matrices A utilizadas para medir la resolución del problema de esparcimiento electromagnético son no hermíticas; sus características son presentadas en la Tabla 5.1. Para todos los sistemas lineales, n es la dimensión de la matriz de coeficientes A , $nnz(A)$ es el número de coeficientes distintos de cero y $cond(A)$ es el número de condición, el *Fill-in* o relleno es el porcentaje de elementos no nulos de la matriz A con respecto al total de elementos. En cuanto al problema electromagnético, M y N son las cantidades enteras positivas para la discretización de la cavidad rectangular.

El método propuesto, aquí llamado A-LGMRES-E para fines de identificación, es comparado con el método estándar GMRES(m) de Saad y Schultz [8] y con otras metodologías que tratan de acelerar la convergencia, las cuales son:

- **GMRES(m_j)**: es el metodo GMRES(m) con el agregado de la estrategia que permite modificar el parámetro m en cada ciclo, sin realizar enriquecimiento,

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Tabla 5.1: Matrices resultantes de la discretización del problema de esparcimiento electromagnético.

A	n	nnz(A)	Fill-in (%)	cond(A)	M	N	k_0
cavity01	390	3258	2.14	1.1974e+03	39	9	2π
cavity02	2475	21633	0.35	1.2469e+04	99	24	2π
cavity03	885	7583	0.96	1.8851e+03	59	14	4π
cavity04	2475	21633	0.35	4.5310e+04	99	24	4π
cavity05	2475	21633	0.35	8.1072e+03	99	24	8π
cavity06	9950	88258	0.08	1.4982e+05	199	49	8π
cavity07	22425	199883	0.03	2.8573e+04	299	74	10π

Fuente: [42].

- **GMRES-E(m, d)**: agrega en cada ciclo d valores armónicos de Ritz calculados en el ciclo anterior, y
- **LGMRES(m, l)**: agrega los últimos l errores de aproximación en cada ciclo.

5.1. Configuración inicial de los parámetros

El sistema $Au = f$ fue resuelto con los siguientes métodos iterativos: GMRES(m), GMRES(m_j) con parámetro de reinicio variable, GMRES-E(m, d), LGMRES(m, l) y A-LGMRES-E(m_j, l, d). Por lo general, para los métodos iterativos con reinicio se elige un parámetro $m \ll n$ [8; 29]. Se elige $m = 30$ ya que a menudo es la opción predeterminada en paquetes de software de Álgebra Lineal Numérica, por ejemplo PETSc. Los valores de l y d se eligen pequeños para no aumentar significativamente la cantidad de operaciones matriz-vector [33; 34]. Para limitar el uso de memoria en los métodos adaptativos, se restringe la variación de m entre un valor mínimo y un valor máximo, $m_{min} \leq m_j \leq m_{max}$. El problema de vectores armónicos de Ritz se resuelve con la función *eigs* de MATLAB [43].

En general:

- la estimación inicial de la solución es el vector nulo,

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

- el criterio de parada sobre la norma residual relativa es $\frac{\|r_j\|_2}{\|r_0\|_2} < 10^{-6}$ o un cantidad máxima de 2000 reinicios de ciclo,
- los tiempos de ejecución reportados son el promedio de 5 ejecuciones del método iterativo,
- los parámetros propios de cada método iterativo son:
 - GMRES(m): $m = 30$.
 - GMRES(m_j): $m_0 = m_{min} = 30$, $m_{max} = 100$
 - LGMRES(m, l): $m = 27$, $l = 3$.
 - GMRES-E(m, d): $m = 27$, $d = 3$.
 - A-LGMRES-E(m_j, l, d) $m_0 = m_{min} = 30$, $m_{max} = 100$, $l = 1$, $d = 3$, $\delta \in [0,1; 1]$ and $\alpha \in \{2, 3, \dots, 6\}$.

La cavidad rectangular pertenece a un dominio $\Omega = [0; 1] \times [-0,25; 0]$ y está rellena de un medio no homogéneo:

$$\epsilon_r = \begin{cases} 2 & \text{in } \Omega_1 = (0,2; 0,8) \times (-0,25; -0,20), \\ 1 & \text{in } \Omega \setminus \Omega_1. \end{cases} \quad (5.1)$$

5.2. Resultados y discusión

Los resultados se presentan en la tabla 5.2. Las métricas son: el tiempo requerido para resolver el sistema lineal, y la cantidad de ciclos ejecutados en el proceso. Si el método no converge para un problema dado, se indica con la abreviatura NC. Los detalles de las curvas de convergencia, donde se mide la norma residual relativa en función de la cantidad de ciclos, se pueden observar en las figuras 5.2 y 5.3. Si todos los métodos iterativos convergen para un problema,

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

definimos que un método iterativo es óptimo si tiene los menores tiempos de ejecución y logra la convergencia a la solución del sistema lineal.

Tabla 5.2: Métricas para las matrices seleccionadas y métodos iterativos: tiempo de ejecución en segundos y cantidad de ciclos de ejecucion.

Problemas	GMRES(m)	GMRES(m_j)	LGMRES(m, l)	GMRES-E(m, d)	A-LGMRES-E
	Tiempo (Ciclos)	Tiempo (Ciclos)	Tiempo (Ciclos)	Tiempo (Ciclos)	Tiempo (Ciclos)
cavity01	3.22 (106)	2.19 (17)	1.88 (67)	3.38 (80)	1.82 (21)
cavity02	NC (2000)	113.67 (136)	176.46 (1493)	179.5 (1224)	73.44 (108)
cavity03	13.69 (292)	21.28 (74)	11.27 (242)	11.83 (195)	10.54 (66)
cavity04	127.34 (1286)	98.94 (140)	54.01 (509)	74.78 (599)	46.17 (110)
cavity05	80.65 (795)	213.86 (272)	100.14 (929)	99.75 (786)	182.27 (269)
cavity06	NC (2000)	NC (2000)	NC (2000)	NC (2000)	NC (2000)
cavity07	NC (2000)	NC (2000)	NC (2000)	NC (2000)	16158 (1783)

Fuente: Elaboración propia.

Para los problemas difíciles, se observa que la norma $\|y_j\|$ es muy pequeña, por ej., menor que 10^{-2} , y coincide con la lentitud en el decrecimiento de la norma residual ($\|r_j\| \approx \|r_{j-1}\|$). La tabla 5.3 presenta la cantidad de ciclos necesaria para la convergencia de A-LGMRES-E, así como los valores máximos, mínimos y medios de $\|y_j\|$. El método propuesto no se estanca para los problemas, excepto para el problema cavity06. La convergencia para cavity07 se logra con un alto requisito de ciclos de ejecución.

Las figuras 5.1-(a) y 5.1-(b) muestran las curvas de convergencia y el valor $\|y_j\|$ versus la cantidad de ciclos de reinicio para los problemas cavity02 y cavity04. Al comparar los métodos de parámetro variable, se observa que el A-LGMRES-E, que incluye enriquecimiento de subespacio y actualiza el parámetro de reinicio según la regla 3.12, presenta una mejor tasa de convergencia que el método GMRES(m_j), que solo actualiza el parámetro de reinicio usando la misma regla pero no incluye información de ciclos anteriores. Los valores altos de $\|y_j\|$ indican una mejor tasa de convergencia. Comparando la dimensión de

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

Tabla 5.3: Resultados para A-LGMRES-E($m_j, 3, 1$). Valores máximos, mínimos y medios de $\|y_j\|$ para cada problema de la Tabla 1 y cantidad de ciclos requeridos para que $\|r_j\|/\|r_0\| < 10^{-6}$ o cantidad máxima de 2000 ciclos.

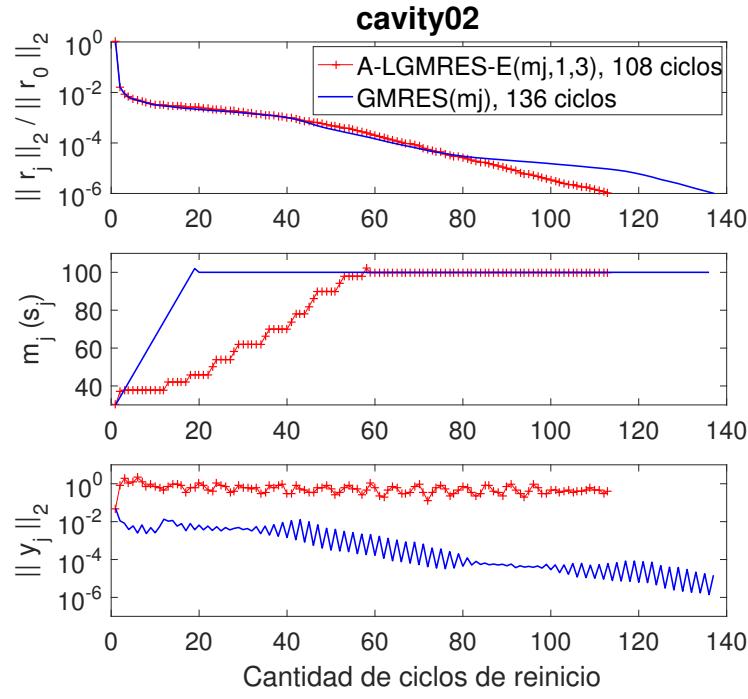
Problema	mín($\ y_j\ $)	media($\ y_j\ $)	máx($\ y_j\ $)	Ciclos
cavity01	0.03	0.91	5.27	21
cavity02	0.04	0.64	5.53	108
cavity03	0.18	1.01	5.53	66
cavity04	0.05	0.73	4.10	110
cavity05	0.06	0.76	18.45	269
cavity06	0.02	0.34	18.34	2000
cavity07	0.02	0.44	45.46	1783

Fuente: Elaboración propia.

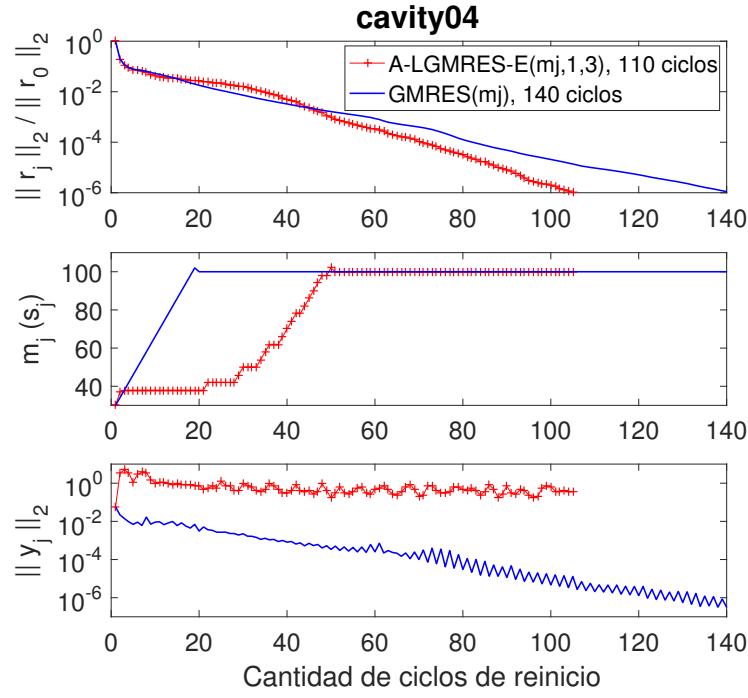
subespacio de búsqueda de los métodos adaptativos de la Figura 5.1 (parámetros m_j y s_j), el A-LGMRES-E tiene menor crecimiento de dimensión en comparación a GMRES(m_j). Esto permite tener un número menor de operaciones matriz-vector.

Al comparar A-LGMRES-E con los métodos con enriquecimiento, ver Figura 5.2, el A-LGMRES-E mejora las tasas de convergencia para estos problemas al reducir los tiempos de ejecución y el número de ciclos de reinicio. Además, GMRES-E(m, d) y LGMRES (m, l) no convergen para los dos últimos problemas (Figura 5.3). Esto muestra que la adición de información de ciclos anteriores con el parámetro de reinicio fijo no es suficiente para obtener la convergencia, por lo que se necesita un ajuste de m para mejorar el descenso de la norma residual relativa. Para el problema cavity06, el A-LGMRES-E no converge para el número máximo de ciclos, pero tiene la norma residual relativa más baja ($\|r_j\|/\|r_0\| < 10^{-5}$). En el caso del problema cavity07, para el cual ningún método de la literatura científica converge, se observa que la propuesta es la única que logra la convergencia.

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS



(a)



(b)

Figura 5.1: Comparación de curvas de convergencia entre dos métodos adaptativos. Norma residual relativa ($\|r_j\|/\|r_0\|$), dimensión del subespacio de búsqueda (m_j para GMRES(m_j) y s_j para A-LGMRES-E($m_j, 1, 3$)) y norma de la variable de control $\|y_j\|$, para las matrices (a) cavity02 y (b) cavity04. **Fuente:** Elaboración propia.

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS

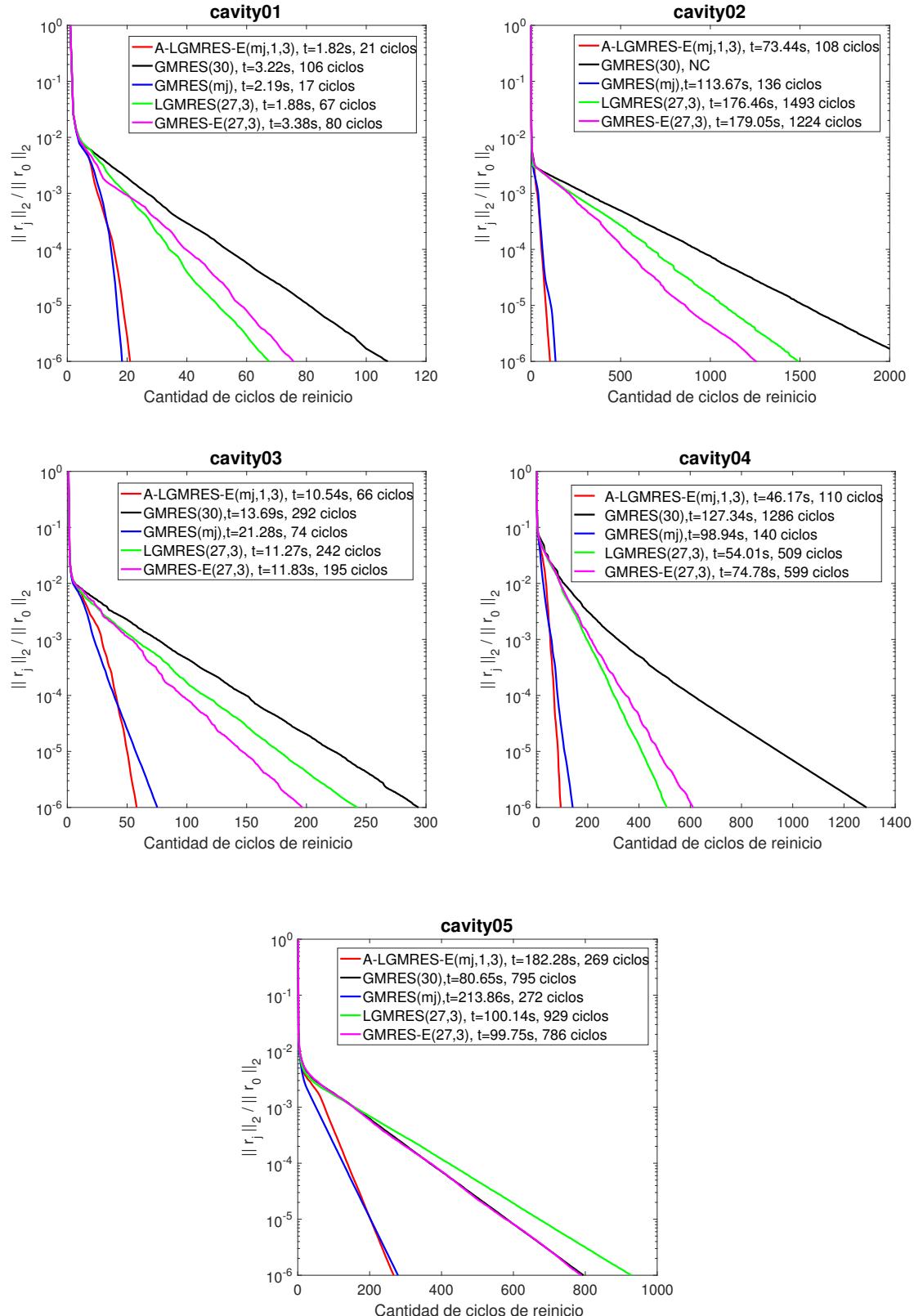


Figura 5.2: Curvas de convergencia para problemas cavity01 a cavity05: norma residual relativa vs. cantidad de ciclos. **Fuente:** Elaboración propia.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

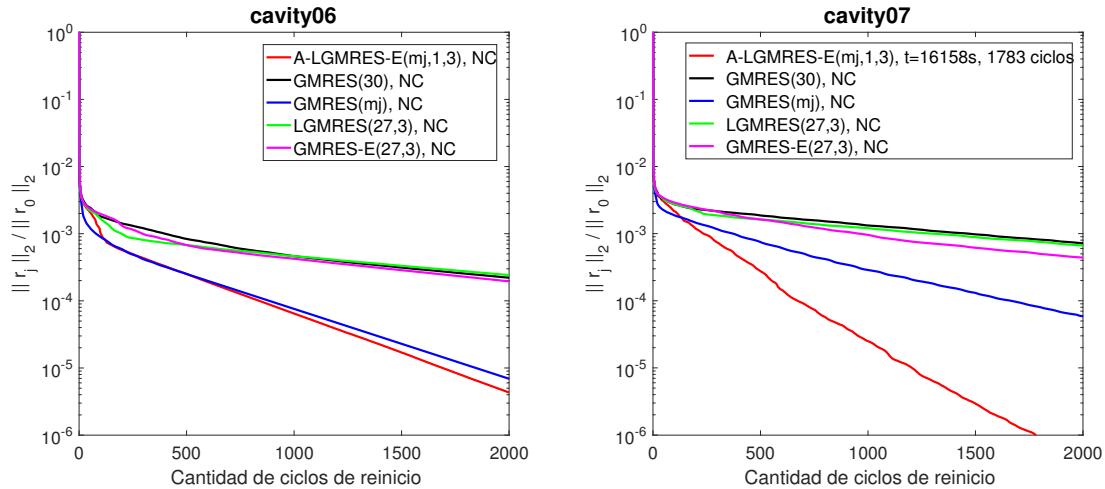


Figura 5.3: Curvas de convergencia para problemas cavity06 y cavity07: norma residual relativa vs. cantidad de ciclos. **Fuente:** Elaboración propia.

CAPÍTULO 6

Conclusiones y trabajos futuros

Se ha implementado una modificación del método iterativo GMRES(m), y su correspondiente código en Matlab, para la resolución de sistemas lineales grandes dispersos no simétricos (o no hermíticos, en el caso de matrices complejas). En la experimentación numérica del método se ha estudiado distintas metodologías, todas ellas variaciones del método mencionado.

El método presentado A-LGMRES-E tiene buenas propiedades de convergencia para diferentes grupos de problemas, tanto con tasas de convergencia rápida como lenta; se consigue una buena competitividad al reducir los tiempos de ejecución y la cantidad de ciclos para la resolución de sistemas ya convergentes, y se logra superar los problemas inherentes a los sistemas lineales de convergencia lenta o estancamiento. Además, aumentar el valor de m cuando un método muestra estancamiento produce en el en el GMRES reiniciado una mejora de la información necesaria para mejorar la convergencia.

En la resolución de matrices asociadas al problema de esparcimiento electromagnético, la resolución con el método propuesto todavía presenta ventajas con respecto a los métodos comparados, hasta el número de onda $k_0 = 8\pi$. Para valores mayores de k_0 el método necesita otras estrategias de control.

A continuación se mencionan otros trabajos futuros necesarios para el avance de esta línea de investigación:

- La robustez y la eficiencia de los métodos iterativos puede ser mejorada aplicando uno o más precondicionadores a la matriz del sistema. Sin em-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

bargo, la elección de las mejores técnicas de precondicionamiento es un área de estudio muy incipiente.

- Modificar de manera adaptativa los parámetros de enriquecimiento del subespacio de búsqueda l y d .
- Migrar el código fuente a arquitecturas heterogéneas CPU/GPU y/o paralelas. La elección de lenguajes de programación como C, C++ o Fortran entre otros será necesaria para acelerar los tiempos de ejecución.
- Estudiar más detalladamente los problemas del repositorio de matrices [6], pertenecientes a otros problemas del Álgebra Lineal Numérica, correspondientes a varias áreas de la Ingeniería: dinámica de fluidos, sistemas de potencia, simulación de circuitos electrónicos, etc.
- Considerar el problema inverso de esparcimiento electromagnético para la determinación de parámetros eléctricos de materiales a partir de la onda esparcida. Los problemas inversos son usualmente mucho más difíciles de abordar que los problemas directos.

APÉNDICE A

Presentaciones nacionales e internacionales

En el marco del proyecto *CABIBESKRY: Control de algoritmos bloque iterativo basado en el Subespacio de Krylov*, los resultados parciales han sido presentados en eventos de iniciación científica nacionales e internacionales. La modalidad elegida fue de presentación tipo poster, con consultas del Comité Científico y del público asistente.

La primera presentación «*Comparison of some strategies for restarted GMRES*» fue realizada en el XXXVII Congreso Nacional de Matemática Aplicada y Computacional (CNMAC) organizado por la Sociedad Brasilera de Matemática Aplicada y Computacional (SBMAC), en la Universidad Federal de Sao Paulo (UNIFESP), ciudad Sao Jose dos Campos, estado de Sao Paulo, Brasil, entre el 19 y el 22 de septiembre de 2017.

La siguiente, denominada «*Application of an Adaptive GMRES(m) on an Electromagnetic Scattering from a Cavity*», fue llevada a cabo en el XXXVIII CNMAC, en la Universidad Estatal de Campinas (UNICAMP), ciudad de Campinas, estado de Sao paulo, Brasil, entre el 17 y el 21 de septiembre de 2018.

Finalmente, el Trabajo «*Aplicación de un GMRES(m) adaptativo a un problema de esparcimiento electromagnético*» fue exhibido en el II Encuentro de Investigadores organizado por la Sociedad Científica del Paraguay (SCP), del 6 al 8 de noviembre de 2018.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Comparison of some strategies for restarting GMRES

Gustavo Espínola¹
Juan C. Cabral²
Pedro Torres López³
Christian E. Schaerer⁴
National University of Asuncion, San Lorenzo, Paraguay

1 Introduction

Restarted Generalized Minimal Residual Method (GMRES(m)) is one of the most successful methods for solving linear system of equations [8]. At each cycle, GMRES(m) uses the residual at the previous cycle as starting guess, and constructs a Krylov subspace of dimension m with $m \ll n$ (where n is the dimension of the linear system) for computing a new residual, which is used as the starting residual for the next cycle, i.e., the next call to a GMRES routine. Rate of GMRES(m) convergence depends on an appropriate election of the restarting parameter m . In this context several algorithms have been proposed for choosing statically and dynamically the parameter m or introducing vectors for enriching the subspace.

2 Models comparison

In this work we compare three strategies (GMRES-E(m, d), LGMRES(m, l) and PD-GMRES(m)) for choosing iteratively an appropriate variation or enrichment of the Krylov subspace for improving GMRES(m) convergence. In [6] the GMRES-E(m, d) was proposed, with d approximate eigenvectors added to the Krylov subspace, and in [7] it was pointed out that augmenting with harmonic Ritz vectors leads to better convergence results. In [1] a new restarted augmented LGMRES(m, l) algorithm was introduced, where the Krylov subspace augments with l previous approximations of the error, while in [4], the PD-GMRES(m) was formulated as a control problem for which m is the control variable to be modified at each cycle by a discrete proportional-derivative controller. The controller has the capacity to augment or deflate the dimension of the Krylov subspace if any convergence problem is detected. In [3] a combination of both feedback control of m and subspace enrichment with a fixed amount of harmonic Ritz vectors was added.

¹gustavoespinala1989@gmail.com

²jccabral19@gmail.com

³torres.pedrozpk@gmail.com

⁴cschaer@pol.una.py

Figura A.1: Resumen sometido al CNMAC 2017, pág. 1. **Fuente:** Elaboración propia.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

3 Conclusion

This is an ongoing undergraduate research work with partial tests on classic problems from the SuiteSparse matrix collection [5], specially on problems that stagnate with the standard GMRES(m). The modified method has good convergence for the described subset of matrices, and an improvement in terms of number of iterations until convergence is observed. A discussion of the appropriate variation of the search subspace is performed for identifying when to use either an acceleration or an overcoming stagnation strategy in contraposition to the standard GMRES(m).

Acknowledgements

CES acknowledges PRONII-CONACyT-Paraguay. This work is partially supported by CIMA through CONACyT grant 14-INV-186.

References

- [1] A. H. Baker, E. R. Jessup, and T. Manteuffel, *A Technique for Accelerating the Convergence of Restarted GMRES*, SIAM Journal on Matrix Analysis and Applications, 1995, Vol. 26 , Issue 4, pp. 962-984, DOI:10.1137/S0895479803422014.
- [2] A. Bhaya and E. Kaszkurewicz, *Control Perspectives on Numerical Algorithms and Matrix Problems*, Society for Industrial and Applied Mathematics, Philadelphia, 2006.
- [3] J. C. Cabral and C. Schaefer, *Harmonic Ritz control strategy for restarting GMRES(m)*, 3rd. Conference of Computational Interdisciplinary Sciences, pp. 133-138, 2014.
- [4] R. Cuevas, C. Schaefer and A. Bhaya., *A control inspired strategy for varying the restart parameter m of GMRES(m)*, Congresso Nacional de Matematica Aplicada e Computacional - CNMAC XXXIII, pp. 1000-1001, 2010.
- [5] T. Davis and Y. Hu, *The SuiteSparse Matrix Collection (formerly known as the University of Florida Sparse Matrix Collection)*, University of Florida and AT&T Research. Available online at <http://www.cise.ufl.edu/research/sparse/matrices/>.
- [6] R. B. Morgan, *A restarted GMRES method augmented with eigenvectors*, SIAM Journal on Matrix Analysis and Applications, 1995, Vol. 16 , Issue 4, pp. 1154-1171, DOI: 10.1137/S0895479893253975.
- [7] R. B. Morgan, *Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations*, SIAM Journal on Matrix Analysis and Applications, 1995, Vol. 21 , Issue 4, pp. 1112-1135, DOI: 10.1137/S0895479897321362.
- [8] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual method for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 1986, Vol. 7, No. 3, pp. 856-869, DOI:10.1137/0907058.

Figura A.2: Resumen sometido al CNMAC 2017, pág. 2. **Fuente:** Elaboración propia.




Comparison of some strategies for Restarting GMRES

Gustavo Espínola Mena¹, Juan C. Cabral², Pedro Torres³, Christian E. Schaefer⁴
 1,2,3,4National University of Asuncion - UNA



Introduction

The Restarted Generalized Minimal Residual Method (GMRES(m)) is one of the most successful methods for solving linear systems of equations $Ax = b$, where A is a nonsymmetric sparse matrix[6]. At each cycle, GMRES(m) uses the residual at the previous cycle as starting guess, and constructs a Krylov subspace of dimension m ($m \ll n$ where n is the dimension of the linear system) for computing a new residual, which is used as the starting residual for the next cycle, i.e., the next call to a GMRES routine. Rate of GMRES(m) convergence depends on an appropriate selection of the restarting parameter m . In this context several algorithms have been proposed for choosing statically and dynamically the parameter m or introducing vectors for enriching the subspace [2, 3].

Numerical results

Models comparison

In this work we compare the performance of the proposed method called Adaptive-GMRES with the standard GMRES(m) and other methods that try to accelerate the convergence. These methods are:

- GMRES- $E(m, d)$ method proposed by R. B. Morgan [5], improves the convergence by appending d approximate eigenvectors to the Krylov subspace.
- LGMRES(m, l) method proposed by A. H. Baker [1], improves the convergence by appending l error approximation vectors.

Control formulation

At each cycle, GMRES(m) finds a solution of the form

$$x_j = x_{j-1} + V_m y_j, \quad (1)$$

where x_{j-1} is the previous approximate solution of x , and the residual is $r_{j-1} = b - Ax_{j-1}$; then V_m is a $n \times m$ matrix where its columns form an orthogonal basis of the Krylov Subspace $K_m(A, r_{j-1}) \equiv \text{span}\{r_{j-1}, Ar_{j-1}, A^2r_{j-1}, \dots, A^{m-1}r_{j-1}\}$. Furthermore, y_j minimize the l_2 -norm of the residual $\|r_j\|_2 = \|b - A(x_{j-1} + V_m y_j)\|_2 = \|\beta e_j - \tilde{H}_m y_j\|_2$.

When the l_2 -norm of the last y_j is very small, then $x_j \approx x_{j-1}$ and stagnation occurs. Hence, the proposed strategy called Adaptive-GMRES(m) consists in modifying the value of m before each restarted cycle.

An example of a proportional controller for m is given by:

$$m_j = m_{j-1} + u_j, \quad (2)$$

where

$$u_j = \begin{cases} 1 & \text{if } \|y_j\|_2 < \epsilon_0 \\ 0 & \text{if } \|y_j\|_2 \geq \epsilon_0 \end{cases} \quad (3)$$

Selected problems

Partial tests on classic problems from the SuiteSparse matrix collection [4] are performed. For the Group A, GMRES(m) converges before 2000 restart cycles, and for Group B, GMRES(m) does not converge before 2000 restart cycles. n is the size of A , nnz is the number of nonzero elements in A and $\text{cond}(A)$ is the condition number of A .

Problem Group A	n	nnz	Application area	cond(A)
A1 add20	2395	17319	circuit simulation problem	12047.1
A2 cavity05	1182	32632	computational fluid dynamics problem	577065
A3 circuit_2	4510	21199	circuit simulation problem	131925
A4 tppg.trans.01	1220	7382	circuit simulation problem	12214.3
A5 memplus	17758	99147	circuit simulation problem	129436
A6 sherman4	3312	20793	computational fluid dynamics problem	2178.63

Conclusion

The Adaptive-GMRES(m) method has good convergence properties for both groups of problems. We show that increasing the value of m when we have stalling improves the information in the restarted GMRES. The criterion of increasing the value of m when the value of $\|y_j\|_2$ is small, allows to avoid slow convergences and stagnations in standard GMRES(m). Future work may find better heuristics for u_j in order to reduce execution times.

Acknowledgements

GEE acknowledges the technical support given by Polytechnic School, UNA and the financial support given by CIMA. CES acknowledges PRONI-COACyT-Paraguay. This work is partially supported by CIMA through CONACyT grant 14-INV-186 CABIBESKRY-PROCIENCIA.

References

- [1] A. H. Baker, E. R. Jessup, and T. Manteuffel, *A Technique for Accelerating the Convergence of Restarted GMRES*, SIAM Journal on Matrix Analysis and Applications, 1995, Vol. 26 , Issue 4, pp. 962-984, DOI:10.1137/S0895479803422014.
- [2] J. C. Cabral and C. Schaefer, *Harmonic Ritz control strategy for restarting GMRES(m)*, 3rd. Conference of Computational Interdisciplinary Sciences, pp. 133-138, 2014.
- [3] R. Cuevas, C. Schaefer and A. Bhaya., *A control inspired strategy for varying the restart parameter m of GMRES(m)*, Congreso Nacional de Matemática Aplicada e Computacional - CNMAC XXXIII, pp. 1000-1001, 2010.
- [4] T. Davis and Y. Hu, *The SuiteSparse Matrix Collection (formerly known as the University of Florida Sparse Matrix Collection)*, University of Florida and AT&T Research. Available online at <http://www.cise.ufl.edu/research/sparse/matrices/>.
- [5] R. B. Morgan, *A restarted GMRES method augmented with eigenvectors*, SIAM Journal on Matrix Analysis and Applications, 1995, Vol. 16 , Issue 4, pp. 1154-1171, DOI: 10.1137/S0895479893253975.
- [6] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual method for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 1986, Vol. 7, No. 3, pp. 856-869, DOI:10.1137/0907058.

http://www.cc.pol.una.py/

XXXVII CNMAC, September 18-22, 2017

gustavoespinal1989@gmail.com

Figura A.3: Póster presentado al CNMAC 2017. Fuente: Elaboración propia.



Figura A.4: Certificado de participación en el CNMAC 2017. **Fuente:** SBMAC.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Application of an Adaptive GMRES(m) on an Electromagnetic Scattering Problem from a Cavity

Gustavo Espínola¹

Juan C. Cabral²

Christian E. Schaeerer³

National University of Asuncion, San Lorenzo, Paraguay

1 Introduction

In this work an adaptive GMRES(m) solver is applied to an electromagnetic scattering problem from a cavity embedded in a ground plane [3,4]. The problem is focused on a 2-D geometry by assuming that the medium is invariant in the z -direction and nonmagnetic with constant magnetic permeability $\mu(x,y) = \mu_0$. The ground plane (x axis) and the wall of the cavity are perfect electric conductors, and the interior of the cavity is filled with inhomogeneous material characterized with its relative permittivity $\epsilon_r(x,y)$.

For a TM (transverse magnetic) polarization, in which the magnetic field is transverse to the invariant direction and the electric field is $E = (0, 0, u(x, y))$, the approximation to the cavity problem consists of the discrete Helmholtz equation (1) together with a Sommerfeld's radiation condition imposed at infinity -see equation (3)-, in order to reduce the problem to the following bounded-domain problem:

$$\Delta u + k_0^2 \epsilon_r u = f, \text{ in } \Omega = [0, a] \times [b, 0], \quad (1)$$

$$u = 0, \text{ on } S, \quad (2)$$

$$\partial_n u = \mathcal{T}(u) + g, \text{ on } \Gamma \quad (3)$$

where k_0 is the free space wave number, Ω is the problem domain, f is the source term and $f = 0$ in the free space, S denotes the walls of cavity, \mathcal{T} is a nonlocal boundary operator, Γ is the aperture between the cavity and the free space and $g(x) = -2i\beta e^{i\alpha x}$.

2 Methodology

GMRES(m) is frequently selected for its robustness in problems whose discretization results in a large sparse non-hermitian linear system [5]. GMRES(m) approximates the

¹gustavoespino1989@gmail.com

²jcabral19@gmail.com

³cschaer@pol.una.py

Figura A.5: Resumen sometido al CNMAC 2018, pág. 1. **Fuente:** Elaboración propia.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

solution of the subsequent linear system $Au = f$ at each cycle j as follows [2]:

$$z_j = GMRES(A, r_{j-1}, m) \quad (4)$$

$$u_j = u_{j-1} + z_j \quad (5)$$

$$r_j = f - Au_j \quad (6)$$

where u_j is the approximate solution at the cycle. When its main parameter m is fixed, the convergence is not guaranteed, causing possibly stagnation [5].

An adaptive control strategy based on switching between at least two Krylov subspace methods is performed. The search subspace is enriched with information vectors of previous cycles by using error approximation vectors in an augmented GMRES(m). If stagnation occurs, the latest error approximation vectors are discarded because they do not help to avoid the stagnation, hence we return to standard GMRES(m) increasing the restart parameter m by a constant. The new approximate solution is found in a $(m+k)$ -dimensional subspace, constructed by the Krylov Subspace using the new restart parameter and augmented with k recently generated error approximations. Generally, $k \leq 3$ and performance decreases for larger k [1]. For this work, we choose $k = 3$.

3 Conclusion

Numerical experiments for different discrete domain sizes and values of wave number k_0 are compared. Preliminary results show that a switching strategy is good enough to improve convergence when comparing with other iterative methods with fixed parameters. However, the computation is specially challenging when k_0 is increasing.

References

- [1] A. H. Baker, E. R. Jessup and T. Manteuffel, A Technique for Accelerating the Convergence of Restarted GMRES, *SIAM Journal on Matrix Analysis and Applications*, Vol. 26, No. 4, pp. 962-984, DOI:10.1137/S0895479803422014
- [2] J. C. Cabral, P. Torres and C. E. Schaerer, On control strategy for stagnated restarting GMRES(m), *Proceedings of the 4th. Conference of Computational Interdisciplinary Sciences*, 2016.
- [3] K. Du, GMRES with adaptively deflated restarting and its performance on an electromagnetic cavity problem, *Applied Numerical Mathematics*, Vol. 61, No. 9, 2011. DOI: 10.1016/j.apnum.2011.04.003.
- [4] J. M. Jin, *The Finite Element Method in Electromagnetics*, 3rd. ed., Wiley-IEEE Press, 2015.
- [5] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual method for solving nonsymmetric linear systems, *SIAM Journal on Scientific and Statistical Computing*, 1986, Vol. 7, No. 3, pp. 856-869, DOI:10.1137/0907058.

Figura A.6: Resumen sometido al CNMAC 2018, pág. 2. **Fuente:** Elaboración propia.

Application of an Adaptive GMRES(m) on an Electromagnetic Scattering from a Cavity

Gustavo E. Espínola^a, Juan C. Cabral^b & Christian E. Schaeerer^{b,c}

^aEngineering School, National University of Asuncion. SL, Central, Paraguay.

^bPolytechnic School, National University of Asuncion. P.O.Box: 2111 SL, Central, Paraguay.

^cCentro de Investigación en Matemática - CIMA, César López Moreira Street 693, Asunción, Paraguay.
gustavoespinala1989@gmail.com (G.E. Espínola), jccabral19@gmail.com (J.C. Cabral), chris.schaeerer@cima.org.py (C.E. Schaeerer)

CNMAC 2018

XCVIII Congresso Nacional de Matemática Aplicada e Computacional

17 a 21 de Setembro de 2018 - IMECC/UNICAMP - Campinas - SP

Abstract
In order to use the Restarted GMRES method on a 2-D system, an adaptive solver is applied on an electromagnetic scattering problem from a cavity embedded in a ground plane. The proposed method combines two strategies: modifying the dimension m of the Krylov search subspace and augmenting it with information vectors from previous cycles. Numerical results for selected matrices suggest that this method could, for some problems, overcome the stagnation caused by a standard GMRES(m) with m fixed.

Introduction

The electromagnetic problem is focused on a 2-D geometry by assuming that the medium is invariant in the z -direction and nonmagnetic with constant magnetic permeability $\mu(x, y) = \mu_0$. The ground plane (x axis) and the wall of the cavity are perfect electric conductors, and the interior of the cavity is filled with inhomogeneous material characterized with its relative permittivity $\epsilon_r(x, y)$ [2].

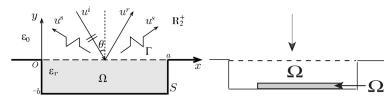


Figure 1: A rectangular cavity geometry, $\epsilon_r = 1$ in $\Omega \setminus \Omega_1$ and $\epsilon_r = 2$ in Ω_1 [2, 5].

For a TM (transverse magnetic) polarization, in which the magnetic field is transverse to the invariant direction and the electric field is $E = (0, 0, u(x, y))$, the approximation to the cavity problem consists of the discrete Helmholtz equation (1) together with a Sommerfeld's radiation condition imposed at infinity - see equation (3):

$$\Delta u + k_0^2 \epsilon_r u = f, \quad \text{in } \Omega = [0, a] \times [b, 0], \quad (1)$$

$$u = 0, \quad \text{on } S, \quad (2)$$

$$\partial_n u = T(u) + g, \quad \text{on } \Gamma \quad (3)$$

where k_0 is the free space wave number. Ω is the problem domain, f is the source term and $f = 0$ in the free space. S denotes the walls of cavity. T is a nonlocal boundary operator, Γ is the aperture between the cavity and the free space and $g(x) = -2h\beta e^{j\omega x}$.

The discrete finite difference system for 1 is given by:

$$\frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h_x^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h_y^2} + k_0^2 \epsilon_r(x_i, y_j) u_{i,j} = f(x_i, y_j) \quad (4)$$

for $i = 1, 2, \dots, M$, $j = 1, 2, \dots, N$.

By 2 we have

$$u_{i,j} = u_{M+1,j} = u_{i,0}, \quad i = 1, 2, \dots, M, \quad j = 1, 2, \dots, N \quad (5)$$

The discrete form of the nonlocal boundary condition 3 is given by

$$\frac{u_{i,N+1} - u_{i,N}}{h_y} = \sum_{j=1}^M g_j u_{i,N+1} + g(x_i), \quad i = 1, 2, \dots, M \quad (6)$$

For more details of the system of equations, see [2].

Methodology

GMRES(m) is frequently selected for its robustness in problems whose discretization results in a large sparse non-hermitian linear system [4]. This method approximates the solution of the subsequent linear system $Au = f$ at each cycle j . Nonetheless, the convergence to the solution is not guaranteed when the selection of the fixed parameter m is not appropriate, causing possibly stagnation.

An adaptive control strategy -called here as **A-LGMRES-E**(m, l, d) and based on switching between other GMRES alternatives found in the scientific literature, is performed to overcome the stagnation.

For A-LGMRES-E, the search subspace is enriched with information vectors of previous cycles by using error approximation vectors and harmonic Ritz vectors in an augmented GMRES(m). If stagnation occurs, the latest error approximation vectors do not help to avoid the stagnation, hence we return to standard GMRES(m) increasing the restart parameter m by a constant α . The new approximate solution is found in a $(m + l + d)$ -dimensional Krylov subspace using the new restart parameter and augmented with l recently generated error approximations and d harmonic Ritz vectors. Generally, $l \leq 3$ and performance decreases for larger l [1]. For this work, we choose $l = 1$ and $d = 3$.

This strategy results in the following structure for each cycle:

$$y_j = \text{A-LGMRES-E}(A, r_{j-1}, s_{j-1}), \quad (7)$$

$$m_j = m_{j-1} + \alpha \text{ if } \|y_j\|_2 < \delta_0; m_{j-1} \text{ else}, \quad (8)$$

$$u_j = u_{j-1} + W_j y_j, \quad (9)$$

$$r_j = f - Au_j, \quad (10)$$

where y_j is the GMRES vector that minimizes the residual norm $\|r_j\|_2$. W_j is a $n \times s_j$ matrix containing a $s_j = m_j + d + l$ Krylov subspace enriched with information from previous cycles , and u_j is the approximate solution at the cycle.

Numerical experiments

The system $Au = f$ was solved with the following iterative methods: GMRES (m), GMRES (m_j) with variable parameter according to equation (8), GMRES-E (m, d) [3], LGMRES (m, l) [1] and A-LGMRES-E (m_j, l, d). The initial estimate of the solution is $x_0 = 0$, the stopping criterion on the

relative residual norm is $\frac{\|r_0\|_2}{\|f\|_2} < 10^{-6}$ or a maximum amount of 2000 restarts cycles. Parameters: GMRES (m): $m = 30$; GMRES (m_j): $m_0 = 30$; LGMRES (m, l): $m = 27, l = 3$; GMRES-E (m, d): $m = 27, d = 3$; A-LGMRES-E (m_j, l, d): $m_0 = 26, l = 1, d = 3, \delta_0 \in [0.1, 1]$ and $\alpha \in \{2, 3, \dots, 6\}$. The reported execution times are the average of 5 executions and are presented in the Table 2.

A	size(A)	cond(A)	M	N	k_0
cavity01	390	1.1974e+03	39	9	2π
cavity02	2475	1.2469e+04	99	24	2π
cavity03	885	1.8851e+03	59	14	4π
cavity04	2475	4.5310e+04	99	24	4π
cavity05	2475	8.1072e+03	99	24	8π
cavity06	9950	1.4982e+05	199	49	8π
cavity07	22425	2.8573e+04	299	74	10π

Table 1: Selected matrices [5].

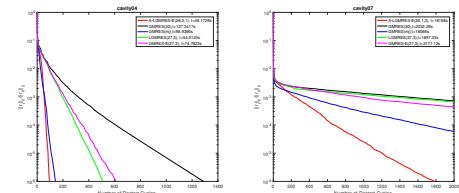


Figure 2: Convergence curves: relative residual norm v. number of restart cycles. For these cases, $\delta_0 = 0.5, \alpha = 4$.

GMRES(m)	GMRES(mj)		LGMRES(m,l)		GMRES-E(m,d)		A-LGMRES-E	
	Time (s)	MVPs	Time (s)	MVPs	Time (s)	MVPs	Time (s)	MVPs
cavity01	3.22	3180	2.19	1054	1.88	2001	3.38	2247
cavity02	243.97*	60000*	113.67	12954	176.46	44781	179.05	37707
cavity03	13.69	8760	21.28	6754	11.27	7251	11.83	5877
cavity04	127.34	38580	98.94	13354	54.01	15261	74.78	17787
cavity05	80.65	23790	213.86	27054	100.14	27831	99.75	23637
cavity06	1078.94*	60000*	1057.15*	199354*	1014.26*	60000*	1260.76*	60000*
cavity07	2032.28*	60000*	18066*	199354*	1897.33*	60000*	2177.12*	60000*

Table 2: Metrics for selected matrices and iterative methods: time in seconds and number of matrix-vector products (MVPs). For the cases highlighted with (*), the method does not reach convergence before 2000 cycles.

Conclusions

An adaptive criterion is performed for identifying stagnation in a standard Restarted GMRES and appending new information vectors when this occurs. Preliminary results show that the proposal is good enough to improve convergence when comparing with other iterative methods with either fixed parameters and enriched subspaces or adaptive parameters without subspace enriching. However, the computation is specially challenging when k_0 is increasing. A more exhaustive research is necessary in linear systems with large k_0 to identify when is more convenient either modifying the restart parameter m or executing an augmented GMRES(m), according to the linear system metrics.

References

- [1] A. Baker, E. Jessup, and T. Manteuffel. A technique for accelerating the convergence of restarted GMRES. *SIAM Journal on Matrix Analysis and Applications*, 26(4):962–984, 2005.
- [2] Kui Du. GMRES with adaptively deflated restarting and its performance on an electromagnetic cavity problem. *Applied Numerical Mathematics*, 61(9):977 – 988, 2011.
- [3] R. Morgan. A restarted GMRES method augmented with eigenvectors. *SIAM Journal on Matrix Analysis and Applications*, 16(4):1154–1171, 1995.
- [4] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.
- [5] Yingxi Wang, Kui Du, and Weiwei Sun. Preconditioning iterative algorithm for the electromagnetic scattering from a large cavity. *Numerical Linear Algebra with Applications*, 16(5):345–363.

Acknowledgements

GEE thanks the technical support given by the Polytechnic School, UNA, and the financial support given by the Mathematics Research Center (CIMA). JCC acknowledges the financial support given by the Paraguayan National Council of Science and Technology (CONACYT) through scholarship Prociencia-2015. CES thanks to the PRONI CONACyT program.

Figura A.7: Póster presentado en el CNMAC 2018. Fuente: Elaboración propia.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**



Figura A.8: Certificado de participación en el CNMAC 2018. **Fuente:** SBMAC.

Aplicación de un GMRES(m) adaptativo a un problema de esparcimiento electromagnético

Gustavo E. Espínola¹; Juan C. Cabral²; Christian E. Schaerer³

¹ Facultad de Ingeniería, Universidad Nacional de Asunción, gustavoespinala1989@gmail.com;

² Facultad Politécnica, Universidad Nacional de Asunción, jccabral19@gmail.com;

³ Centro de Investigación en Matemática, chris.schaerer@cima.org.py.

INTRODUCCIÓN

El Algoritmo Generalizado de Residuo Mínimo con reinicios GMRES(m) es uno de los métodos más robustos para sistemas de ecuaciones lineales grandes dispersos no hermíticos, pero presenta problemas de estancamiento si su parámetro principal m no es elegido apropiadamente. Se aplica un método iterativo adaptativo donde se modifica la dimensión m del subespacio de búsqueda de Krylov y se enriquece el mismo con vectores de información de ciclos previos. Resultados numéricos para matrices seleccionadas de un problema de esparcimiento electromagnético sobre una cavidad rectangular sugieren que este método podría superar los problemas causados con un GMRES(m) estándar con parámetro fijo.

METODOLOGÍA

En cada ciclo, GMRES(m) [4] encuentra una solución al sistema lineal $Ax = b$ mediante la siguiente recurrencia:

$$x_j = x_{j-1} + V_j^{(m)} y_j,$$

donde x_j y x_{j-1} son vectores de solución aproximada hallados en ciclos consecutivos, $V_j^{(m)}$ es una matriz cuyos m vectores columna forman una base ortogonal del subespacio de búsqueda de Krylov, y y_j es un vector que minimiza la norma euclídea del residuo

$$\|r_j\| = \|b - Ax_j\|.$$

Si la norma euclídea del vector y_j es muy pequeña, entonces $x_j \approx x_{j-1}$ y se produce un estancamiento.

En el método propuesto, A-LGMRES-E, se modifica el parámetro m a través de una técnica inspirada en Teoría de Control, y además se enriquece con vectores de aproximación al error [1] y vectores armónicos de Ritz [3].

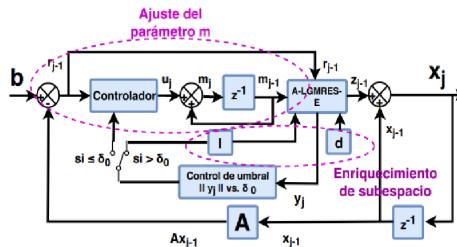


Figura 01: Diagrama de bloques para el método adaptativo propuesto

El método es puesto a prueba sobre un problema de esparcimiento electromagnético en una cavidad rectangular incrustada en un conductor eléctrico perfecto. Para una onda polarizada TM, en la cual el campo eléctrico es $E = (0, 0, u(x, y))$, la aproximación al problema de la cavidad consiste en la ecuación de Helmholtz discretizada por diferencias finitas, junto con una condición de radiación de Sommerfeld [2].

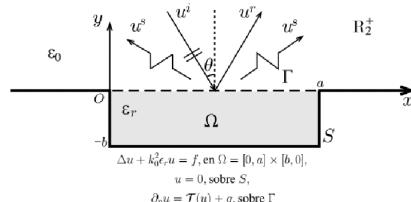


Figura 02: Geometría y ecuaciones del problema de esparcimiento electromagnético

RESULTADOS NUMÉRICOS

El sistema lineal $Ax = b$ fue resuelto con los siguientes métodos iterativos: GMRES(m), GMRES(m_j), LGMRES(m, l), GMRES-E(m, d) y A-LGMRES-E(m_j, d_j). La estimación inicial de la solución es $x_0 = 0$, el criterio de parada sobre la norma euclídea residual relativa es $\|\eta_j\|/\|r_0\| < 10^{-6}$ o un máximo de 2000 ciclos de reinicio.

	A	size(A)	cond(A)	M	N	k_0
cavity01	390	1.1974e+03	39	9	2π	
cavity02	2475	1.2469e+04	99	24	2π	
cavity03	885	1.8851e+03	59	14	4π	
cavity04	2475	4.5310e+04	99	24	4π	
cavity05	2475	8.1072e+03	99	24	8π	
cavity06	9950	1.4982e+05	199	49	8π	
cavity07	22425	2.8573e+04	299	74	10π	

Tabla 01: Matrices elegidas

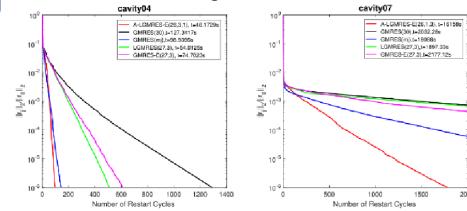


Figura 02: Curvas de convergencia: norma residual vs. cantidad de ciclos de reinicio

	GMRES(m)	GMRES(m_j)	LGMRES(m, l)	GMRES-E(m, d)	A-LGMRES-E
Time (s)	3.22	3.180	2.19	1054	1.88
MVPs					2001
Time (s)	243.97	600*	113.67	12954	176.46
MVPs					44781
Time (s)	870.41	1700*	671.14	72954	880.07
MVPs					1431
Time (s)	127.34	38589	24.89	12354	54.01
MVPs					12551
Time (s)	23990	213.86	27054	100.14	27831
MVPs					99.75
Time (s)	30.65	80.65	101.26	60000*	23637
MVPs					182.28
Time (s)	708.94	60000*	199354*	1014.26	23713
MVPs					8464.02
Time (s)	2032.28	60000*	181666*	1996354*	1897.33
MVPs					60000*
Time (s)	1078.44	60000*	1996354*	1897.33	2177.12
MVPs					16158
Time (s)	2032.28	60000*	181666*	1996354*	178011

Tabla 02: Métricas para diferentes matrices y métodos iterativos probados

CONCLUSIÓN

La propuesta permite aumentar y enriquecer el subespacio de búsqueda de solución, manteniendo propiedades de la norma residual mínima. Usando una regla de comutación apropiada se busca evitar convergencias lentas e inclusive el estancamiento. Es un método robusto que preserva las buenas características de los métodos de enriquecimiento, logrando la convergencia inclusiva donde otros no.

AGRADECIMIENTOS



REFERENCIAS

- [1] A. Baker, E. Jessup, and T. Manteuffel. A technique for accelerating the convergence of restarted GMRES.
- [2] Kui Du. GMRES with adaptively deflated restarting and its performance on an electromagnetic cavity problem.
- [3] R. Morgan. A restarted GMRES method augmented with eigenvectors.
- [4] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems.

Figura A.9: Póster presentado en el II Encuentro de Investigadores de la SCP.
Fuente: Elaboración propia.

Gustavo Enrique Espínola Mena

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**



Figura A.10: Certificado de participación en el II Encuentro de Investigadores.
Fuente: SCP.

APÉNDICE B

Código fuente

```
1 function [xx]=ALGMRESE(A, b, m_min, m_max, d, l, max_ciclos
2 , tol, alpha, delta)
3 %%%
4 %Argumentos de entrada:
5 %A: matriz de coeficientes del sistema lineal
6 %b: vector de constantes del sistema lineal
7 %m_min: valor minimo del parametro m
8 %m_max: valor maximo del parametro m
9 %d: cantidad de vectores armonicos de Ritz
10 %l: cantidad de vectores de aproximacion de error
11 %max_ciclos: cantidad maxima de ciclos de ejecucion
12 % permitidos
13 %tol: umbral de tolerancia para la norma residual relativa
14 %alpha: aumento del parametro m
15 %
16 %Valores de salida:
17 %xx: solucion del sistema lineal, que cumple con el
18 % requisito de la norma
19 %residual relativa:
20 % || rj || / || r0 || < tol
21 %%%
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
19 opts.tol = eps;
20 x0 = zeros(n,1);
21 m = m_min;
22 s = m + d + 1;
23 ST = s;
24 flag = 0; %Para verificar convergencia
25 flag2 = 0; %Para la conmutacion de estrategias de control
26 restart = 1; %Contador de los j ciclos de reinicio
27 r0 = b - A*x0; %Primer residuo
28 beta = norm(r0);
29 logres(1,:) = 1; %||rj|| /||r0||=1 en j = 1
30 iter(1,:) = restart;
31 log_de_m(iter(size(iter,1),:),1) = m;
32 log_de_s(iter(size(iter,1),:),1) = m; %Inicialmente no hay
     d ni l
33 z = [];
34 %%%
35 %SOLVER
36 % Iniciar cronometro
37 inicio = cputime;
38 v(:,1) = r/beta;
39 %Metodo de Arnoldi: creacion de una base ortonormal de
40 %{ r0 , Ar0 , A^2 r0 , A^3 r0 ,..., A^(m-1) r0 }
41 for j = 1:m
42     w(:,j) = A*v(:,j);
43     for i = 1:j
44         h(i,j) = w(:,j)'*v(:,i);
45         w(:,j) = w(:,j)-h(i,j)*v(:,i);
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
46 end
47 h( j+1 ,j ) = norm(w(:, j )) ;
48 if h( j+1 ,j ) == 0
49 m = j ;
50 h2 = zeros(m+1 ,m) ;
51 for k = 1:m
52 h2(:, k) = h(:, k) ;
53 end
54 h = h2 ;
55 else
56 v(:, j+1) = w(:, j ) /h( j+1 ,j ) ;
57 end
58 end
59 H0 = h ;
60 g = zeros(m+1 ,1) ;
61 g(1 ,1) = beta ;
62
63 for j = 1:m % Resolucion del problema de minimos cuadrados
64 P = eye(m+1) ;
65 sin = h( j+1 ,j )/( sqrt(h( j+1 ,j )^2 + h(j ,j )^2)) ;
66 cos = h(j ,j )/( sqrt(h( j+1 ,j )^2 + h(j ,j )^2)) ;
67 P(j ,j ) = cos ;
68 P(j+1 ,j+1) = cos ;
69 P(j ,j+1) = sin ;
70 P(j+1 ,j ) = -sin ;
71 h = P*h ;
72 g = P*g ;
73 end
```

```
74
75 R = zeros(m,m);
76 G = zeros(m,1);
77 V = zeros(n,m);
78 for k = 1:m
79     G(k) = g(k);
80     V(:,k) = v(:,k);
81     for i = 1:m
82         R(k,i) = h(k,i);
83     end
84 end
85 yj = R\G;
86 norm_yj = norm(yj);
87 Z = V*yj; %Vector de aproximacion del error
88 xj = x0+Z;
89 x0 = xj;
90
91 if norm_yj < delta %Para el caso de estancamiento en el
92     primer ciclo.
93     flag2 = 1; %Los vectores de error de aproximacion no
94     sirven
95 else
96     flag2 = 0;
97 end
98 logres(size(logres,1)+1,:)= norm(b-A*xj)/res(1,1);
99 lognormy(size(logres,1)+1,:)= norm_yj;
100
101 %Almacenamiento del vector aproximacion de error
```

DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS

```

100 %para el siguiente ciclo , si es necesario
101 z(:, size(z,2)+1)= Z;
102
103 %Calculo de vectores armonicos de Ritz
104 H0(m+1,:)= [];
105 Fold = H0';
106 opts.v0 = ones(size(Fold,1),1);
107 gf = R'*R;
108 [E2,D2] = eigs(Fold,gf,d,'LM',opts);
109 for pq = 1:d
110     D(pq,1) = abs(D2(pq,pq));
111 end
112 [Y,I] = sort(D,1);
113 for qq = 1:d
114     E(:,qq) = E2(:,I(qq,1));
115 end
116 dy0 = V*E;
117 dy = [];
118 %%%Si dy0 es complejo
119 if isreal(dy0) == 0
120     ij = 1;
121     while size(dy,2) <= d && ij <= d
122         if isreal(dy0(:,ij)) == 0
123             dy(:,size(dy,2)+1) = real(dy0(:,ij));
124         if ij <= d
125             dy(:,size(dy,2)+1) = abs(imag(dy0(:,ij)))*
126             sqrt(1));
127         if ij < d
128             dy(:,size(dy,2)+1) = abs(imag(dy0(:,ij)));
129         end
130     end
131 end

```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
127 if dy0(:, ij) == conj(dy0(:, ij+1))  
128 ij = ij+2;  
129 else  
130 ij = ij+1;  
131 end  
132 end  
133 end  
134 else  
135 dy(:, size(dy,2)+1) = dy0(:, ij);  
136 ij = ij+1;  
137 end  
138 end  
139 else  
140 dy = dy0;  
141 end  
142 E = [];  
143 Ev0 = [];%para comando opts.v0  
144 while flag == 0 %Mientras no se alcance convergencia.  
145 if flag2 == 1  
146 if m < m_max  
147 m = m + alpha;  
148 else  
149 m = m_max;  
150 end  
151 %No se agregan los l vectores de aproximacion de  
error  
152 s = m + d;  
153 ST = s;
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
154     end  
155  
156     v = [] ;  
157     r = b-A*x0 ;  
158     beta = norm(r) ;  
159     P=[] ;  
160     h=[] ;  
161     W=[] ;  
162     V=[] ;  
163     if size(logres,1) <= l+1  
164         s = s-l ;  
165         for j=1:s %Metodo de Arnoldi modificado  
166             if j<=m  
167                 w(:,j) = A*v(:,j) ;  
168             else  
169                 w(:,j) = A*dy(:,j-m) ;  
170             end  
171             for i=1:j  
172                 h(i,j) = w(:,j)'*v(:,i) ;  
173                 w(:,j) = w(:,j)-h(i,j)*v(:,i) ;  
174             end  
175             h(j+1,j) = norm(w(:,j)) ;  
176             if h(j+1,j) == 0  
177                 s = j ;  
178                 h2 = zeros(s+1,s) ;  
179                 for k = 1:s  
180                     h2(:,k) = h(:,k) ;  
181                 end
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
182          h = h2;
183      else
184          v(:, j+1) = w(:, j) / h(j+1, j);
185      end
186  end
187 H0 = h;
188 hs = h;
189 g = zeros(s+1, 1);
190 g(1, 1) = beta;
191 for j = 1:s    % Resolucion del problema de minimos
192     cuadrados
193         P = eye(s+1);
194         sin = h(j+1, j) / (sqrt(h(j+1, j)^2 + h(j, j)^2));
195         cos = h(j, j) / (sqrt(h(j+1, j)^2 + h(j, j)^2));
196         P(j, j) = cos;
197         P(j+1, j+1) = cos;
198         P(j, j+1) = sin;
199         P(j+1, j) = -sin;
200         h = P*h;
201         g = P*g;
202     end
203 R = zeros(s, s);
204 G = zeros(s, 1);
205 V = zeros(n, s);
206 for k = 1:s
207     G(k) = g(k);
208     V(:, k) = v(:, k);
209     for i = 1:s
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
209          R(k,i) = h(k,i);  
210      end  
211      end  
212      yj = R\G;  
213      norm_yj = norm(yj);  
214      for k = m+1:s  
215          V(:,k) = dy(:,k-m);  
216      end  
217      aux = V*yj;  
218      xj = x0+aux;  
219      Z = z;  
220      if l == 1  
221          z = aux;  
222      else  
223          for j = 2:size(logres,1)-1  
224              z(:,j-1) = Z(:,j);  
225          end  
226          z(:,l) = aux;  
227      end  
228      iter(restart+1,:) = restart+1;  
229      log_de_m(iter(size(iter,1),:),1) = m;  
230      log_de_s(iter(size(iter,1),:),1) = s;  
231      logres(size(logres,1)+1,:) = norm(b-A*xj)/res(1,1);  
232      lognormy(size(logres,1)+1,:) = norm_yj;  
233      if norm(b-A*xj)/res(1,1) < tol || size(logres,1)==  
234          max_ciclos  
235          flag = 1;  
236      else
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
236         x0 = xj; %Actualizar la solucion y reiniciar
237         restart = restart+1;
238         if norm_yj < delta
239             flag2 = 1;
240         else
241             flag2 = 0;
242         end
243     end
244
245     %Calcular los vectores armonicos de Ritz
246     W = V(1:n,1:s);
247     Fold = W'*A'*W;
248     gf = R'*R;
249     opts.v0 = ones(size(Fold,1),1);
250     if size(Ev0) ~= 0
251         if s >= size(Ev0,1)
252             Ev1 = zeros(s,1);
253             Ev1(1:size(Ev0,1),:) = Ev0;
254         end
255         if s < size(Ev0,1)
256             Ev0((s+1):size(Ev0,1),:) = [];
257             Ev1 = Ev0;
258         end
259         Ev0 = real(Ev1);
260         opts.v0 = Ev0;
261         Ev1 = [];
262     end
263     E = [];
264     [E2,D2] = eigs(Fold,gf,d,'LM',opts);
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPEROS**

```
264     for pq = 1:d
265         D(pq,1) = abs(D2(pq,pq));
266     end
267     [Y, l] = sort(D,1);
268     for qq = 1:d
269         E(:,qq) = E2(:,l(qq,1));
270     end
271     dy0 = V*E;
272     dy = [];
273
274 %% Si dy0 es complejo
275 if isreal(dy0) == 0
276     ij = 1;
277     while size(dy,2) <= d && ij <= d
278         if isreal(dy0(:,ij)) == 0
279             dy(:,size(dy,2)+1) = real(dy0(:,ij));
280             if ij <= d
281                 dy(:,size(dy,2)+1)= abs(imag(dy0(:,ij)))*sqrt(1));
282             if ij < d
283                 if dy0(:,ij) == conj(dy0(:,ij+1))
284                     ij = ij + 2;
285                 else
286                     ij = ij + 1;
287                 end
288             end
289         end

```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
290           else
291               dy (: , size (dy ,2 )+1) = dy0 (: , ij );
292               ij = ij + 1;
293           end
294       end
295   else
296       dy = dy0 ;
297   end
298
299 else % if size(logres ,1) >= l , la cantidad de vectores
       de errores de
300     % aproximación
301     s = ST;
302     % Método de Arnoldi modificado
303     for j = 1:s
304         if j <= m
305             w (: , j) = A*v (: , j);
306         end
307         if j >= m+1 && j <= m+d
308             w (: , j) = A*dy (: , j-m);
309         end
310         if j >= m+ d + 1 && j <= s -1
311             w (: , j) = A*z (: , l-(j-m-d-1));
312         end
313     for i = 1:j
314         h (i , j) = w (: , j)'*v (: , i);
315         w (: , j) = w (: , j)-h (i , j)*v (: , i);
316     end
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
317      h( j+1 , j ) = norm(w(:, j )) ;  
318      if h( j+1 , j ) == 0  
319          s = j ;  
320          h2 = zeros(s+1,s) ;  
321          for k = 1:s  
322              h2(:,k) = h(:,k) ;  
323          end  
324          h = h2 ;  
325      else  
326          v(:, j+1) = w(:, j )/h( j+1 , j ) ;  
327          end  
328      end  
329      H0 = h ;  
330      hs = h ;  
331  
332      g = zeros(s+1,1) ;  
333      g(1,1) = beta ;  
334  
335      for j = 1:s      % Resolución del problema de  
mínimos cuadrados  
336          P = eye(s+1) ;  
337          sin = h(j+1,j)/(sqrt(h(j+1,j)^2 + h(j,j)^2)) ;  
338          cos = h(j,j)/(sqrt(h(j+1,j)^2 + h(j,j)^2)) ;  
339          P(j,j) = cos ;  
340          P(j+1,j+1) = cos ;  
341          P(j,j+1) = sin ;  
342          P(j+1,j) = -sin ;  
343          h = P*h ;
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
344 g = P*g;
345 end
346 R = zeros(s,s);
347 G = zeros(s,1);
348 V = zeros(n,s);
349 for k = 1:s
    G(k) = g(k);
    V(:,k) = v(:,k);
    for i = 1:s
        R(k,i) = h(k,i);
    end
355 end
356 for k = m+1:m+d
    V(:,k) = dy(:,k-m);
358 end
359 for k = m+d+1:s
    V(:,k) = z(:,l-(k-m-d-1));
361 end
362 yj = R\G;
363 norm_yj = norm(yj);
364 zj = V*yj;
365 xj = x0 + zj;
366
367 Z = z;
368 if l == 1
369     z = aux;
370 else
371     for j = 2:l
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
372          z (: , j -1) = Z (: , j ) ;  
373      end  
374      z (: , l ) = aux ;  
375      end  
376      iter ( restart +1 ,:) = restart +1;  
377      log_de_m ( iter ( size ( iter ,1) ,:) ,1) = m;  
378      log_de_s ( iter ( size ( iter ,1) ,:) ,1) = s;  
379      logres ( size ( logres ,1) +1 ,:) = norm ( b - A * xj ) / res ( 1 ,1) ;  
380      lognormy ( size ( logres ,1) +1 ,:) = norm_yj ;  
381      if norm ( b - A * xj ) / res ( 1 ,1) < tol || size ( logres ,1)  
            == max_ciclos  
            flag = 1;  
383      residuo = norm ( b - A * xj ) / res ( 1 ,1) ;  
384      else  
385          x0 = xj ; % Actualizar la solucion aproximada ,  
            reiniciar  
386          restart = restart + 1;  
387          if norm_yj < delta  
            flag2 = 1;  
388          else  
            flag2 = 0;  
389          end  
390      end  
391      %% Calcular los vectores armonicos de Ritz  
392      W = V ( 1 : n , 1 : s ) ;  
393      Fold = W * A' * W;  
394      gf = R' * R;  
395      opts . v0 = ones ( size ( Fold ,1) ,1) ;
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
398      E = [];  
399      [E2,D2] = eigs(Fold,gf,d,'LM',opts);  
400      for pq = 1:d  
401          D(pq,1) = abs(D2(pq,pq));  
402      end  
403      [Y,I] = sort(D,1);  
404      for qq = 1:d  
405          E(:,qq) = E2(:,I(qq,1));  
406      end  
407      dy0 = V*E;  
408      dy = [];  
409  
410      %% % si dy0 es complejo  
411      if isreal(dy0) == 0  
412          ij = 1;  
413          while size(dy,2) <= d && ij <= d  
414              if isreal(dy0(:,ij)) == 0  
415                  dy(:,size(dy,2)+1) = real(dy0(:,ij));  
416                  if ij <= d  
417                      dy(:,size(dy,2)+1) = abs(imag(dy0  
418                                         (:,ij))*sqrt(1));  
419                  if ij <d  
420                      if dy0(:,ij) == conj(dy0(:,ij  
421                                         +1))  
422                          ij = ij + 2;  
423                  else  
424                      ij = ij + 1;  
425                  end
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

```
424          end
425      end
426      else
427          dy (:, size(dy,2)+1) = dy0 (:, ij);
428          ij = ij +1;
429      end
430  end
431  else
432      dy = dy0;
433  end
434 end % Si hay reinicio
435 end % fin del solver
436 tiempo = cputime - inicio; % Devuelve tiempo de ejecucion
437 save( 'logLGMRESE.mat' , 'xj' , 'tiempo' , 'restart' , sum(
    log_de_s));
438 XX = xj;
```

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

BIBLIOGRAFÍA

- [1] M. D. García León. Estrategias para la resolución de grandes sistemas de ecuaciones lineales. métodos de cuasi-mínimo residuo modificados(tesis doctoral). [En línea]. Disponible: www.dca.iusiani.ulpgc.es/proyecto2015-2017/pdf/tesislola.pdf [Fecha de consulta: Agosto 2017]
- [2] J. Liesen y Z. Strakos, *Krylov Subspace Methods, Principles and Analysis*, 1st ed. Oxford University Press, 2013.
- [3] J. Demmel, *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.
- [4] J. A. Ceballos Cañon, *Alta Precisión Relativa en Problemas de Álgebra Lineal Numérica en Matrices con Estructura (Tesis Doctoral)*. Universidad Carlos III de Madrid, 2013.
- [5] A. Greenbaum, *Iterative Methods for Solving Linear Systems*. Society for Industrial and Applied Mathematics, 1997.
- [6] The SuiteSparse Matrix Collection (formerly known as the University of Florida Sparse Matrix Collection). [En línea]. Disponible: <http://www.cise.ufl.edu/research/sparse/matrices/> [Fecha de consulta: Agosto 2017]
- [7] Control de algoritmos bloque iterativos basados en el Subespacio de Krylov - CABIBESKRY | CONACYT. [En línea]. Disponible: <http://www.conacyt.gov.py/node/1693> [Fecha de consulta: Agosto 2018]
- [8] Y. Saad y M. Schultz, “GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856–869, 1986.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

- [9] A. Bhaya y E. Kaszkurewicz, “A control-theoretic approach to the design of zero finding numerical methods,” *IEEE Transactions on Automatic Control*, vol. 52, no. 6, pp. 1014–1026, 2007.
- [10] R. Cuevas, *Una estrategia inspirada en control para variar el parámetro de reinicio m de GMRES(m)*. Facultad Politécnica, UNA, Diciembre 2010, tesis de Grado de Máster en Informática.
- [11] J. C. Cabral, *Estrategia de control para GMRES(m) aumentado con vectores armónicos de Ritz*. Facultad Politécnica, UNA, Febrero 2014, tesis de Grado de Máster en Ciencias de la Computación.
- [12] I. Ortiz, *Formulación de control para el problema de esparcimiento de Helmholtz*. Facultad de Ingeniería, UNA, 2010, Trabajo Final de Grado de Ingeniería Electrónica.
- [13] E. De Los Santos y J. Gavilán, *Algoritmo Parareal en el estudio de Ondas Electromagnéticas - 2D*. Facultad de Ingeniería, UNA, 2013, Trabajo Final de Grado de Ingeniería Electrónica.
- [14] K. Du, “GMRES with adaptively deflated restarting and its performance on an electromagnetic cavity problem,” *Applied Numerical Mathematics*, vol. 61, no. 9, pp. 977 – 988, 2011.
- [15] H. M. Antia, *Numerical Methods for Scientists and Engineers*, 3rd ed. Hindustan Book Agency, 2012.
- [16] A. Bhaya y E. Kaszkurewicz, *Control Perspectives on Numerical Algorithms and Matrix Problems*. Society for Industrial and Applied Mathematics, 2006.
- [17] G. Larrazábal. Técnicas algebraicas de precondicionamiento para la resolución de sistemas lineales (tesis doctoral). [En línea]. Disponi-

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

ble: <https://upcommons.upc.edu/bitstream/handle/2117/93286/TESIS.pdf?sequence=1&isAllowed=y> [Fecha de consulta: Abril 2018]

- [18] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, y H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods.* Society for Industrial and Applied Mathematics, 1994.
- [19] J. Jaramillo, A. Vidal, y F. Correa, *Métodos directos para la solución de sistemas de ecuaciones lineales simétricos, indefinidos, dispersos de gran dimensión.* Universidad EAFIT - Medellín, 2006.
- [20] G. Strang, *Álgebra lineal y sus aplicaciones.* Paraninfo, 2006.
- [21] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Society of Industrial and Applied Mathematics, 2003.
- [22] J. Brownlee. A gentle introduction to sparse matrices for machine learning. [En línea]. Disponible: <https://machinelearningmastery.com/sparse-matrices-for-machine-learning/> [Fecha de consulta: Agosto 2018]
- [23] ——, *Basics of Linear Algebra for Machine Learning*, 2018.
- [24] K. Ogata, *Ingeniería de control moderna*, 3rd ed. Pearson Educación - México, 2003.
- [25] I. C. F. Ipsen y C. D. Meyer, “The idea behind Krylov methods,” *The American Mathematical Monthly*, vol. 105, no. 10, pp. 889–899, 1998.
- [26] M. Embree, “The tortoise and the hare Restart GMRES,” *SIAM Review*, vol. 45, no. 2, pp. 259–266, 2003. [En línea]. Disponible: <https://doi.org/10.1137/S003614450139961>

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

- [27] V. Simoncini y D. Szyld, “Recent computational developments in Krylov subspace methods for linear systems,” *Numerical Linear Algebra with Applications*, vol. 14, no. 1, pp. 1–59, 2007.
- [28] W. Joubert, “On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems,” *Numerical Linear Algebra with Applications*, vol. 1, no. 5, pp. 427–447, 1994.
- [29] A. H. Baker, E. R. Jessup, y T. V. Kolev, “A simple strategy for varying the restart parameter in GMRES(m),” *Journal of Computational and Applied Mathematics*, vol. 230, pp. 751–761, 2009.
- [30] R. Cuevas, C. Schaefer, y A. Bhaya, “A proportional-derivative control strategy for restarting the GMRES(m) algorithm,” *Journal of Computational and Applied Mathematics*, vol. 337, pp. 209–224, 2018.
- [31] A. Chapman y Y. Saad, “Deflated and augmented Krylov subspace techniques,” *Numerical Linear Algebra with Applications*, vol. 4, no. 1, pp. 43–66, 1997.
- [32] A. Gaul, M. Gutknecht, J. Liesen, y R. Nabben, “A framework for deflated and augmented Krylov subspace methods,” *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 2, pp. 495–518, 2013.
- [33] A. Baker, E. Jessup, y T. Manteuffel, “A technique for accelerating the convergence of Restarted GMRES,” *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 4, pp. 962–984, 2005.
- [34] R. Morgan, “A restarted GMRES method augmented with eigenvectors,” *SIAM Journal on Matrix Analysis and Applications*, vol. 16, no. 4, pp. 1154–1171, 1995.

**DESARROLLO DE UN SISTEMA DE CONTROL PARA EL ALGORITMO GMRES(m) DE
RESOLUCIÓN DE SISTEMAS LINEALES GRANDES DISPERSOS**

- [35] D. B. Davidson, *Computational Electromagnetics for RF and Microwave Engineering*, 2nd ed. Cambridge University Press, 2010.
- [36] F. M. Kahnert, “Numerical methods in electromagnetic scattering theory,” *Journal of Quantitative Spectroscopy and Radiative Transfer*, vol. 79-80, pp. 775–824, 2003.
- [37] M. Spiegel, *Análisis Vectorial, Serie Schaum*. McGraw-Hill Interamericana, 1995.
- [38] A. Kirsch y F. Hettlich, *The Mathematical Theory of Time-Harmonic Maxwell's Equations*. Springer, 2015.
- [39] W. Hayt y J. Buck, *Teoría electromagnética*. McGraw-Hill, 2006.
- [40] R. L. Burden y J. D. Faires, *Numerical Analysis*, 9th ed. Cengage Learning, 2011.
- [41] D. R. Durran, *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag New York, 1999.
- [42] Y. Wang, K. Du, y W. Sun, “Preconditioning iterative algorithm for the electromagnetic scattering from a large cavity,” *Numerical Linear Algebra with Applications*, vol. 16, no. 5, pp. 345–363, 2009.
- [43] Subset of eigenvalues and eigenvectors - MATLAB eigs - Mathworks América Latina. [En línea]. Disponible: <https://la.mathworks.com/help/matlab/ref/eigs.html> [Fecha de consulta: Agosto 2018]