

Universidad Autónoma de Querétaro

Facultad de Ingeniería

Maestría en Ciencias

**“Método de fuentes discretas en
problemas de dispersión de
ondas”**

Tesis

que como parte de los requisitos para obtener el título de:
Maestro en Ciencias

es presentada por:

Jesús Felipe Centeno Mora

Dirigido por:

Dra. Kira Khmelnytskaya Gerasimova

SINODALES:

Dra. Kira Khmelnytskaya Gerasimova

PRESIDENTA

Dr. Ulises Velasco García

SECRETARIO

Dr. Roberto Augusto Gómez Loenzo

VOCAL

Dr. Víctor Antonio Aguilar Arteaga

SUPLENTE

M. C. José Enrique Crespo Baltar

SUPLENTE

Santiago de Querétaro, Qro. Junio de 2025

La presente obra está bajo la licencia:
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>



CC BY-NC-ND 4.0 DEED

Atribución-NoComercial-SinDerivadas 4.0 Internacional

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciatario no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:

 **Atribución** — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciatario.

 **NoComercial** — Usted no puede hacer uso del material con [propósitos comerciales](#).

 **SinDerivadas** — Si [remezcla, transforma o crea a partir](#) del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni [medidas tecnológicas](#) que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

Avisos:

No tiene que cumplir con la licencia para elementos del material en el dominio público o cuando su uso esté permitido por una [excepción o limitación](#) aplicable.

No se dan garantías. La licencia podría no darle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como [publicidad, privacidad, o derechos morales](#) pueden limitar la forma en que utilice el material.

*Al Creador de Todo;
a la memoria de mi padre,
a mi madre y a mis hermanos.*

AGRADECIMIENTOS

Expreso mi sincera gratitud a la Dra. Kira Khmelnytskaya Gerasimova, por haberme distinguido al aceptar dirigir esta tesis y por su generosa paciencia. Agradezco también al Doctor Ulises Velasco García, al Doctor Roberto Augusto Gómez Loenzo, al Doctor Víctor Antonio Aguilar Arteaga y al Maestro José Enrique Crespo Baltar, por su competente y amable labor como sinodales de la presente investigación.

Agradezco el apoyo otorgado por el entonces Consejo Nacional de Ciencia y Tecnología (CONACYT), en la forma de beca, el cual me fue muy útil en la realización de mis estudios de Maestría.

Índice general

DEDICATORIA	III
AGRADECIMIENTOS	IV
RESUMEN	XII
ABSTRACT	XIII
ESTRUCTURA DE LA TESIS	XIV
I. CONCEPTOS PRELIMINARES	1
1.1. Ecuaciones diferenciales parciales con condiciones en la frontera	1
1.2. Análisis Funcional	3
1.3. Conceptos adicionales	7
II. INTRODUCCIÓN	8
2.1. Antecedentes	10
2.2. Justificación	12
2.3. Hipótesis	13
2.4. Objetivo	13
III. EL MÉTODO DE FUENTES DISCRETAS (DSM)	14
3.1. La ecuación de Helmholtz	14
3.2. Problemas internos y problemas externos	16
3.3. Dispersión de ondas	17
3.4. Problemas Dirichlet en dominios acotados	20
3.5. Problemas Dirichlet en dominios no acotados	21
3.6. Descripción del problema	22
3.7. Fundamentación teórica	24
3.8. Metodología	27
IV. EJEMPLOS DE PRUEBA	33
4.1. Acerca del parámetro <i>factor de distancia (f. d.)</i>	33
4.2. Acerca de la generación de los <i>puntos de prueba</i>	34
4.3. Problemas internos	36

4.3.1. Ejemplo 1	36
4.3.2. Ejemplo 2	40
4.4. Problemas externos	49
4.4.1. Ejemplo 3	49
4.4.2. Ejemplo 4	53
V. CONCLUSIONES	63
Bibliografía	65
Apéndice A. — CÓDIGOS DE LAS FUNCIONES GENERALES	68
Apéndice B. — CÓDIGOS DE LOS SCRIPTS Y FUNCIONES PARA LOS EJEMPLOS	73
Apéndice C. — NOTACIÓN	104

Índice de figuras

1.	Esquema para los problemas del tipo interno.	16
2.	Esquema para los problemas del tipo externo.	17
3.	Esquema del procedimiento de ubicación de las fuentes discretas, para un problema interno.	29
4.	Esquema de la ubicación de la frontera real y de la frontera virtual, usando para esta última un factor de distancia $f. d. = 0.8$	34
5.	Esquema de la ubicación de los puntos de prueba. En este caso se considera un problema del tipo externo.	35
6.	Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 1. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	36
7.	Gráficas de la superficie de la solución exacta y de los valores de la solución aproximada en los puntos de prueba para el ejemplo 1. En la gráfica de la izquierda se utilizaron $N = 50$ términos; en la gráfica de la derecha se utilizaron $N = 150$ términos. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	37
8.	Gráfica de la superficie del error absoluto para el ejemplo 1. $N = 150$ términos. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	38
9.	Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 1. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	39

10. Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 2, inciso a. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.	41
11. Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 2, inciso a. En la gráfica de la izquierda se utilizaron N = 40 términos; en la gráfica de la derecha se utilizaron N = 120 términos. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.	41
12. Gráfica de la superficie del error absoluto para el ejemplo 2, inciso a. N = 120 términos. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.	42
13. Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 2, inciso a. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.	43
14. Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 2, inciso b. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	45
15. Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 2, inciso b. En la gráfica de la izquierda se utilizaron N = 40 términos; en la gráfica de la derecha se utilizaron N = 120 términos. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	46
16. Gráfica de la superficie del error absoluto para el ejemplo 2, inciso b. N = 120 términos. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	47

17. Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 2, inciso b. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.	48
18. Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 3. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	50
19. Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 3. En la gráfica de la izquierda se utilizaron N = 10 términos; en la gráfica de la derecha se utilizaron N = 30 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	51
20. Gráfica de la superficie del error absoluto para el ejemplo 3. N = 30 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	52
21. Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 3. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	53
22. Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 4, inciso a. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	54
23. Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 4, inciso a. En la gráfica de la izquierda se utilizaron N = 12 términos; en la gráfica de la derecha se utilizaron N = 28 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	55

24. Gráfica de la superficie del error absoluto para el ejemplo 4, inciso a. N = 28 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	56
25. Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 4, inciso a. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	57
26. Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 4, inciso b. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	59
27. Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 4, inciso b. En la gráfica de la izquierda se utilizaron N = 8 términos; en la gráfica de la derecha se utilizaron N = 24 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	59
28. Gráfica de la superficie del error absoluto para el ejemplo 4, inciso b. N = 24 términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	60
29. Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 4, inciso b. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.	61

Índice de cuadros

1.	Soluciones fundamentales para algunas ecuaciones diferenciales parciales comunes	31
2.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 1. Las distancias mínimas entre la frontera real y la frontera real son: 0.1 para f. d.=1.1; 0.5 para f. d.=1.5 y 0.9 para f. d.=1.9.	40
3.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 2, inciso a. Las distancias mínimas entre la frontera real y la frontera real son: 0.19 para f. d.=3.3523; 1.19 para f. d.=4.3523 y 2.19 para f. d.=5.3523.	44
4.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 2, inciso b. Las distancias mínimas entre la frontera real y la frontera real son: 0.1 para f. d.=1.1; 0.5 para f. d.=1.5 y 0.9 para f. d.=1.9.	49
5.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 3. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.	54
6.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 4, inciso a. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.	58
7.	Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 4, inciso b. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.	62

RESUMEN

En esta investigación se hace una exposición del Método de Fuentes Discretas (DSM, por sus siglas en inglés), el cual es un método numérico utilizado para resolver problemas de dispersión de ondas con condiciones en la frontera. La exposición incluye la justificación teórica del método en el caso particular del operador de Helmholtz, así como ejemplos de prueba para tal operador con condiciones en la frontera del tipo Dirichlet sobre dominios bidimensionales con variantes en cuanto a la configuración, como lo son la aplicación del método sobre dominios elípticos y dominios rectangulares, y la elección de diferentes formas de fronteras virtuales. La implementación del DSM utiliza la técnica de colocación de los datos de frontera. Se muestran datos, conclusiones y sugerencias, tanto en el caso de problemas internos, como en el caso de problemas externos.

(**Palabras clave:** Método de Fuentes Discretas, DSM, dispersión, ondas, obstáculo *sound-soft*, problema interno, problema externo)

ABSTRACT

This paper presents the Discrete Source Method (DSM), a numerical method used to solve wave scattering problems with boundary conditions. The dissertation includes the theoretical justification of the method in the particular case of the Helmholtz operator, as well as test examples for such operator with Dirichlet boundary conditions on two-dimensional domains with variations concerning configuration, such as the application of the method to elliptical and rectangular domains, and the choice of different virtual boundary shapes. The DSM implementation uses the boundary data collocation technique. Data, conclusions, and suggestions are presented for both interior and exterior problems.

(**Keywords:** Discrete Source Method, DSM, scattering, waves, sound-soft obstacle, interior problem, exterior problem)

ESTRUCTURA DE LA TESIS

La presente tesis está conformada de la manera siguiente:

en el capítulo I: Conceptos preliminares, se presentan algunos conceptos y resultados útiles en la lectura de los capítulos siguientes, a manera de un breve glosario. En el capítulo II: Introducción, se presentan los antecedentes del Método de Fuentes Discretas, así como la Justificación, Hipótesis y Objetivo de esta investigación. En el capítulo III: El Método de Fuentes Discretas (DSM), se comienza tratando la ecuación de Helmholtz, se continúa con consideraciones de carácter físico en las que se encuentra enmarcado dicho método, y concluye con la Fundamentación Teórica y la Metodología empleada en la implementación numérica del mismo. En el capítulo IV: Ejemplos de prueba, se presentan los experimentos numéricos y los resultados obtenidos. Finalmente, en el capítulo V se muestran las conclusiones de la presente investigación.

En los apéndices A y B el lector podrá encontrar los códigos en Matlab, que se usaron en el capítulo IV.

Capítulo I

CONCEPTOS PRELIMINARES

El propósito de este capítulo es presentar a manera de glosario, y de forma no exhaustiva, conceptos y teoremas generales que serán utilizados a lo largo del presente trabajo. Debido al carácter general de los teoremas incluidos en este capítulo, se presentan únicamente los enunciados, indicándose la referencia bibliográfica donde pueden verificarse las demostraciones respectivas.

1.1. *Ecuaciones diferenciales parciales con condiciones en la frontera*

Se puede representar un operador diferencial de orden m sobre un conjunto abierto $\Omega \subset \mathbb{R}^n$ de la manera siguiente

$$\mathcal{L} = \mathcal{L}(x, D) = \sum_{|\alpha| \leq m} a_\alpha(x) D^\alpha, \quad x \in \mathbb{R}^n.$$

Para las definiciones que se presentan a continuación consúltese: en el caso de los conceptos de símbolo de un operador diferencial parcial, de parte principal de un operador y de operador diferencial parcial elíptico, el libro [Renardy & Rogers, 2004, pp. 37 - 42]; para la definición de función analítica, puede consultarse [Vladimirov, 1971, p. 59].

Definición 1 *El símbolo de un operador diferencial parcial \mathcal{L} , denotado por $\mathcal{L}(x, i\xi)$, se define de la manera siguiente:*

$$\mathcal{L}(x, i\xi) = \sum_{|\alpha| \leq m} a_\alpha(x)(i\xi)^\alpha, \quad i^2 = -1; x, \xi \in \mathbb{R}^n.$$

La parte principal del operador \mathcal{L} , denotado por $\mathcal{L}_m(x, i\xi)$, también conocida como ‘parte principal del símbolo del operador’ queda definida por:

$$\mathcal{L}_m(x, i\xi) = \sum_{|\alpha|=m} a_\alpha(x)(i\xi)^\alpha, \quad i^2 = -1; x, \xi \in \mathbb{R}^n.$$

Definición 2 Suponga que X y Y son espacios vectoriales complejos, y sea $A : X \rightarrow Y$ un operador lineal. El **kernel** (o espacio nulo) del operador A es el subespacio de X definido por

$$\ker A = \{u \in X : Au = 0\}.$$

Esta definición se aplica también a los operadores diferenciales lineales.

Definición 3 Una función $f(x)$, $x = (x_1, \dots, x_n)$, es llamada **función analítica** en el punto x_0 , si en cierta vecindad de ese punto puede ser representada en la forma de una serie de potencias uniformemente convergente

$$f(x) = \sum_{|\alpha| \geq 0} c_\alpha (x - x_0)^\alpha = \sum_{|\alpha| \geq 0} \frac{D^\alpha f(x_0)}{\alpha!} (x - x_0)^\alpha.$$

Si una función $f(x)$ es analítica en todo punto de alguna región G , se dice que es una función analítica en G . El conjunto de todas las funciones analíticas en un cierto abierto G se denota por $C^\omega(G)$.

Definición 4 Sea $\mathcal{L} = \sum_{|\alpha| \leq m} a_\alpha(x) D^\alpha$ un operador diferencial parcial de orden m en $\Omega \subset \mathbb{R}^n$. Se dice que \mathcal{L} es un **operador diferencial parcial elíptico** en $x \in \Omega$ si NO hay características reales en x , o de forma equivalente, si se cumple que

$$\mathcal{L}_m(x, i\xi) \neq 0 \quad \forall \xi \neq 0 \in \mathbb{R}^n.$$

Se dice que el operador es elíptico en Ω , si es elíptico en todo punto de Ω .

Definición 5 Se dice que la frontera de un dominio Ω es **frontera Lyapunov** si en cada punto $\mathbf{x} \in \partial\Omega$ el vector normal \mathbf{n} a la superficie existe, y además existen constantes positivas L y α , tales que el ángulo $\theta(\mathbf{x}, \mathbf{y})$ entre los vectores normales en \mathbf{x} y en \mathbf{y} cumple que $\theta(\mathbf{x}, \mathbf{y}) \leq L |\mathbf{x} - \mathbf{y}|^\alpha$, $\forall \mathbf{x}, \mathbf{y} \in \partial\Omega$.

Definición 6 Se dice que un conjunto abierto $\Omega \subset \mathbb{R}^2$ con frontera acotada es un **dominio Lipschitz** (o con frontera Lipschitz) si (1) existe una familia finita de conjuntos abiertos $\{W_j\}_{j=1, \dots, J}$ que cubren a $\partial\Omega$ (i. e. $\partial\Omega \subset \cup_{j=1, \dots, J} W_j$), (2) existe una familia de hipografos Lipschitz rotados $\{\Omega_j\}_{j=1, \dots, J}$ (i. e. $\Omega_j = \{(x'_1, x'_2) : x'_2 < f_j(x'_1)\}$,

donde (x'_1, x'_2) es un sistema rotado de coordenadas cartesianas en \mathbb{R}^2 y f_j es una función Lipschitz), y (3) $W_j \cap \Omega = W_j \cap \Omega_j$. Intuitivamente, lo anterior significa que para cada $\mathbf{x} \in \partial\Omega$, existe una vecindad en la cual la frontera se puede representar como la gráfica de una función Lipschitz, y Ω yace solamente en un lado de $\partial\Omega$. Los dominios con frontera suave, así como los polígonos son ejemplos de dominios Lipschitz. Se dice que el dominio Ω es **dominio de clase C^m** , $m = 0, 1, \dots, \infty$ si las funciones f_j son de clase C^m . [Moiola, 2020, p. 16].

1.2. Análisis Funcional

Definición 7 Sea X un espacio métrico con métrica d . Un subconjunto A de X es llamado **conjunto acotado** si existe un número M tal que

$$d(a_1, a_2) \leq M$$

para todo par a_1, a_2 de puntos en A . [Munkres, 2000, p. 121].

Definición 8 Un subconjunto M de un espacio métrico X es llamado **subconjunto denso** en X si

$$\overline{M} = X.$$

[Kreyszig, 1978, p. 21].

Definición 9 Sean X y Y espacios normados y $T : D(T) \rightarrow Y$ un operador lineal, donde $D(T) \subset X$. El operador T se llama **operador acotado** si existe un número real c tal que $\forall x \in D(T)$ se tiene que $\|Tx\| \leq c\|x\|$.

Definición 10 Si $\phi : X \rightarrow \mathbb{R}$, donde X es un espacio topológico, se define el **soporte de la función ϕ** como la cerradura del conjunto $\phi^{-1}(\mathbb{R} \setminus \{0\})$. Por lo tanto si x no pertenece al soporte de ϕ , existe una vecindad de x en la cual ϕ vale cero. [Munkres, 2000, p. 225].

Definición 11 Si Ω es un dominio abierto en \mathbb{R}^n , entonces el espacio $\mathbf{C}^\ell(\Omega)$, donde ℓ es un entero no negativo, es el espacio que contiene todas las funciones u , las cuales junto con todas sus derivadas parciales $D^\alpha u$ de órdenes $|\alpha| \leq \ell$, son continuas en Ω ; además se define

$C^\infty(\Omega) = \bigcap_{\ell \in \mathbb{N}} C^\ell(\Omega)$. El espacio $\mathbf{C}^\ell(\overline{\Omega})$ consiste de todas las funciones $u \in C^\ell(\Omega)$ para las cuales $D^\alpha u$ es uniformemente continua y acotada en Ω para todo $|\alpha| \leq \ell$. De hecho, $C^\ell(\overline{\Omega})$ es un espacio Banach. Además, $\mathbf{C}_0^\ell(\mathbb{R}^n)$ es el conjunto de funciones u las cuales, junto con todas sus derivadas parciales $D^\alpha u$ de órdenes $|\alpha| \leq \ell$, son continuas en \mathbb{R}^n y satisfacen que $\lim_{|x| \rightarrow \infty} D^\alpha u(x) = 0$, para cada $|\alpha| \leq \ell$; $C_0^\ell(\mathbb{R}^n)$ es también un espacio Banach. Los elementos de $(C^\ell(\overline{\Omega}))'$ definen funcionales lineales acotados sobre $C_0^\ell(\mathbb{R}^n)$.

Proposición 1 Si \mathcal{L} es un operador elíptico con coeficientes constantes, de orden m en \mathbb{R}^n y $e = e(x)$ es una solución fundamental de \mathcal{L} , entonces $D^\alpha e \in L_{loc}^1(\mathbb{R}^n)$ para cada $|\alpha| < m$. Para la justificación de esta proposición se remite al lector a [Weinstock, 1973, pp. 513-514].

Definición 12 Espacios L^p . Suponga que U es un subconjunto abierto de \mathbb{R}^n , y $1 \leq p \leq \infty$. Si $f : U \rightarrow \mathbb{R}$ es medible, se define

$$\|f\|_{L^p(u)} := \begin{cases} \left(\int_U |f|^p dx \right)^{1/p} & \text{si } 1 \leq p < \infty \\ \text{ess sup}_U |f| & \text{si } p = \infty. \end{cases}$$

Se define $L^p(U)$ como el espacio lineal de todas las funciones medibles $f : U \rightarrow \mathbb{R}$ para las cuales $\|f\|_{L^p(u)} < \infty$. L^p es un espacio Banach. Además se define $L_{loc}^p(U) = \{u : U \rightarrow \mathbb{R} : u|_V \in L^p(V) \text{ para todo } V \subset U, V \text{ compacto}\}$.

Definición 13 Espacios Sobolev. Fije $1 \leq p \leq \infty$ y sea k un entero no negativo. Se definen ciertos espacios funcionales, cuyos elementos poseen derivadas débiles de varios órdenes en varios espacios L^p . El espacio Sobolev

$$W^{k,p}(U)$$

consiste en todas las funciones localmente sumables $u : U \rightarrow \mathbb{R}$, tales que para cada multiíndice α con $|\alpha| \leq k$, $D^\alpha u$ exista en el sentido débil y pertenezca a $L^p(U)$. Si $p = 2$, usualmente se escribe

$$H^k(U) = W^{k,2}(U), \quad k = 0, 1, \dots$$

Se usa la letra H dado que $H^k(U)$ es un espacio Hilbert.

Definición 14 Denote por $D(\Omega)$ al espacio de **funciones test**, las cuales son funciones en C^∞ , de valor complejo, definidas en Ω y con soporte compacto en Ω .

Definición 15 El espacio $H_0^1(\Omega)$ es el subespacio de $H^1(\Omega)$ de los elementos que pueden ser aproximados en la norma de $H^1(\Omega)$ mediante una sucesión de elementos en $D(\Omega)$.

Definición 16 Se dice que una función $u \in H_{loc}^1(\bar{\Omega})$ si la restricción de u a cualquier subconjunto abierto acotado D en Ω pertenece a $H^1(D)$. En el caso de que Ω sea un conjunto acotado, entonces $H_{loc}^1(\bar{\Omega}) = H^1(\Omega)$.

Definición 17 Sea $S^1 = \{\mathbf{x} \in \mathbb{R}^2, \|\mathbf{x}\| = 1\}$ la circunferencia unitaria. Para una función v definida en S^1 se escribe $v(\theta)$ para su valor en coordenadas polares. Se dice que $v \in L^2(S^1)$ si $\|v\|_{L^2(S^1)}^2 := \int_0^{2\pi} |v|^2 d\theta < \infty$ y $v \in H^1(S^1)$ si $\|v\|_{H^1(S^1)}^2 := \int_0^{2\pi} (|v|^2 + |v'|^2) d\theta < \infty$, donde v' es la derivada en la coordenada angular. La expansión de v en armónicos circulares es $v(\theta) = \sum_{\ell \in \mathbb{Z}} \hat{v}_\ell e^{i\ell\theta}$, para una sucesión de coeficientes $\hat{v}_\ell \in \mathbb{C}$. Se pueden calcular las normas usando esta expansión:

$$\|v\|_{L^2(S^1)}^2 = \int_0^{2\pi} |v|^2 d\theta = 2\pi \sum_{\ell \in \mathbb{Z}} |\hat{v}_\ell|^2; \quad \|v\|_{H^1(S^1)}^2 = \int_0^{2\pi} (|v|^2 + |v'|^2) d\theta = 2\pi \sum_{\ell \in \mathbb{Z}} |\hat{v}_\ell|^2 (1 + \ell^2).$$

Por lo tanto, una función definida en la circunferencia está en $L^2(S^1)$ si la sucesión de coeficientes de Fourier es un elemento en el espacio de sucesiones $l^2(\mathbb{Z}) = \{(a_\ell), \ell \in \mathbb{Z}, \|a_\ell\|_{l^2}^2 := \sum_{\ell \in \mathbb{Z}} |a_\ell|^2 < \infty\}$, y en $H^1(S^1)$ si sus coeficientes de Fourier ponderados con $(1 + \ell^2)^{1/2}$ están en $l^2(\mathbb{Z})$. Lo anterior nos sugiere formas de definir **espacios Sobolev con otras regularidades**:

$$\|v\|_{H^s(S^1)}^2 := 2\pi \sum_{\ell \in \mathbb{Z}} |\hat{v}_\ell|^2 (1 + \ell^2)^s; \quad H^s(S^1) := \{v(\theta) = \sum_{\ell \in \mathbb{Z}} \hat{v}_\ell e^{i\ell\theta} : \|v\|_{H^s(S^1)} < \infty\}, \quad \forall s \in \mathbb{R}.$$

Para $s = 0$ y $s = 1$ se tiene $H^0(S^1) = L^2(S^1)$ y $H^1(S^1)$ como fueron definidos arriba. Mientras mayor sea el valor de s , mayor será la suavidad de los elementos en $H^s(S^1)$. Si $s > 1/2$, los elementos de $H^s(S^1)$ son funciones continuas, si $s \geq 0$ serán simplemente clases de equivalencia $L^2(S^1)$; para $s < 0$ los elementos sólo pueden ser entendidos como distribuciones. [Moiola, 2020, p. 17]

Definición 18 Dado un dominio Lipschitz acotado Ω , si existe un mapeo bi-Lipschitz $\Phi : \overline{B_1} = \{|\mathbf{x}| \leq 1\} \rightarrow \overline{\Omega}$ que mapee S^1 en $\partial\Omega$, se puede definir el espacio $H^s(\partial\Omega)$ como el espacio de funciones v definidas en $\partial\Omega$ cuyo pullback (operador de composición) $v^*(\mathbf{x}) = v(\Phi(\mathbf{x}))$ es un elemento de $H^s(S^1)$. Para $-1 \leq s \leq 1$ esto da un espacio bien definido, independientemente de la elección de Φ . En el contexto de la presente investigación, serán necesarias sólo los espacios $H^s(\partial\Omega)$ para $s = \pm 1/2$: $H^{1/2}(\partial\Omega) \subset L^2(\partial\Omega) \subset H^{-1/2}(\partial\Omega)$.

Teorema 19 (Aproximación global por funciones suaves). Suponga que U es acotado, suponga también que $u \in W^{k,p}(U)$ para algún $1 \leq p < \infty$. Entonces existen funciones $u_m \in C^\infty(U) \cap W^{k,p}(U)$ tales que

$$u_m \rightarrow u \text{ en } W^{k,p}(U)$$

[Evans, 1998].

Al plantearse la posibilidad de asignar valores de frontera a lo largo de ∂U a una función $u \in W^{1,p}(U)$, asumiendo que ∂U es clase C^1 . Si $u \in C(\overline{U})$, entonces claramente u tiene valores en ∂U en el sentido usual. El problema es que una función típica en $W^{1,p}(U)$ no es en general continua e, incluso peor, está sólo definida en casi todas partes en U . Puesto que ∂U tiene medida Lebesgue cero n-dimensional, no hay significado directo que se le pueda dar a la expresión ' u restringida a ∂U '. El concepto de operador traza soluciona este problema. Se considerará $1 \leq p < \infty$.

Teorema 20 Suponga que U es acotado y que ∂U es C^1 . Entonces existe un operador lineal acotado

$$T : W^{1,p}(U) \rightarrow L^p(\partial U)$$

tal que (i) $Tu = u|_{\partial U}$ si $u \in W^{1,p}(U) \cap C(\overline{U})$, y (ii) $\|Tu\|_{L_p(\partial U)} \leq C \|u\|_{W^{1,p}(U)}$, para cada $W^{1,p}(U)$, con C dependiendo sólo de p y U .

Definición 21 Se llamará a Tu la **traza** de u en ∂U .

Definición 22 Sea H un espacio Hilbert. Un sistema de elementos $\{\psi_i\}_{i=1}^\infty$ es llamado un **sistema completo** en H si el span lineal de $\{\psi_i\}_{i=1}^\infty$ o el conjunto de todas las

combinaciones lineales finitas de $\{\psi_i\}_{i=1}^{\infty}$

$$Sp\{\psi_1, \psi_2, \dots\} = \{u = \sum_{i=1}^N \alpha_i \psi_i : \alpha_i \in \mathbb{C}, N = 1, 2, \dots\}$$

es un conjunto denso en H , i. e. $\overline{Sp\{\psi_1, \psi_2, \dots\}} = H$. [Doicu et al., 2000, p. 7].

1.3. Conceptos adicionales

Definición 23 Las funciones de **Hankel** del primer y del segundo tipo están relacionadas con las funciones de Bessel, a través de las siguientes expresiones, respectivamente:

$$\begin{aligned} H_{\nu}^{(1)}(z) &= J_{\nu}(z) + iY_{\nu}(z), \\ H_{\nu}^{(2)}(z) &= J_{\nu}(z) - iY_{\nu}(z). \end{aligned}$$

donde $i^2 = -1$, J_{ν} representa la función de Bessel del primer tipo de orden ν , y Y_{ν} representa la función de Bessel del segundo tipo de orden ν . [Polyanin & Nazaikinskii, 2016, p. 1525].

Definición 24 En la literatura sobre la dispersión de ondas, se dice que un obstáculo es **sound-soft** cuando

$$u^{tot}|_{\Gamma} = 0,$$

o viendo a u^{scat} como incógnita

$$u^{scat}|_{\Gamma} = -u^{inc}|_{\Gamma}.$$

Obstáculos **sound-hard** son aquéllos para los cuales

$$\partial_{\nu} u^{tot}|_{\Gamma} = 0,$$

o equivalentemente

$$\partial_{\nu} u^{scat}|_{\Gamma} = -\partial_{\nu} u^{inc}|_{\Gamma}.$$

Aquí ∂_{ν} es la derivada normal, donde el vector normal apunta siempre hacia afuera [Sayas, 2006, p. 5]. En esta investigación se utiliza esta terminología en inglés al no disponerse de una traducción adecuada al español.

Capítulo II

INTRODUCCIÓN

El Método de Fuentes Discretas, es un método que permite resolver de forma numérica problemas de dispersión de ondas, con condiciones en la frontera.

Dicho método tiene como principal antecedente el método de soluciones fundamentales (MFS, por sus siglas en inglés). El MFS es usado para la resolución de ecuaciones diferenciales parciales elípticas con valores en la frontera. Una ecuación diferencial elíptica es una ecuación diferencial parcial de segundo orden que puede escribirse en la forma canónica

$$u_{\xi\xi} + u_{\eta\eta} + \ell_1[u] = G(\xi, \eta) \quad (1)$$

donde u es la función que se está modelando, ξ y η son las variables independientes, ℓ_1 es un operador diferencial lineal de primer orden y G es una función [Pinchover & Rubinstein, 2005, p. 66].

Definición 25 Si \mathcal{L} denota un operador diferencial parcial en $\Omega \subset \mathbb{R}^n$ y se tiene la ecuación $\mathcal{L}u = 0$ en Ω , se dice que una **solución fundamental** de dicha ecuación diferencial es una función $\phi(x, s)$ [esta notación se usará de aquí en adelante] tal que:

$$\mathcal{L}_x \phi(\cdot, s) = \delta_s, \quad s \in \mathbb{R}^n \quad (2)$$

donde $\mathcal{L}_x \phi$ indica que la función ϕ se deriva con respecto a x y δ_s es la medida de Dirac con masa unitaria en s .

Nótese que la función ϕ está definida en todas partes, excepto cuando $x = s$, donde es singular. Por lo tanto se dice que s es la singularidad de la solución fundamental. La idea central del MFS, así como del Método de Fuentes Discretas, es expresar la solución de una ecuación diferencial elíptica como una combinación lineal de soluciones fundamentales del operador involucrado. Es decir, se propone que

$$u(\mathbf{x}, \mathbf{c}) = \sum_{j=1}^N c_j \phi(\mathbf{x} - \mathbf{s}_j) \quad (3)$$

donde las s_j son las ubicaciones de las fuentes discretas, y los coeficientes c_j se deben determinar a partir de las condiciones de frontera.

En esta investigación se aplica el método de fuentes discretas a distintas configuraciones de problemas de dispersión, usando diferentes familias de soluciones fundamentales en cada caso, para comparar la eficiencia del método en estos planteamientos. Se plantea implementar el método y elaborar *ejemplos de prueba*, los cuales tienen las siguientes características: se trata de problemas de dispersión en medios homogéneos y libres de fuentes, sobre dominios en \mathbb{R}^2 , en los cuales se cuenta con solución exacta. Se analiza el comportamiento del error en dichas configuraciones con respecto a la familias de soluciones fundamentales utilizada, la cantidad de términos que se usa en la aproximación, la distribución de las fuentes discretas, así como de las condiciones de frontera requeridas y se comparan los niveles de error absoluto entre los casos. La hipótesis de esta investigación es:

El método de fuentes discretas presenta diferencias significativas en su eficiencia, medida a través del error absoluto, siendo uno de los factores por considerar: la configuración del problema de dispersión (dominios acotados o dominios no acotados). La familia de soluciones fundamentales que se utilice en cada caso también es un factor significativo sobre la eficiencia. Ambos factores serán considerados como variables categóricas.

En los problemas de dispersión, cuando el tamaño del elemento dispersor es del mismo orden de magnitud que la longitud de onda, λ , de la radiación incidente, el Método de Fuentes Discretas tiene ventajas sobre los métodos de malla (método de elemento finito, método de diferencias finitas, etc.), ya que requiere menor tiempo de cómputo y su implementación es más sencilla.

El uso del método de fuentes discretas no está muy difundido, a pesar de sus ventajas con respecto a los métodos de malla. Este trabajo pretende contribuir a la difusión del conocimiento de esta herramienta.

2.1. Antecedentes

El Método de Fuentes Discretas (DSM, por las siglas en inglés de *Discrete Sources Method*), es un método que permite resolver de forma numérica problemas de dispersión de ondas, con condiciones en la frontera.

El DSM tiene como principal antecedente el método de soluciones fundamentales (MFS, por sus siglas en inglés), el cual fue presentado en [Kupradze & Aleksidze, 1963] e inicialmente se conoció como Método de series generalizadas de Fourier . En esta versión inicial del método, para el problema de Dirichlet siguiente

$$\begin{cases} \Delta u = 0, & \text{en } \Omega, \\ u = f, & \text{sobre } \partial\Omega, \end{cases} \quad (4)$$

donde:

- Δ representa el operador de Laplace,
- u es la función incógnita,
- Ω designa el dominio en el que se busca la solución,
- f es la función que indica el comportamiento de u en la frontera, y
- $\partial\Omega$ indica la frontera del dominio,

en un dominio acotado $\Omega \subset \mathbb{R}^n$, $n = 2, 3$, sin agujeros, las fuentes puntuales (denominadas fuentes discretas en el DSM) eran colocadas sobre una *frontera virtual*, es decir, sobre una frontera prescrita $\partial\Omega'$ de un dominio Ω' , satisfaciendo que $\overline{\Omega} \subset \Omega'$. Kupradze también aplicó el MFS en problemas de elasticidad lineal, así como en la ecuación de calor [Kupradze & Aleksidze, 1963].

“El método de soluciones fundamentales es una técnica para la solución numérica de ciertos problemas elípticos con valores en la frontera que pertenece a la clase de métodos conocida generalmente como métodos de frontera.” [Fairweather & Karageorghis, 1998].

“Los primeros usos del MFS se enfocaron en la solución de problemas de potencial lineal en dos y tres variables espaciales. Desde entonces ha sido aplicado a una variedad

de problemas más complicados tales como problemas de potencial plano involucrando condiciones en la frontera no lineales tipo radiación, problemas de frontera libre, problemas biarmónicos, problemas elastostáticos y problemas de dispersión de ondas.” [Fairweather & Karageorghis, 1998].

Puede considerarse que el momento en el cual el MFS comenzó a utilizarse en problemas de dispersión de ondas (principalmente ondas electromagnéticas), marca el inicio del método de fuentes discretas (DSM) como tal. Las primeras aplicaciones de este tipo se dieron a conocer en el año 1979 y desde entonces el DSM quedó enmarcado en el tipo de métodos conocido como «Técnica de Multipolos Generalizada» [Eremin et al., 1999, p. 1296]. Inicialmente se usó para modelar la dispersión de ondas electromagnéticas generada por obstáculos que presentaran simetría axial en cuanto a su geometría, y que fueran perfectamente conductores o aislantes de la electricidad. En esta etapa inicial del método, era posible representar los campos electromagnéticos mediante combinaciones lineales de dipolos eléctricos y magnéticos ubicados sobre el eje de simetría, estas fuentes de excitación fueron denominadas ‘fuentes discretas’.

El nombre «Técnica de Multipolos Generalizada» (GMT, por sus iniciales en inglés) fue acuñado por Art Ludwig para referirse a un conjunto de métodos relacionados para resolver problemas electromagnéticos con valores en la frontera, dichos métodos fueron desarrollados en forma independiente por varios grupos de investigación. La característica en común de estas técnicas consiste en la expansión del campo a través de un conjunto de multipolos colocados lejos de la superficie de frontera [Wriedt & Eremin, 2018].

En 1983 se realizó una continuación (extensión) analítica de las fuentes discretas en el plano complejo. Esta mejora teórica permitió que se pudieran modelar obstáculos de forma alargada más general, ya que inicialmente el método sólo se aplicaba a objetos dispersores con simetría axial (sólidos de revolución). En 1985 el método fue ampliado para considerar fuentes de excitación no necesariamente de carácter axial, es decir, que las fuentes discretas pudieran ubicarse ya sea sobre el eje de simetría o bien sobre el plano complejo.

El siguiente escalón de mejora en el DSM se llevó a cabo en el año 1993. Esta actualización permitió que también pudiera tomarse en cuenta en los problemas el tipo de

polarización de las ondas electromagnéticas incidentes. Esta evolución del DSM ha permitido ampliar la gama de aplicaciones de este procedimiento en la actualidad, entre las cuales se pueden mencionar: la microscopía de reflexión interna total, el análisis de la dispersión de ondas evanescentes, la caracterización de películas delgadas, la reconstrucción de imágenes de nanoestructuras, así como la detección de partículas contaminantes y defectos en tarjetas de circuito impreso [Eremin & Wriedt, 2004, p. 53].

2.2. Justificación

Diversas técnicas se han desarrollado para analizar los problemas de dispersión de ondas. Cada una de estas técnicas, generalmente, tiene un rango de aplicabilidad, determinado por el tamaño del objeto dispersor en relación a la longitud de onda de la radiación incidente [Doicu et al., 2000, p. ix].

Considerando la longitud de onda (λ) de la radiación electromagnética, pueden considerarse tres casos:

1. El tamaño del elemento dispersor es mucho mayor a λ .
2. El tamaño del dispersor es mucho menor a λ .
3. El tamaño del elemento dispersor es del mismo orden de magnitud que λ .

Para tratar los casos 1 y 2 existe una variedad de métodos que se utilizan en el modelado de problemas de dispersión de ondas; sin embargo los métodos de los cuales se dispone para el caso 3 presentan como principales desventajas el considerable tiempo de cómputo que se requiere en su ejecución, así como la complejidad en su implementación, al tratarse de 'métodos de malla'.

Por otra parte, la existencia de diversas familias de soluciones fundamentales que se pueden aplicar en la implementación del DSM requiere investigar la influencia que la elección de esta familia puede tener en la precisión de los resultados, de acuerdo a la configuración del problema específico.

En esta investigación se realiza una presentación del Método de Fuentes Discretas (DSM) y se analizan diferentes familias de soluciones fundamentales y la posible influencia que la

elección de estas familias tenga sobre la precisión del método en diferentes configuraciones de problemas de dispersión de ondas.

El uso del método de fuentes discretas no está muy difundido, a pesar de sus ventajas con respecto a los métodos de malla. Este trabajo pretende contribuir a la difusión del conocimiento de esta herramienta.

2.3. Hipótesis

El método de fuentes discretas presenta diferencias significativas en su eficiencia, medida a través del error absoluto, dependiendo de la configuración del problema de dispersión (dominios acotados o dominios no acotados), así como de la familia de soluciones fundamentales que se utilice en cada caso, considerando estos dos factores como variables categóricas.

2.4. Objetivo

Aplicar el método de fuentes discretas a distintas configuraciones de problemas de dispersión, usando diferentes familias de soluciones fundamentales en cada caso, para comparar la eficiencia del método en estos planteamientos.

Capítulo III

EL MÉTODO DE FUENTES DISCRETAS (DSM)

3.1. *La ecuación de Helmholtz*

La siguiente ecuación diferencial parcial es conocida como la ecuación de Helmholtz:

$$\Delta u + k^2 u = 0 \quad (5)$$

donde Δ denota al operador de Laplace en n variables, $n \in \{2, 3\}$, y $k > 0$ es un parámetro conocido como 'número de onda' y u es la función incógnita. Esta ecuación es relevante dado que describe la intensidad de fenómenos físicos en todos los puntos de un dominio bajo estudio y que siguen una sola ley de evolución en el tiempo [Karapetyants & Kravchenko, 2022]. De hecho, tanto la ecuación de calor, como la ecuación de onda se reducen a la ecuación de Helmholtz en los casos en los cuales los fenómenos físicos correspondientes sean armónicos en el tiempo. Se dice que una función es armónica en el tiempo si su dependencia con respecto a la variable tiempo es de carácter sinusoidal. La importancia de la ecuación de Helmholtz, específicamente en los problemas de dispersión de fenómenos ondulatorios, radica en el hecho de que describe todas las soluciones armónicas en el tiempo para la ecuación de onda. De igual forma esta ecuación surge en problemas físicos modelados por la llamada *ecuación del telégrafo*, y por ecuaciones derivadas a partir de ella, como por ejemplo la *ecuación de calor*, la *ecuación de difusión* que nos permite modelar reacciones químicas y reacciones en cadena que tengan lugar en algún medio material; y por supuesto la misma ecuación de onda, ya mencionada.

De acuerdo a la clasificación estándar de las ecuaciones diferenciales parciales de segundo orden, la ecuación de Helmholtz es una ecuación del tipo elíptico. Recuerde que una ecuación diferencial elíptica es una ecuación diferencial parcial de segundo orden que puede escribirse en la forma canónica

$$u_{\xi\xi} + u_{\eta\eta} + \ell_1[u] = G(\xi, \eta) \quad (6)$$

donde u es la función que se está modelando, ξ y η son las variables independientes, ℓ_1 es un operador diferencial lineal de primer orden y G es una función [Pinchover & Rubinstein, 2005, p. 66]. De hecho, la parte principal del operador de Helmholtz no es otra cosa que el operador de Laplace y comparte propiedades con dicho operador; como la de que todas sus soluciones son funciones en el espacio C^∞ en sus respectivos dominios.

Se muestra a continuación, a manera de ejemplo, la forma en la que se deduce la ecuación de Helmholtz a partir de la ecuación de onda:

$$\frac{1}{c^2} \frac{\partial^2 U}{\partial t^2} = \Delta U \quad (7)$$

en la que se buscan soluciones que sean armónicas en el tiempo, es decir, se plantea que U tenga la forma

$$U(\mathbf{x}, t) = u(\mathbf{x})e^{-i\omega t} \quad (8)$$

donde u es la parte espacial de la solución, \mathbf{x} representa las coordenadas espaciales, t representa el tiempo, i es la unidad imaginaria, $\omega > 0$ es la frecuencia angular. Sustituyendo (8) en (7), se tiene:

$$-\frac{1}{c^2} u(\mathbf{x}) \omega^2 e^{-i\omega t} = e^{-i\omega t} \Delta u(\mathbf{x}) \quad (9)$$

$$\implies -\frac{1}{c^2} u(\mathbf{x}) \omega^2 = \Delta u(\mathbf{x}) \quad (10)$$

$$\implies \Delta u(\mathbf{x}) + \left(\frac{\omega}{c}\right)^2 u(\mathbf{x}) = 0 \quad (11)$$

y definiendo $k = \omega/c$, finalmente

$$\Delta u(\mathbf{x}) + k^2 u(\mathbf{x}) = 0 \quad (12)$$

con lo que se muestra que la parte espacial de la solución, i.e. $u(\mathbf{x})$ debe cumplir con la ecuación de Helmholtz. La importancia de buscar soluciones armónicas en el tiempo está basada en el hecho de que fenómenos arbitrarios dependientes del tiempo pueden ser representados a través de superposiciones de componentes armónicas en el tiempo [Karapetyants & Kravchenko, 2022].

3.2. Problemas internos y problemas externos

Se pueden plantear dos tipos de problemas de dispersión directos para la ecuación de Helmholtz, a saber: los problemas internos y los problemas externos.

Problemas internos

En este trabajo se llamarán también 'problemas en dominios acotados'. Sea $\Omega \subset \mathbb{R}^n, n \in \{2, 3\}$ un dominio acotado con frontera Lyapunov; un problema interno consiste en determinar una función que satisfaga la ecuación de Helmholtz en dicho dominio y que además cumpla con las condiciones de frontera estipuladas. Dependiendo del tipo de condiciones de frontera, se tienen problemas del tipo Dirichlet, del tipo Neumann o del tipo Robin (para mayor detalle véase la descripción de los problemas en la sección 3.6).

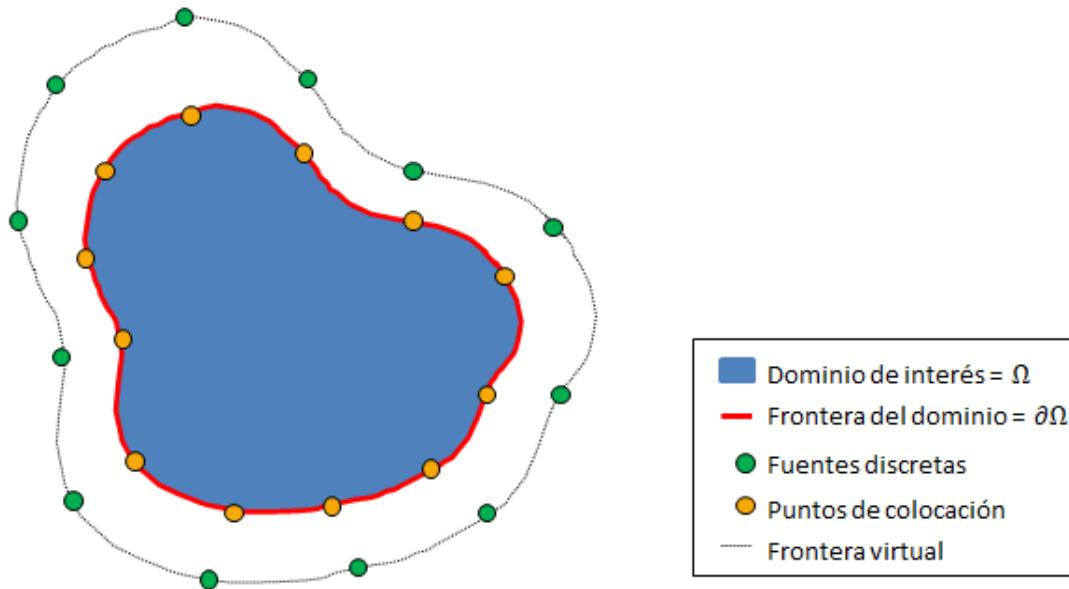


Figura 1: Esquema para los problemas del tipo interno.

Problemas externos

En este trabajo se llamarán también 'problemas en dominios no acotados'. Sea $V \subset \mathbb{R}^n, n \in \{2, 3\}$ una región acotada con frontera Lyapunov; un problema externo consiste en determinar una función que satisfaga la ecuación de Helmholtz en el dominio $\Omega = \mathbb{R}^n \setminus \overline{V}, n \in \{2, 3\}$, que cumpla con las condiciones de frontera estipuladas, y además una

condición adicional en el infinito. Dependiendo del tipo de condiciones de frontera, se tienen problemas del tipo Dirichlet, del tipo Neumann o del tipo Robin (para mayor detalle véase la descripción de los problemas en la sección 3.6).

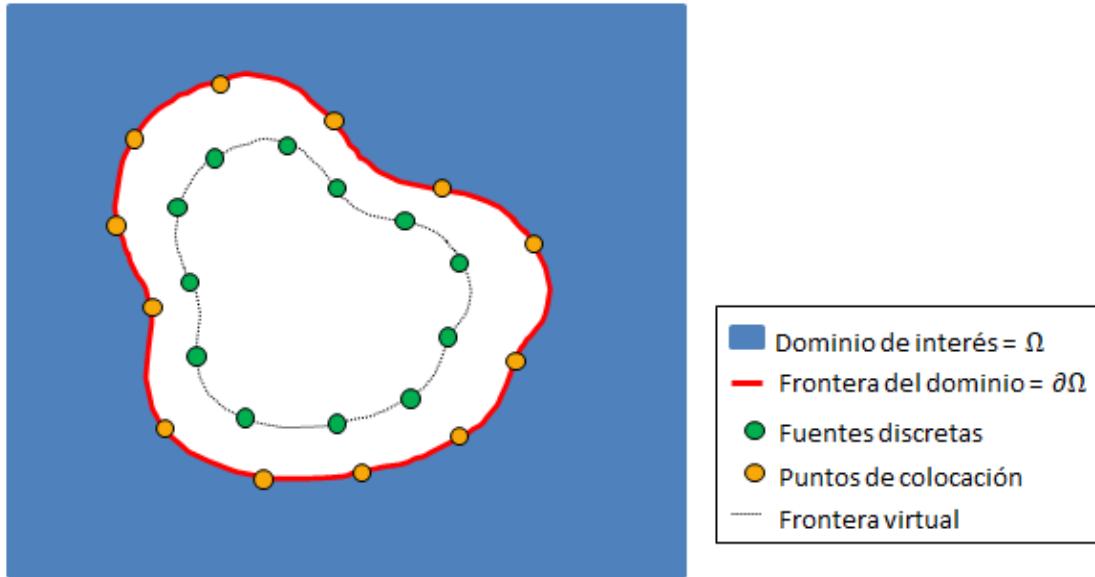


Figura 2: Esquema para los problemas del tipo externo.

3.3. Dispersión de ondas

Al mencionarse el estudio de la dispersión de ondas, se está haciendo referencia al estudio de la propagación de ondas armónicas en el tiempo en la presencia de obstáculos locales que presentan diferentes propiedades materiales [Eremin & Sveshnikov, 1993]. Considerando el caso acústico, que es el foco de la presente investigación, la descripción matemática de la dispersión de ondas armónicas en el tiempo por un obstáculo D conduce a problemas con condiciones de frontera para la ecuación de Helmholtz.

Para entender lo que sucede cuando una onda se encuentra con un obstáculo impenetrable, se comienza por definir algunos conceptos. Sea $u^{inc}(\mathbf{x}) = e^{ik\mathbf{x}\cdot\mathbf{d}}$ una onda plana con $|\mathbf{d}| = 1$, donde \mathbf{d} es el vector que define la dirección de propagación de dicha onda. A esto se le denomina *campo incidente* (u *onda incidente*). A la onda que se refleja al encuentro del obstáculo se le llama *campo dispersado* (u *onda dispersada*), se acostumbra

denotarlo por $u^{scat}(\mathbf{x})$. Finalmente $u^{tot}(\mathbf{x}) = u^{inc}(\mathbf{x}) + u^{scat}(\mathbf{x})$, es el *campo total*, el cual es el campo físico susceptible de ser medido en el dominio de interés Ω [Moiola, 2020].

Dado u^{inc} , que es conocido, lo que se quiere determinar es u^{tot} , de manera que cumpla la ecuación de Helmholtz en el dominio Ω , y que satisfaga las condiciones estipuladas sobre la frontera del mismo, i. e. en $\partial\Omega$. Dependiendo de las características del obstáculo, es posible imponer condiciones sobre u^{tot} , de manera que en la práctica lo que se determinará es u^{scat} de manera que satisfaga la ecuación de Helmholtz en la misma región y que cumpla con condiciones de frontera que dependerán de los valores de u^{inc} sobre $\partial\Omega$. Esto es lo que se realiza en los ejemplos de prueba que se presentan en este trabajo.

Cuando las ondas acústicas encuentran un obstáculo a través del cual no pueden propagarse, en la interfaz entre el obstáculo y el medio de propagación algunas condiciones de frontera deben ser establecidas. Dependiendo de las características del obstáculo y del medio, se pueden presentar dos situaciones:

- Las características del obstáculo y del medio son tales que la presión acústica en la frontera debe ser cero. En este caso se dice que el obstáculo es del tipo *sound-soft*. En este trabajo se consideran únicamente obstáculos de este tipo.
- Las características del obstáculo y del medio son tales que la velocidad normal en la frontera debe ser igual a cero. En este caso se dice que el obstáculo es del tipo *sound-hard*.

Las ondas planas reflejadas por interfaces *sound-soft* o por interfaces *sound-hard* (una interfaz se corresponde con la frontera del dominio Ω) tienen la misma amplitud que las ondas planas incidentes; en cuanto a la fase, ésta puede ser opuesta (en el caso *sound-soft*) o la misma (en el caso *sound-hard*).

Establecer los valores de u^{tot} en la frontera del obstáculo, físicamente se interpreta como prescribir la presión de la onda acústica. Las condiciones *sound-soft* se traducen en la condición Dirichlet $u^{tot} = 0$, entonces siendo la onda acústica total de la forma $u^{tot} = u^{inc} + u^{scat}$ donde u^{scat} denota la onda dispersada implica que la presión total deber ser cero

en la frontera, por lo que en la misma debe establecerse $u^{scat} = -u^{inc}$. De manera similar, el prescribir el valor de la derivada normal de u^{tot} en la frontera, equivale físicamente a prescribir la componente normal de la velocidad de la onda acústica. Las condiciones para obstáculos *sound-hard* se traducen entonces en la condición Neumann $\mathbf{n} \cdot \nabla u^{tot} = 0$ [Moiola, 2020].

Es importante mencionar que el parámetro k en la ecuación de Helmholtz se definió como ω/c , pero esto es válido sólo si se considera que el medio no presenta ningún tipo de amortiguamiento (no hay viscosidad), pero si se considera la propagación de ondas acústicas en un medio con coeficiente de amortiguamiento ζ , entonces el número de onda es dado por $k^2 = \omega(\omega + j\zeta)/c^2$. Se elige el signo de k , de manera tal que $\text{Im } k \geq 0$ [Doicu et al., 2000]. Para mostrar la razón de esta definición de k se presenta a continuación la forma en la que se deduce la ecuación de Helmholtz a partir de la ecuación de onda amortiguada:

$$\frac{1}{c^2} \left(\frac{\partial^2 U}{\partial t^2} + \zeta \frac{\partial U}{\partial t} \right) = \Delta U \quad (13)$$

donde ζ es el coeficiente de amortiguamiento, y en la que se buscan soluciones que sean armónicas en el tiempo, es decir, se plantea que U tenga la forma

$$U(\mathbf{x}, t) = u(\mathbf{x})e^{-j\omega t} \quad (14)$$

donde u es la parte espacial de la solución, \mathbf{x} representa las coordenadas espaciales, t representa el tiempo, j es la unidad imaginaria, $\omega > 0$ es la frecuencia angular. Sustituyendo (14) en (13), se tiene:

$$-\frac{1}{c^2} (u(\mathbf{x})\omega^2 e^{-j\omega t} + u(\mathbf{x})\zeta j\omega e^{-j\omega t}) = e^{-j\omega t} \Delta u(\mathbf{x}) \quad (15)$$

$$\implies -\frac{1}{c^2} (u(\mathbf{x})\omega^2 + u(\mathbf{x})\zeta j\omega) = \Delta u(\mathbf{x}) \quad (16)$$

$$\implies \Delta u(\mathbf{x}) + \left[\frac{\omega(\omega + j\zeta)}{c^2} \right] u(\mathbf{x}) = 0 \quad (17)$$

y definiendo $k^2 = \omega(\omega + j\zeta)/c^2$, finalmente

$$\Delta u(\mathbf{x}) + k^2 u(\mathbf{x}) = 0 \quad (18)$$

con lo que se muestra que la parte espacial de la solución, i.e. $u(\mathbf{x})$ debe cumplir con la ecuación de Helmholtz también en el caso amortiguado.

3.4. Problemas Dirichlet en dominios acotados

Sea $\Omega \subset \mathbb{R}^2$ un dominio abierto, acotado y con frontera Lyapunov. El problema Dirichlet para la ecuación de Helmholtz es: dado un término F definido en Ω y un dato de frontera f definido en $\partial\Omega$, encontrar u en Ω tal que

$$\Delta u + k^2 u = -F \text{ en } \Omega; \quad u = f \text{ en } \partial\Omega. \quad (19)$$

Es conocido que para el caso $k = 0$, el problema anterior está bien planteado cuando los datos F, f son suficientemente suaves. Sin embargo estas condiciones no son suficientes para el caso de la ecuación de Helmholtz. Por ejemplo, si se considera el rectángulo $\Omega = (0, L_1) \times (0, L_2)$, entonces para todo $j_1, j_2 \in \mathbb{N}$ el campo $u_{j_1, j_2}(\mathbf{x}) = \sin(\frac{j_1\pi}{L_1}x_1)\sin(\frac{j_2\pi}{L_2}x_2)$ vale cero en la frontera del dominio y además es solución de $\Delta u + k_{j_1, j_2}^2 u = 0$ con $k_{j_1, j_2}^2 = \frac{j_1^2\pi^2}{L_1^2} + \frac{j_2^2\pi^2}{L_2^2}$. De manera que hay una cantidad infinita de valores de k tales que el problema Dirichlet homogéneo ($F = 0, f = 0$) para la ecuación de Helmholtz admite soluciones no triviales, es decir, para esos valores específicos de k el problema no está bien planteado.

A las soluciones del problema Dirichlet homogéneo para la ecuación de Helmholtz se les llama *funciones propias Dirichlet* del laplaciano, con valor propio $\Lambda = k^2$, dado que satisfacen $-\Delta u = \Lambda u$. La situación del ejemplo anterior se presenta no sólo para dominios rectangulares, y se presenta incluso si no es posible determinar en forma explícita los valores y las funciones propias.

De hecho, dados Ω y k , sólo pueden darse dos situaciones:

- Si Ω y k son tales que el problema Dirichlet homogéneo ($F = 0, f = 0$) para la ecuación de Helmholtz admite únicamente la solución trivial $u = 0$, i. e. k^2 no es un valor propio, entonces el problema admite una solución única para cualquier $F \in L^2(\Omega)$.
- Si no se cumple la condición anterior, i. e. k^2 es un valor propio, entonces el problema no está bien planteado para ninguna F . Es decir, podría no haber soluciones para algunos F , y soluciones no únicas para otros F .

El enunciado del teorema de buen planteamiento para el problema Dirichlet para la ecuación de Helmholtz, en dominios acotados, se encuentra en la sección de fundamentación

teórica de la presente investigación (sección 3.7).

3.5. Problemas Dirichlet en dominios no acotados

Sea $V \subset \mathbb{R}^2$ un dominio acotado con frontera Lyapunov. Defina $\Omega = \mathbb{R}^2 \setminus \overline{V}$. Se asume que Ω es conexo, es decir, V no posee agujeros. Se considerarán únicamente obstáculos del tipo *sound-soft*. Sea u^{inc} la onda incidente. Dado que $u^{tot} = 0$ para este tipo de obstáculos, el problema consiste en determinar el campo u^{scat} dispersado por V , que sea solución de la ecuación de Helmholtz en Ω , y tal que $u^{scat} = -u^{inc}$ en $\partial\Omega$.

El campo dispersado es producido por la interacción del obstáculo y el campo incidente u^{inc} . Así que su comportamiento debería ser el de una onda propagándose hacia el infinito desde V .

Definición 26 (Solución radiante). *Sea $u \in H_{loc}^1(\mathbb{R}^2 \setminus B_R)$ solución de la ecuación de Helmholtz en el complemento de una bola. Se dice que u es radiante si satisface la condición de radiación de Sommerfeld:*

$$|\partial_r u - iku| = o(r^{-1/2}), \quad r \rightarrow \infty. \quad (20)$$

La condición de Sommerfeld se debe cumplir uniformemente en todas direcciones, i. e.

$$\lim_{r \rightarrow \infty} \sup_{\theta \in [0, 2\pi]} \sqrt{r} |\partial_r u(r, \theta) - iku(r, \theta)| = 0. \quad (21)$$

El cumplimiento de esta condición es indispensable para que la solución de los problemas del tipo de externo tenga sentido físico, ya que garantiza que el campo dispersado sólo tenga componentes que se alejan del obstáculo hacia el infinito, y que no existan componentes que vayan del infinito hacia el obstáculo.

Tomando en cuenta lo anterior, el problema externo Dirichlet se plantea así: Sea V un dominio acotado con frontera Lyapunov, $k > 0$ y $f \in H^{1/2}(\partial\Omega)$. Se dice que $u \in H_{loc}^1(\overline{\Omega})$ es solución del problema externo Dirichlet para la ecuación de Helmholtz si

$$\begin{aligned} \Delta u + k^2 u &= 0 && \text{en } \Omega, \\ u &= f && \text{en } \partial\Omega \\ u &\text{ es radiante.} \end{aligned} \quad (22)$$

El problema de dispersión *sound-soft* es un problema externo Dirichlet con $u = u^{scat}$ y $f = -u^{inc}$. Este problema es bien planteado y el enunciado del teorema de buen planteamiento se puede ver en la sección de fundamentación teórica de la presente investigación (sección 3.7).

Es importante mencionar que los problemas internos y externos tratados en esta tesis, pertenecen al tipo de problemas denominado: *problema directo de dispersión*, lo que quiere decir que se conoce la onda incidente, se conoce la geometría del obstáculo y lo que se desea determinar es el campo dispersado. En aplicaciones, tales como imagenología médica, detección remota atmosférica, detección de fallas en materiales, imágenes de radar, etc., es importante considerar los *problemas inversos de dispersión*, en los cuales el campo dispersado es conocido (mediante mediciones) y lo que se quiere determinar es la geometría del obstáculo y/o la onda incidente. Los problemas inversos por lo general son mal planteados y mucho más difíciles que los problemas directos, tanto desde el punto de vista teórico como del computacional.

3.6. Descripción del problema

Al mencionarse el estudio de la dispersión de ondas, se está haciendo referencia al estudio de la propagación de ondas armónicas en el tiempo en la presencia de obstáculos locales que presentan diferentes propiedades materiales [Eremin & Sveshnikov, 1993]

En la presente investigación, el método de fuentes discretas se aplicará a la ecuación de Helmholtz:

$$\Delta u + k^2 u = 0 \quad (23)$$

donde $k \in \mathbb{R}$, y la solución se busca en un dominio abierto y simplemente conexo, $\Omega \subset \mathbb{R}^2$. La frontera del dominio, denotada por $\partial\Omega$, debe ser suficientemente suave (generalmente basta con que sea Lipschitz continua), siendo éste uno de los requisitos para la existencia y unicidad de la solución, una vez que a la ecuación se le añadan las condiciones de frontera [Golberg & Chen, 1998].

Esta ecuación se presenta cuando se buscan soluciones *armónicas en el tiempo* para la ecuación de onda, i.e. soluciones que dependan de la variable tiempo a través de una función

senoidal o cosenoidal. Al agregar condiciones de frontera, se pueden plantear tres tipos de problemas [Karapetyants & Kravchenko, 2022, p. 359]:

Problemas del primer tipo (o problema Dirichlet): consisten en determinar una función $u(x) \in C^2(\Omega) \cap C(\bar{\Omega})$ que cumpla con

$$\begin{aligned}\Delta u(x) + k^2 u(x) &= 0, & x \in \Omega \\ u(x) &= f(x), & x \in \partial\Omega\end{aligned}\tag{24}$$

donde f es una función prestablecida.

Problemas del segundo tipo (o problema Neumann): en este caso el objetivo es encontrar una función $u(x) \in C^2(\Omega) \cap C^1(\bar{\Omega})$ que satisfaga

$$\begin{aligned}\Delta u(x) + k^2 u(x) &= 0, & x \in \Omega \\ \frac{\partial u(x)}{\partial \mathbf{n}} &= g(x), & x \in \partial\Omega\end{aligned}\tag{25}$$

donde g es una función prestablecida y $\frac{\partial u(x)}{\partial \mathbf{n}}$ denota la derivada normal de la función u .

Problemas del tercer tipo (o problema Robin): consisten en determinar una función $u(x) \in C^2(\Omega) \cap C^1(\bar{\Omega})$ que cumpla con

$$\begin{aligned}\Delta u(x) + k^2 u(x) &= 0, & x \in \Omega \\ \frac{\partial u(x)}{\partial \mathbf{n}} + k(x)u(x) &= h(x), & x \in \partial\Omega\end{aligned}\tag{26}$$

donde h y k son funciones conocidas y $\frac{\partial u(x)}{\partial \mathbf{n}}$ denota la derivada normal de la función u .

En general, si se quiere expresar un problema de dispersión, sin especificar las condiciones de frontera, se puede hacer de la siguiente manera:

$$\begin{aligned}\Delta u + k^2 u &= 0, & \text{en } \Omega \\ Bu &= j, & \text{sobre } \partial\Omega\end{aligned}\tag{27}$$

donde j es una función prestablecida, y B es un operador que se define de manera que se obtenga alguno de los problemas (24), (25) o (26).

Los planteamientos para los tres tipos de problemas, en el caso de dominios no acotados, se realiza de manera similar, pero debe añadirse en cada caso la condición de radiación de Sommerfeld presentada ya en la sección 3.5.

3.7. Fundamentación teórica

La aplicación del método de fuentes discretas requiere el estudio de los siguientes aspectos:

1. La existencia de la solución del problema bajo estudio.
2. La unicidad de la solución.
3. La completitud de los sistemas de soluciones fundamentales.

Para el estudio de los dos primeros puntos se recurre a la teoría de ecuaciones integrales en la mayoría de los textos; sin embargo una desventaja de este tratamiento es que para probar de forma constructiva la existencia de soluciones para los problemas de dispersión se requiere la suposición de que las fronteras de los dominios son de clase C^2 . Es posible debilitar ligeramente esta suposición al permitir *fronteras Lyapunov* en lugar de las fronteras C^2 y aún permanecer en el marco teórico de los operadores compactos en los espacios Hölder de funciones continuas. El tratamiento teórico de los problemas Dirichlet para fronteras Lyapunov es muy similar al que se da para fronteras C^2 .

Sin embargo, la situación cambia de forma drástica si se permite que la frontera tenga aristas y esquinas. En este caso lo conveniente es permitir que los dominios posean fronteras del tipo Lipschitz [Colton. & Kress, 2013, p. 55].

A continuación se presentan los teoremas de buen planteamiento de los problemas Dirichlet internos y externos para la ecuación de Helmholtz, el primero de estos resultados fue tomado del texto de [Moiola, 2020]; el segundo se extrajo del texto de [McLean, 2000]. Para las definiciones de los espacios funcionales involucrados, consúltese la sección 1.2. Téngase en cuenta que k representa el número de onda en la ecuación de Helmholtz:

Teorema 27 (Buen planteamiento del problema Helmholtz-Dirichlet interno). Para un dominio acotado Ω , con frontera Lipschitz, existe una sucesión de números positivos (los valores propios del problema Dirichlet homogéneo) $k_1 < k_2 < \dots$, con $k_j \rightarrow \infty$, tales que:

- Si $k = k_j$ para algún j , entonces el problema Dirichlet (19) no está bien planteado.

En particular, el caso homogéneo con $F = 0$ y $f = 0$ admite soluciones no triviales.

- Si $0 < k \neq k_j$ para todo j , entonces el problema Dirichlet (19) está bien planteado en $H_0^1(\Omega)$ para toda $F \in L^2(\Omega)$ y $f \in H^{\frac{1}{2}}(\partial\Omega)$.

Teorema 28 (Buen planteamiento del problema Helmholtz-Dirichlet externo).

Sea $\Omega = \mathbb{R}^2 \setminus \overline{V}$, donde V es un dominio acotado con frontera Lipschitz. Si $f \in H^{\frac{1}{2}}(\partial\Omega)$, entonces el problema Dirichlet externo para la ecuación de Helmholtz (22), tiene una única solución radiante $u \in H_{loc}^1(\Omega)$.

Una vez que se cumplen estas condiciones se requiere garantizar que el espacio de combinaciones lineales de soluciones fundamentales sea denso en el espacio de soluciones del operador diferencial de Helmholtz, es decir, se necesita mostrar que los sistemas de soluciones fundamentales sean completos. Para ello se presentan a continuación los teoremas de completitud del artículo [Khmelnitskaya et al., 2003] (en donde pueden consultarse las demostraciones), con algunos cambios en la notación. Sea Γ una superficie cerrada en \mathbb{R}^3 que sea frontera de un dominio acotado Ω^+ y de un dominio no acotado $\Omega^- = \mathbb{R}^3 \setminus \overline{\Omega^+}$. El símbolo Γ^- denotará una superficie encerrada en Ω^+ y encerrando un dominio V , el símbolo Γ^+ denotará una superficie que encierra a $\overline{\Omega^+}$.

Se denota por $\{s_j^-\}_{j=1}^\infty$ al conjunto de puntos distribuidos en Γ^- y denso en Γ^- , y por $\{s_j^+\}_{j=1}^\infty$ al conjunto de puntos distribuidos en Γ^+ y denso en Γ^+ . Se asocian los sistemas de soluciones fundamentales $\{\phi_k(x - s_j^-)\}_{j=1}^\infty$ y $\{\phi_k(x - s_j^+)\}_{j=1}^\infty$ a los conjuntos anteriores, respectivamente. Para simplificar un poco la notación, se define $\phi_{k,j}^-(x) := \phi_k(x - s_j^-)$ y $\phi_{k,j}^+(x) := \phi_k(x - s_j^+)$. Se considera que las singularidades de las funciones del primer sistema están distribuidas en la superficie interior Γ^- y por lo tanto solucionan la ecuación de Helmholtz en Ω^- , suponga que también cumplen con la condición de radiación de Sommerfeld. Suponga además que las funciones del segundo sistema tienen sus singularidades en la superficie externa Γ^+ y por lo tanto son soluciones de la ecuación de Helmholtz en Ω^+ .

Los resultados se presentan de la manera siguiente: primero el teorema de la completitud en $L_2(\Omega) \cap \ker(\Delta + k^2)$. Después se presenta el resultado ampliado a $H^s(\Omega) \cap \ker(\Delta +$

k^2). Finalmente, como $H^s(\Gamma)$ puede ser considerado como un espacio de las trazas de las correspondientes soluciones de la ecuación de Helmholtz, se puede probar la completitud de los sistemas de soluciones fundamentales para el operador de Helmholtz en este espacio.

Resultados para dominios acotados

Teorema 29 *Sea Ω^+ un dominio acotado en \mathbb{R}^3 con frontera Lyapunov Γ . El sistema de funciones $\{\phi_{k,j}^+\}_{j=1}^\infty$ es completo en $L_2(\Omega^+) \cap \ker(\Delta + k^2)$.*

Teorema 30 *Bajo las hipótesis del teorema 29 el sistema de funciones $\{\phi_{k,j}^+\}_{j=1}^\infty$ es completo en $H^s(\Omega^+) \cap \ker(\Delta + k^2)$, $s \geq 0$.*

Teorema 31 *Sea Γ una superficie cerrada suficientemente suave (el espacio $H^s(\Gamma)$ está definido). El sistema de funciones $\{\phi_{k,j}^+\}_{j=1}^\infty$ es completo en $H^s(\Gamma)$, $s \in \mathbb{R}$.*

Resultados para dominios no acotados

Sea B_R una bola arbitraria con un radio R tal que $\Omega^+ \subset B_R$. Denote $\Omega_R^- = \Omega^- \cap B_R$. Por lo tanto Ω_R^- es un dominio en \mathbb{R}^3 con frontera consistente en Γ y en la frontera de la esfera, i. e. ∂B_R .

Teorema 32 *Sea Γ una superficie Lyapunov y suponga que k^2 no es un eigenvalor del problema Dirichlet en V . El sistema de funciones $\{\phi_{k,j}^-\}_{j=1}^\infty$ es completo en $L_2(\Omega_R^-) \cap \ker(\Delta + k^2)$.*

Teorema 33 *Bajo las hipótesis del teorema 32 el sistema de funciones $\{\phi_{k,j}^-\}_{j=1}^\infty$ es completo en $H^s(\Omega_R^-) \cap \ker(\Delta + k^2)$, $s \geq 0$.*

Teorema 34 *Sea Γ una superficie cerrada suficientemente suave y suponga que k^2 no es un eigenvalor del problema Dirichlet en V . El sistema de funciones $\{\phi_{k,j}^-\}_{j=1}^\infty$ es completo en $H^s(\Gamma)$, $s \in \mathbb{R}$.*

En cuanto a las herramientas que se requieren en la implementación numérica, para la determinación de los coeficientes de las combinaciones lineales de soluciones fundamentales, se puede hacer uso de algoritmos de optimización no lineal, si se implementa un modelo

de fuentes 'móviles'. En el caso de usar un modelo de fuentes fijas, se puede usar simplemente algún algoritmo de resolución de sistemas de ecuaciones lineales [Fairweather & Karageorghis, 1998].

La implementación del método se realizará a través del software MATLAB[®]. La exactitud del método se determinará con base en el conocimiento de soluciones exactas de los problemas elípticos abordados (se considerarán problemas con dominios en \mathbb{R}^2).

3.8. Metodología

La idea física del método de fuentes discretas está ligada al principio de Huygens, el cual establece que: *todo punto de un frente de onda inicial puede considerarse como una fuente de ondas esféricas secundarias que se extienden en todas las direcciones con la misma velocidad, frecuencia y longitud de onda que el frente de onda del que proceden*. De esta manera, el elemento dispersor de las ondas, es decir, el obstáculo, siendo una fuente de campo secundario, es sustituido por un conjunto de fuentes ficticias, las cuales generan el mismo campo secundario que el obstáculo real [Doicu et al., 2000].

Para abordar un problema de dispersión de ondas, considerando una solución que sea armónica en el tiempo, como el que se plantea en la descripción del problema (27), el DSM plantea que la solución sea aproximada a través de una combinación lineal de soluciones fundamentales para el operador de Helmholtz. Es decir, se plantea que u tenga la forma:

$$u(\mathbf{x}, \mathbf{c}) = \sum_{j=1}^N c_j \phi(\mathbf{x} - \mathbf{s}_j) \quad (28)$$

donde la solución fundamental $\phi(\mathbf{x} - \mathbf{s}_j)$ satisface en forma exacta la ecuación diferencial parcial homogénea, excepto en el punto singular (fuente) \mathbf{s}_j , pero no necesariamente satisface las condiciones de frontera. De hecho, los valores de los coeficientes c_j se determinarán de manera que las condiciones de frontera se satisfagan. Los puntos singulares (fuentes discretas) deben colocarse fuera del dominio donde se busca la solución. Para obtener los valores de los c_j se usa el método de colocación sobre la frontera, es decir, se plantea:

$$\sum_{j=1}^N c_j B[\phi(\mathbf{x}_i - \mathbf{s}_j)] = g(\mathbf{x}_i), \quad \mathbf{x}_i \in \partial\Omega, i = 1, \dots, M \geq N \quad (29)$$

donde B es un operador que define el tipo de condiciones de frontera, lo cual define un sistema de ecuaciones lineales. Si $M > N$ se usará un procedimiento de mínimos cuadrados para su resolución, mientras que en el caso $M = N$, se llega a un sistema de N ecuaciones con N incógnitas que se puede resolver con los métodos usuales [Chen et al., 2009].

Antes de exponer los pasos que se seguirán en la implementación del DSM, cabe mencionar que Bogomolny (1985) propuso colocar las fuentes puntuales (fuentes discretas), *i. e.* las singularidades de las soluciones fundamentales, uniformemente distribuidas sobre la frontera de un círculo que *abarque* al dominio computacional Ω . Por otro lado Heise (1976) propone colocarlas sobre una frontera virtual, denotada por $\partial\Omega'$, que sea equidistante de la frontera real, $\partial\Omega$. El autor Tsai (véase [Tsai et al., 2006]) sugiere seguir el método de Heise, que generalmente proporciona mejores resultados en dominios con formas básicas; de manera que, siguiendo tal recomendación en lo posible, el procedimiento para colocar las fuentes puntuales (cuando las formas de la frontera real y la virtual coinciden) que se siguió en los experimentos numéricos en esta investigación es el siguiente: considerando el dominio computacional en cuestión, Ω , y siendo $\partial\Omega$ su frontera, entonces:

- I. Los puntos de colocación se distribuyen uniformemente sobre $\partial\Omega$.
- II. Se localiza el centro geométrico, x_c , del dominio Ω (el centro geométrico también es conocido como **centroide**. El centroide es entonces el centro de masa de una lámina plana bidimensional o de un sólido tridimensional. En este trabajo se consideran todos los dominios como láminas con densidad de masa no nula y uniforme).
- III. Las fuentes puntuales se colocan de acuerdo a la ecuación siguiente

$$s = x_k + (f.d.)(x_k - x_c) \quad (30)$$

donde s_j y x_k son las coordenadas de las ubicaciones de las fuentes puntuales y de los puntos de colocación, respectivamente; $f.d.$ (factor de distancia) es un parámetro escalar que determina la ‘cercanía’ de las fuentes puntuales, s_j , a la frontera real, $\partial\Omega$. Nótese que de acuerdo a la ecuación anterior, la ubicación de una fuente puntual dada, es la imagen homotética de un correspondiente punto de colocación; el centro

de homotecia es x_c y la razón de homotecia es $f.d.$, que por supuesto debe tener un valor mayor que 1 para problemas internos, y menor que 1 para problemas externos, ya que las fuentes puntuales deben quedar ‘fuera’ de la cerradura del dominio Ω .

La figura 3 muestra de manera esquemática el resultado de aplicar el procedimiento anterior a cierto dominio (en este caso un dominio rectangular):

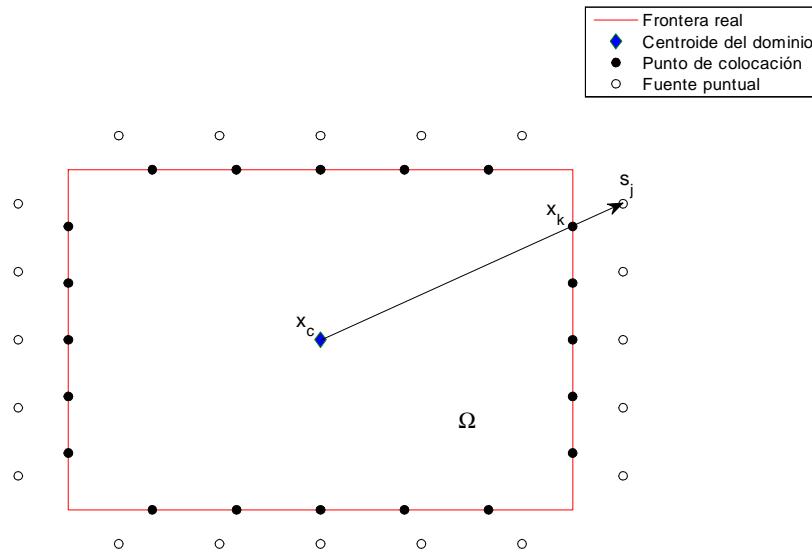


Figura 3: Esquema del procedimiento de ubicación de las fuentes discretas, para un problema interno.

Sin embargo, uno de los propósitos de esta investigación es explorar distintas variantes en cuanto a las configuraciones de las fronteras virtuales, de manera que cuando se usó una frontera virtual de distinta forma a la de la frontera real, se usó el procedimiento siguiente:

- I. Los puntos de colocación se distribuyen uniformemente sobre $\partial\Omega$.
- II. Se localiza el centro geométrico, x_c , del dominio Ω .
- III. Las fuentes puntuales se generan sobre la frontera virtual, de manera que x_c también sea centroide de esta frontera y de manera que estén distribuidas uniformemente. Los parámetros que definen el tamaño de la frontera virtual se obtienen al multiplicar los parámetros de la frontera real por el parámetro $f.d.$. Nótese que en este caso las fuentes puntuales no son imágenes homotéticas de los puntos de colocación. Se debe cumplir que $f.d. \in (0, 1)$ para problemas externos; $f.d. \in (1, \infty)$ para problemas internos.

Una vez explicada la metodología seguida para ubicar a las fuentes puntuales, ahora se continúa con la exposición de los pasos por seguir en la implementación del Método de Fuentes Discretas. La implementación numérica del DSM consiste en:

1. Si la ecuación diferencial correspondiente a nuestro problema es $\mathcal{L}u = 0$ en Ω , determinar una función, $\phi(x, s)$, tal que $\mathcal{L}_x\phi(\cdot, s) = \delta_s$, con $s \in \mathbb{R}^n$ (*i. e.* determinar la solución fundamental para la ecuación dada).
2. Seleccionar $\{x_k\}_{k=1}^N$ de puntos sobre $\partial\Omega$ (*frontera real*). Éstos son los puntos de colocación.
3. De acuerdo a la configuración específica del experimento, determinar la colocación de las fuentes discretas $\{s_j\}_{j=1}^N$ usando alguno de los procedimientos de los párrafos inmediatos anteriores de esta misma sección.
4. Se toma una de las s_j y se obtiene $\phi(x, s_j)$; el resultado de dicha sustitución debe ser acompañado por un coeficiente c_j que queda por determinar. Se realiza el mismo procedimiento para las demás s_j y finalmente se obtiene $w(x) := \sum_{j=1}^N c_j\phi(x, s_j)$.

5. Se toma uno de los x_k (ver el paso 2) y se sustituye en la condición de frontera, se obtiene después $w(x_k)$ y se establece una ecuación entre los resultados correspondientes. Se realiza el mismo procedimiento para las demás x_k . De esta manera, se obtiene el sistema lineal (29).
6. Se soluciona el sistema lineal (29), obtenido en el paso 5. De esta manera quedan determinados los coeficientes c_j de la combinación lineal del paso 4; dicha combinación lineal de soluciones fundamentales es la aproximación, $u_N(x; c) = \sum_{j=1}^N c_j \phi(x, s_j)$, a la solución del problema que está en cuestión.

Se presenta a continuación un cuadro con las soluciones fundamentales correspondientes a algunas de las ecuaciones diferenciales parciales más comunes [Tsai et al., 2006] [Smyrlis, 2009] [Balakrishnan & Ramachandran, 2000] [Barnett & Betcke, 2008].

Operador involucrado	Solución fundamental (\mathbb{R}^2)	Solución fundamental (\mathbb{R}^3)
Laplaciano	$\frac{-1}{2\pi} \ln\left(\frac{ x-s_i }{R}\right)$, $R > 0$	$\frac{1}{\omega_2} x - s_i ^{-1}$, donde ω_2 es el área de la superficie de la esfera unitaria S^2 en \mathbb{R}^3 .
Helmholtz	$H_0^{(1)}(k x - s_i)$, $H_0^{(1)}$ es la función de Hankel del primer tipo, de orden cero	$\frac{e^{-jk x - s_i }}{ x - s_i }$, j es la unidad imaginaria.
Helmholtz modificado	$K_0(k x - s_i)$, K_0 es la función de Bessel modificada del segundo tipo, de orden cero	$\frac{e^{-k x - s_i }}{ x - s_i }$
Biarmónico	$c_{1i} \ln(x - s_i) + c_{2i} x - s_i ^2 \ln(x - s_i)$	$-\frac{1}{2\omega_2} x - s_i $, donde ω_2 es el área de la superficie de la esfera unitaria S^2 en \mathbb{R}^3 .

x es un punto de prueba en el dominio Ω ; $s_i \in \partial\Omega'$ son las ubicaciones de las fuentes puntuales. R representa el reescalamiento. k es el número de onda.

Cuadro 1: Soluciones fundamentales para algunas ecuaciones diferenciales parciales comunes

En el presente trabajo se analizará la exactitud del DSM, así como su estabilidad numérica; para ello se seleccionarán *ejemplos de prueba*, es decir, problemas con soluciones exactas conocidas [dichas soluciones se denominarán *soluciones de prueba*], se aplicará el método, se tomarán puntos en el dominio del problema [denominados de aquí en adelante como *puntos de prueba*] y se determinará el error absoluto. De esta manera se pretende observar cómo varía la exactitud y la estabilidad numérica del método con respecto a los siguientes factores: la distancia de la frontera virtual, $\partial\Omega'$, al dominio del problema (Ω); la cantidad de términos, N , usados en la aproximación; la configuración de la frontera virtual usada y la manera en que se distribuyen los puntos de colocación y las fuentes discretas.

Capítulo IV

EJEMPLOS DE PRUEBA

NOTA IMPORTANTE: TODAS LAS FUNCIONES Y SCRIPTS MENCIONADOS EN ESTE CAPÍTULO FUERON CREADOS EN LA PLATAFORMA MATLAB[®] R2014a. LOS CÓDIGOS FUENTE DE LAS *FUNCIONES GENERALES* PUEDEN SER CONSULTADOS EN EL **APÉNDICE A**, MIENTRAS QUE LOS CÓDIGOS PARA CADA UNO DE LOS EJEMPLOS SE ENCUENTRAN EN EL **APÉNDICE B** DE ESTA INVESTIGACIÓN.

4.1. Acerca del parámetro factor de distancia (*f. d.*)

En los experimentos de este capítulo las fuentes discretas se colocaron sobre una curva a la que se le denomina *frontera virtual*, para distinguirla de la frontera del dominio (*frontera real*). La generación de la frontera virtual se controló a través del parámetro que fue llamado *factor de distancia* (*f. d.*). Para ello los parámetros que definen a la frontera real se multiplicaron por el factor de distancia, obteniéndose así los parámetros que definen a la frontera virtual. Por ejemplo: si la frontera del dominio Ω es un rectángulo cuyas dimensiones son: $base = a$, $altura = b$ y el valor del parámetro factor de distancia es $f. d. < 1$, y se quiere que la frontera virtual sea una elipse, entonces las dimensiones de esta elipse serán: $eje\ horizontal = a \times f.d.$, $eje\ vertical = b \times f.d.$; con lo que además se deduce que en este caso la frontera virtual se encuentra totalmente en el interior del área acotada por el rectángulo (véase la figura 4); si se asignan a *f. d.* valores positivos aún menores se obtienen fronteras virtuales más alejadas de la frontera real.

En los experimentos de esta sección, cuando la forma de la frontera real y la virtual coinciden (son ambas elipses, o ambas son rectangulares), el procedimiento anterior implica que la frontera virtual es una transformación homotética de la frontera real, siendo *f.d.* la razón de homotecia.

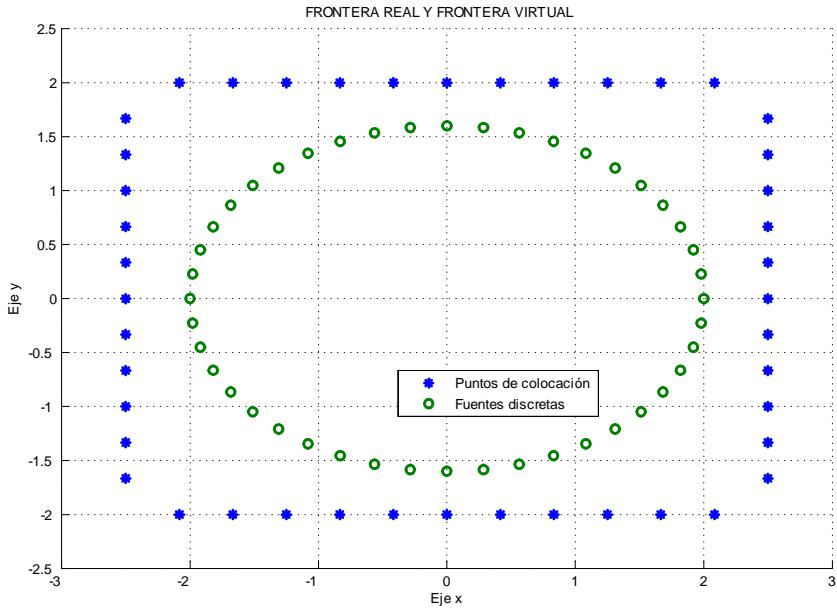


Figura 4: Esquema de la ubicación de la frontera real y de la frontera virtual, usando para esta última un factor de distancia f. d. = 0.8

Nótese entonces que si $f. d. \rightarrow 1$, entonces la distancia entre la frontera virtual y la frontera real tiende a cero. De manera que el parámetro factor de distancia se puede interpretar como un indicador del alejamiento de la frontera virtual con respecto a la frontera real (frontera del dominio). Dado que el método exige que las fuentes discretas no se encuentren en el dominio Ω , se tiene que para el caso de problemas internos, $f. d.$ debe elegirse en el intervalo $(1, \infty)$; en el caso de problemas externos debe elegirse en el intervalo $(0, 1)$.

4.2. Acerca de la generación de los puntos de prueba

En todos los experimentos de este capítulo, a los puntos en el dominio en los cuales se determinó el error absoluto, se les denominó *puntos de prueba*. Dichos puntos fueron generados sobre sucesivas curvas homotéticas a la frontera del dominio (frontera real). De manera que aquellos puntos de prueba ubicados sobre curvas homotéticas con razón de homotecia próxima a 1.0 son puntos de prueba más cercanos a la frontera del dominio, que aquellos punto de prueba ubicados sobre curvas homotéticas con razón de homotecia más

alejada de dicho valor.

Por ejemplo: si la frontera del dominio Ω es un rectángulo cuyas dimensiones son: $base = 4$, $altura = 4$ y el tipo de problema que se aborda es uno del tipo externo, los puntos de prueba se generarían como se puede ver en la figura 5.

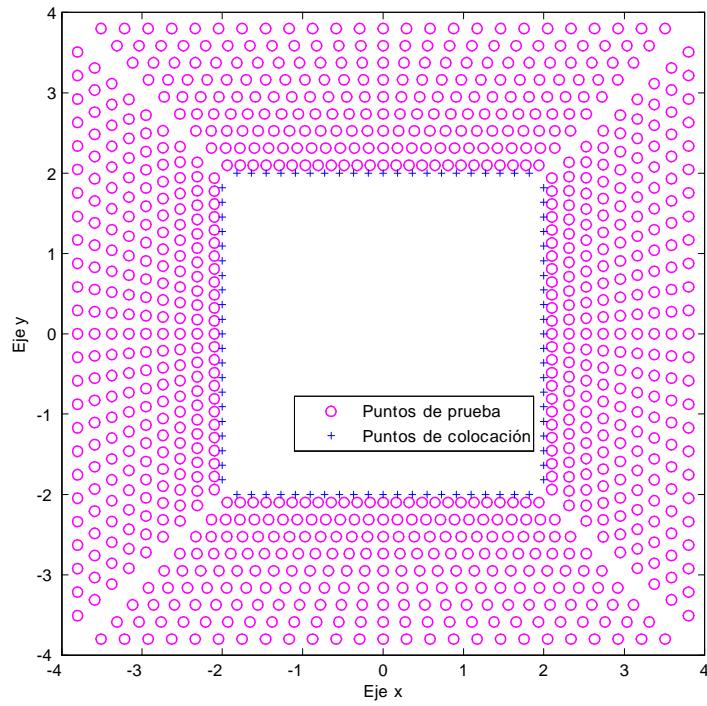


Figura 5: Esquema de la ubicación de los puntos de prueba. En este caso se considera un problema del tipo externo.

Nótese que en el caso de los problemas del tipo interno, los valores de las razones de homotecia para los puntos de prueba deben pertenecer al intervalo $(0, 1)$; mientras que para los problemas del tipo externo, deben tomarse en el intervalo $(1, \infty)$.

4.3. Problemas internos

4.3.1. Ejemplo 1

$$\begin{cases} \Delta u + k^2 u = 0, & k \in \mathbb{R}, (x, y) \in \Omega \\ u(x, y) = \cos \frac{kx}{\sqrt{2}} \cos \frac{ky}{\sqrt{2}}, & (x, y) \in \partial\Omega. \end{cases}$$

En este ejemplo se considera un dominio elíptico $\Omega = \{(x, y) \in \mathbb{R}^2 : \frac{x^2}{25} + y^2 < 1\}$, cuyo semieje horizontal tiene longitud *cinco* y la longitud del semieje vertical es *uno*.

Para la validación y el cálculo de los errores absolutos se utilizó la

Solución exacta: $u(x, y) = \cos \frac{kx}{\sqrt{2}} \cos \frac{ky}{\sqrt{2}}$

con el valor $k = \sqrt{2}$.

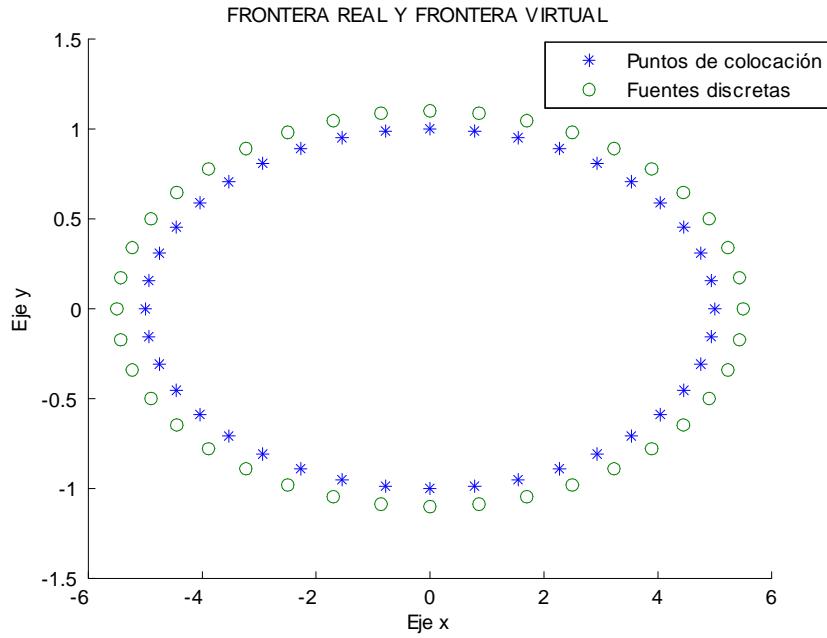


Figura 6: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 1. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

En la implementación numérica las fuentes discretas fueron colocadas sobre una frontera virtual elíptica (figura 6), utilizando un factor de distancia de 1.1 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor mínimo de N utilizado fue 50, el máximo fue $N = 500$ con un tamaño de paso igual a cincuenta. En la figura 7 se muestra la solución exacta y la solución aproximada para dos

valores de N , observándose la mejoría en la aproximación al incrementarse el valor de tal parámetro.

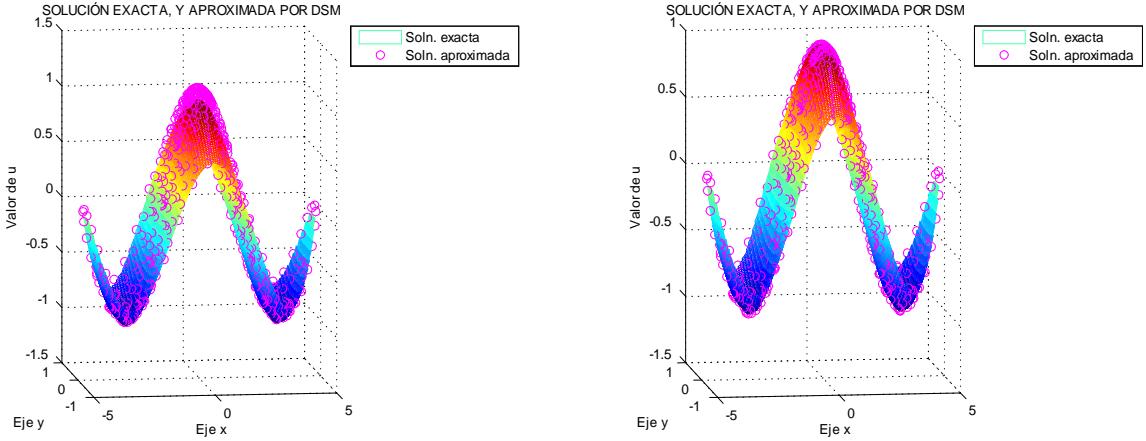


Figura 7: Gráficas de la superficie de la solución exacta y de los valores de la solución aproximada en los puntos de prueba para el ejemplo 1. En la gráfica de la izquierda se utilizaron $N = 50$ términos; en la gráfica de la derecha se utilizaron $N = 150$ términos. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

En la figura 8 se muestra la gráfica de la superficie del error absoluto sobre el dominio Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio, y longitudinalmente a lo largo del eje menor de la elipse.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consultese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 9 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, excepto para los valores más grandes de N , en los que el error se incrementa conforme los puntos de prueba se encuentran más cercanos a la frontera del dominio. De la separación entre las sucesivas gráficas se puede inferir que el error disminuye de forma exponencial, en función de N . Para apoyar esta última inferencia, se consideraron los valores de error absoluto máximo obtenidos sobre las curvas con razón de homotecia igual a 0.1 y los valores correspondientes de N , y se realizó un ajuste de curva exponencial; obteniéndose:

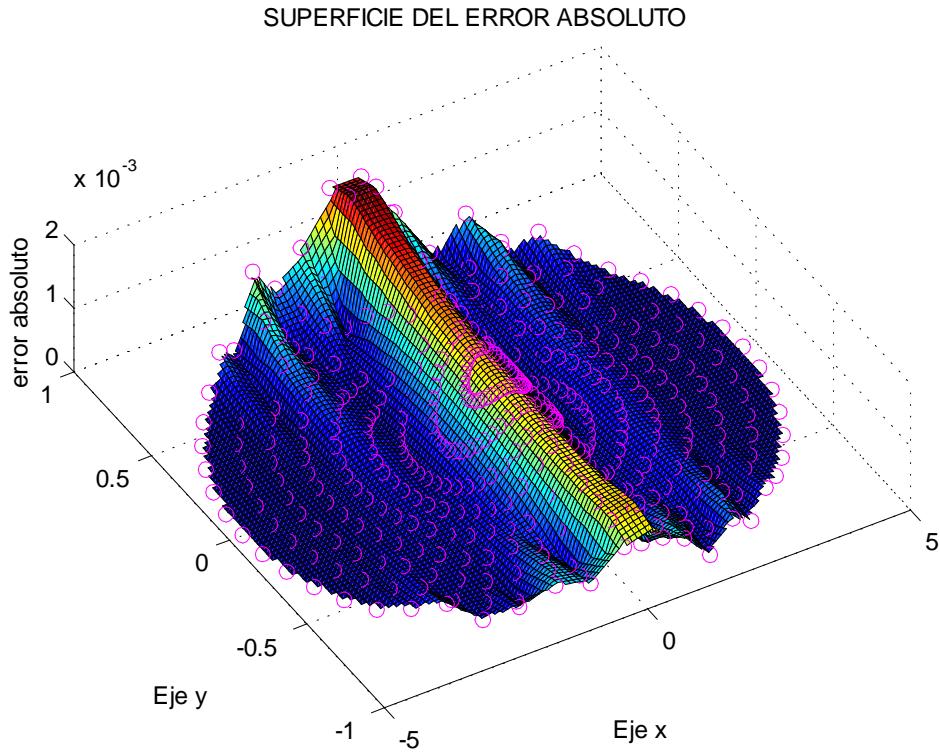


Figura 8: Gráfica de la superficie del error absoluto para el ejemplo 1. $N = 150$ términos. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

$$\text{error abs. máx} = 1.04e^{-0.04264N} \quad (31)$$

con un coeficiente de determinación R cuadrado ajustado con valor 1.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 1.1 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 1100$, donde el orden de magnitud del error absoluto fue del orden de 10^{-14} , el nivel de error ya no disminuye necesariamente.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 2 se presentan

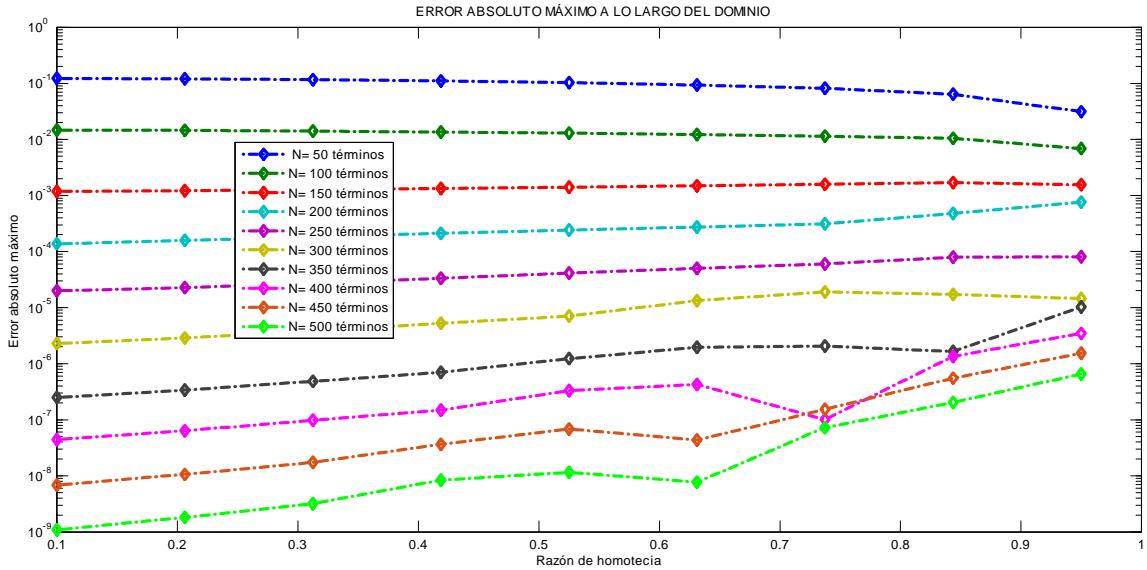


Figura 9: Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 1. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas homotéticas con razón de homotecia 0.1, para diferentes valores de N y de $f.d.$.

Se puede observar que para los valores menores de N (primeros renglones de la tabla), el error absoluto máximo disminuye en al menos dos órdenes de magnitud en cada paso de alejamiento de la frontera virtual con respecto a la frontera real; sin embargo, para los valores mayores de N este efecto ya no es tan pronunciado.

N	Factor de distancia (f. d.)		
	1.1	1.5	1.9
50	0.12338	0.00124	1.16300e-05
100	0.01474	0.00017	6.08688e-06
150	0.00118	1.05905e-06	1.34803e-08
200	0.00013	1.34944e-08	4.67471e-10
250	2.00316e-05	1.42266e-09	4.72117e-12
300	2.29835e-06	2.58314e-11	1.34891e-13
350	2.50937e-07	3.56992e-12	1.68830e-14
400	4.45402e-08	9.83123e-14	3.10231e-14
450	6.86380e-09	1.75839e-14	1.52209e-14
500	1.08823e-09	2.84433e-14	4.85350e-14

Cuadro 2: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 1. Las distancias mínimas entre la frontera real y la frontera real son: 0.1 para f. d.=1.1; 0.5 para f. d.=1.5 y 0.9 para f. d.=1.9.

4.3.2. Ejemplo 2

$$\begin{cases} \Delta u + k^2 u = 0, & k \in \mathbb{R}, (x, y) \in \Omega \\ u(x, y) = (1 - \frac{y}{ik}) e^{iyk} - \frac{\sinh(iyk)}{k^2}, & x \in \{-1, 1\}, -3 < y < 3 \\ u(x, y) = (x^2 - \frac{3}{ik}) e^{3ik} - \frac{\sinh(3ik)}{k^2}, & y = 3, -1 < x < 1 \\ u(x, y) = (x^2 + \frac{3}{ik}) e^{-3ik} - \frac{\sinh(-3ik)}{k^2}, & y = -3, -1 < x < 1. \end{cases}$$

En este ejemplo se considera un dominio rectangular $\Omega = \{(x, y) \in \mathbb{R}^2 : -1 < x < 1, -3 < y < 3\}$, cuya base tiene longitud *dos* y cuya longitud de la altura es *seis*. Este problema se abordará en dos formas, presentadas como incisos 2a) y 2b).

2a) Problema rectangular interno.

Para la validación y el cálculo de los errores absolutos se utilizó la

$$\text{Solución exacta: } u(x, y) = (x^2 - \frac{y}{ik}) e^{iyk} - \frac{\sinh(iyk)}{k^2}$$

con el valor $k = \sqrt{2}$.

En la implementación numérica las fuentes discretas fueron colocadas sobre una frontera virtual circular (figura 10), utilizando un factor de distancia de 3.3523 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor mínimo de N utilizado fue 40, el máximo fue $N = 400$ con un tamaño de paso igual a cuarenta. En la figura 11 se muestra la parte real de la solución exacta y la parte real de la solución aproximada para dos valores de N , observándose la mejoría en la aproximación

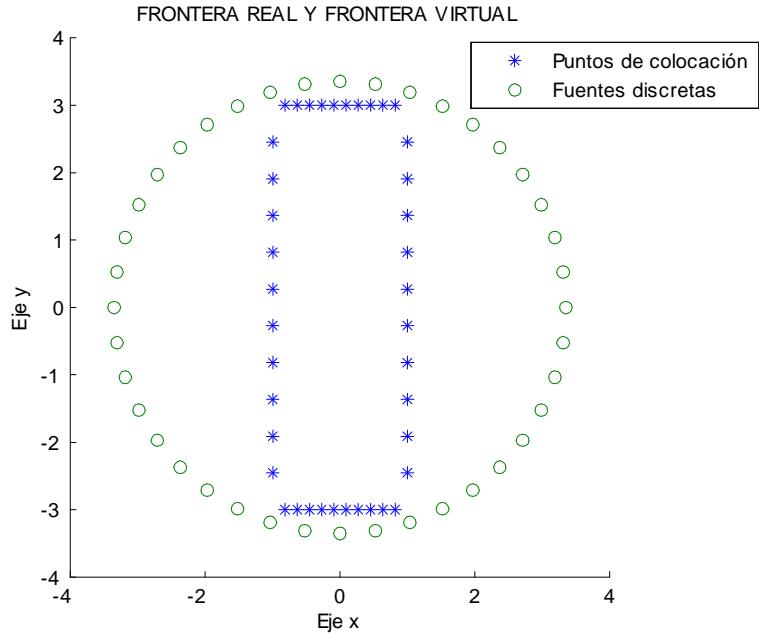


Figura 10: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 2, inciso a. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.

al incrementarse el valor de tal parámetro.

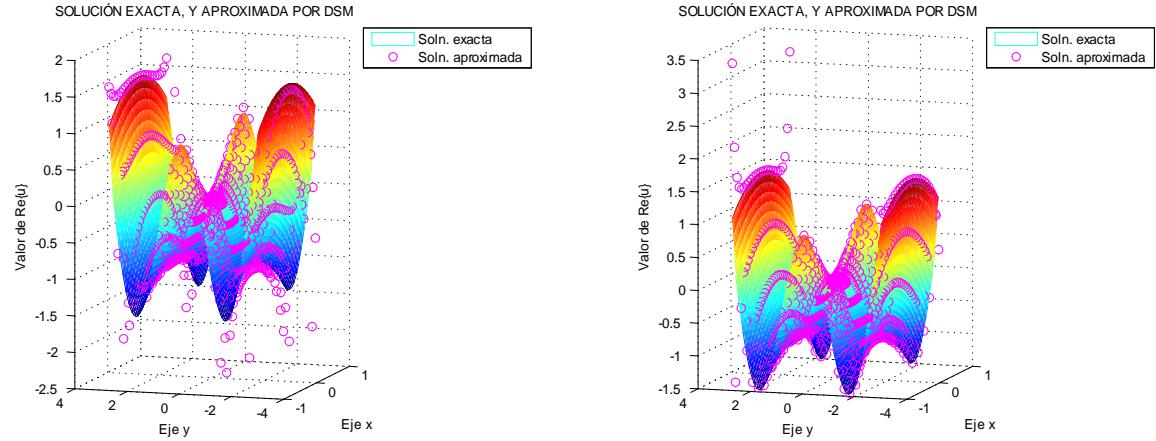


Figura 11: Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 2, inciso a. En la gráfica de la izquierda se utilizaron $N = 40$ términos; en la gráfica de la derecha se utilizaron $N = 120$ términos. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.

En la figura 12 se muestra la gráfica de la superficie del error absoluto sobre el dominio

Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio, especialmente en las cercanías de las esquinas del rectángulo.

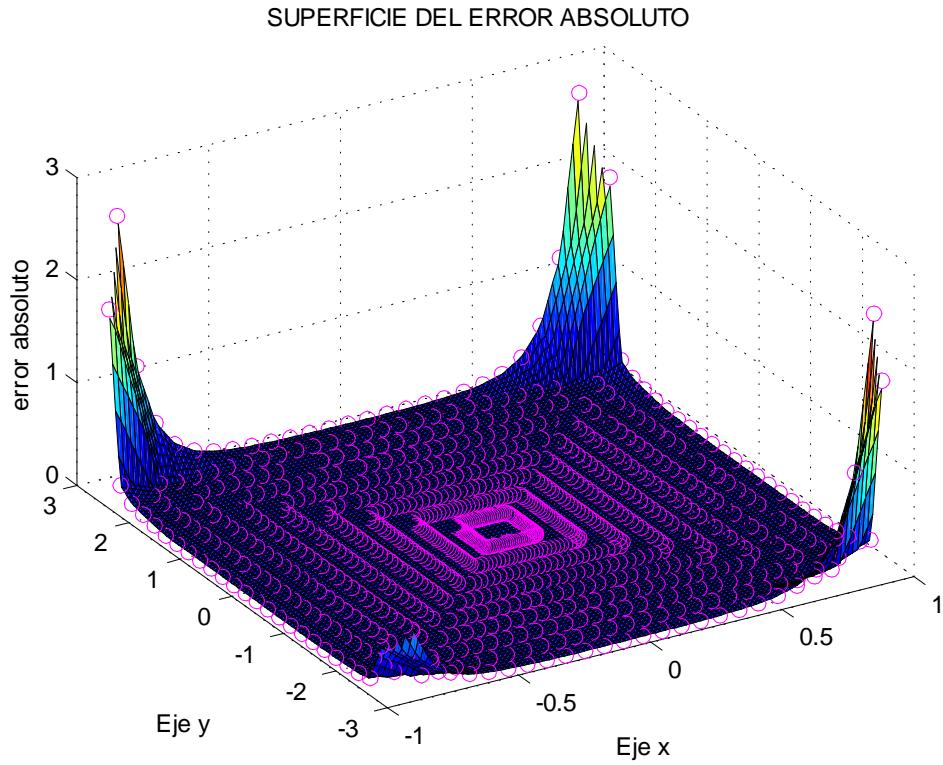


Figura 12: Gráfica de la superficie del error absoluto para el ejemplo 2, inciso a. $N = 120$ términos. $f. d.=3.3523$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consúltese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 13 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, excepto para los puntos de prueba que se encuentran más cercanos a la frontera del dominio, donde el error se incrementa. De la separación entre las sucesivas gráficas no es claro que se pueda inferir que el error disminuye de forma exponencial, en función de N . Para apoyar esta última inferencia, se consideraron los valores de error absoluto máximo obtenidos sobre las

curvas con razón de homotecia igual a 0.1 y los valores correspondientes de N , y se realizó un ajuste de curva exponencial; obteniéndose:

$$\text{error abs. máx} = 0.2129e^{-0.009733N} \quad (32)$$

con un coeficiente de determinación R cuadrado ajustado con valor 0.1848.

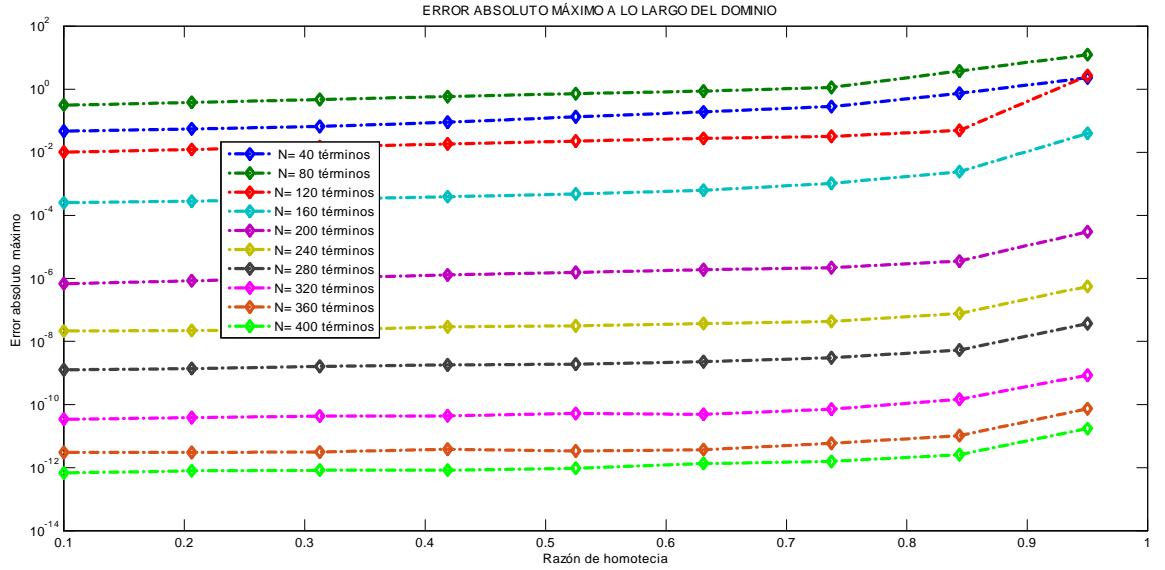


Figura 13: Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 2, inciso a. f. d.=3.3523, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.19.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 3.3523 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 800$, donde el orden de magnitud del error absoluto fue del orden de 10^{-13} , el nivel de error ya no disminuye necesariamente.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 3 se presentan los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas homotéticas con razón de homotecia 0.1, para diferentes valores de N y de *f.d.*

N	Factor de distancia (f. d.)		
	3.3523	4.3523	5.3523
40	0.04706	4.35081e-06	6.33143e-09
80	0.31688	1.55080e-11	5.95803e-14
120	0.01026	2.99306e-14	1.01930e-13
160	0.00025	3.58586e-14	1.03833e-13
200	6.92710e-07	1.45979e-13	7.50559e-14
240	2.16600e-08	5.62743e-14	1.03702e-13
280	1.27711e-09	6.28705e-14	1.03876e-13
320	3.47415e-11	5.83660e-14	3.52594e-13
360	3.05755e-12	1.02956e-13	2.02645e-13
400	6.80675e-13	2.46652e-13	1.16814e-13

Cuadro 3: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 2, inciso a. Las distancias mínimas entre la frontera real y la frontera real son: 0.19 para f. d.=3.3523; 1.19 para f. d.=4.3523 y 2.19 para f. d.=5.3523.

Se puede observar que sólo para los valores menores de N (primeros dos renglones de la tabla), el error absoluto máximo disminuye en al menos tres órdenes de magnitud en cada paso de alejamiento de la frontera virtual con respecto a la frontera real; sin embargo, para los valores mayores de N este efecto ya no se presenta, incluso el error se puede incrementar.

2b) Problema rectangular interno.

Para la validación y el cálculo de los errores absolutos se utilizó la

$$\text{Solución exacta: } u(x, y) = (x^2 - \frac{y}{ik})e^{iyk} - \frac{\sinh(iyk)}{k^2}$$

con el valor $k = \sqrt{2}$.

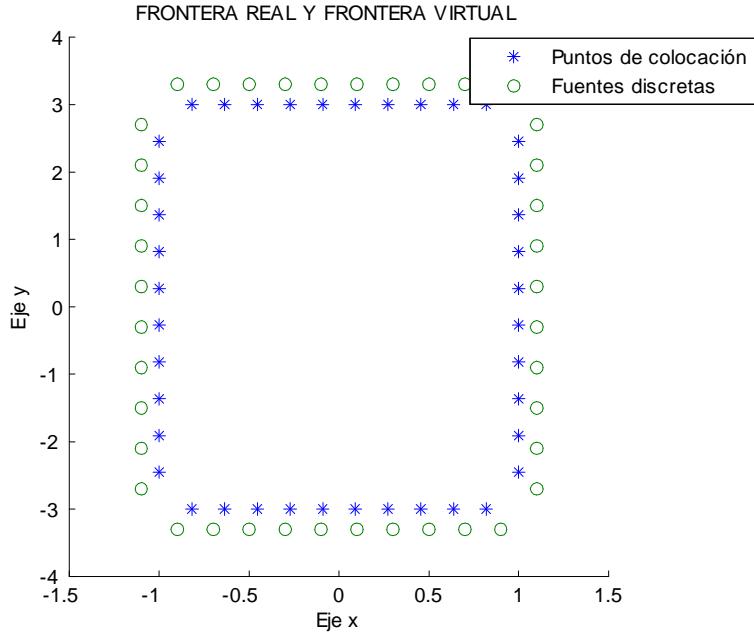


Figura 14: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 2, inciso b. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

En la implementación numérica las fuentes discretas fueron colocadas, en este caso, sobre una frontera virtual rectangular (figura 14). Nótese que se trata del mismo problema presentado en 2a), con la diferencia de que la frontera virtual es distinta, esto se hizo con el propósito de investigar si la el grado de suavidad de la misma influye en el nivel de error obtenido. Se utilizó un factor de distancia de 1.1 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor mínimo de N utilizado fue 40, el máximo fue $N = 400$ con un tamaño de paso igual a cuarenta. En la figura 15 se muestra la parte real de la solución exacta y la parte real de la solución aproximada para dos valores de N , observándose la mejoría en la aproximación al incrementarse el valor de tal parámetro.

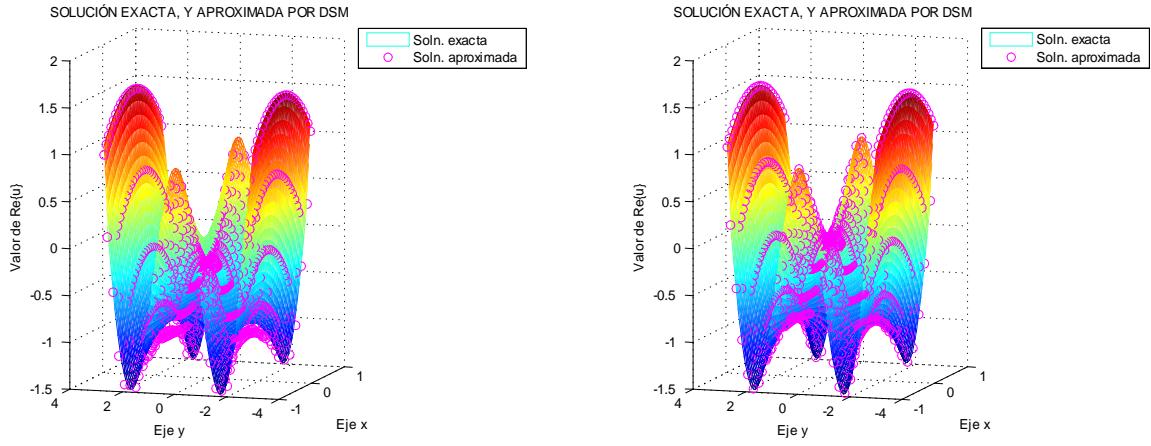


Figura 15: Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 2, inciso b. En la gráfica de la izquierda se utilizaron $N = 40$ términos; en la gráfica de la derecha se utilizaron $N = 120$ términos. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

En la figura 16 se muestra la gráfica de la superficie del error absoluto sobre el dominio Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio; sin embargo, en comparación con lo observado en el inciso a, el error ya no se encuentra tan localizado en las cercanías de las esquinas del rectángulo, y el nivel de error general es menor.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consúltese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 17 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, para los primeros valores de N , sin embargo, para los valores mayores de dicho parámetro, se observa que el error comienza a incrementarse desde los puntos de prueba cercanos al centroide del dominio, alcanzando sus máximos en la cercanía de la frontera del mismo. De la separación entre las sucesivas gráficas no es claro que se pueda inferir que el error disminuye de forma exponencial, en función de N . Sin embargo, al considerarse los valores de error absoluto máximo obtenidos sobre las curvas con razón de homotecia igual a 0.1 y

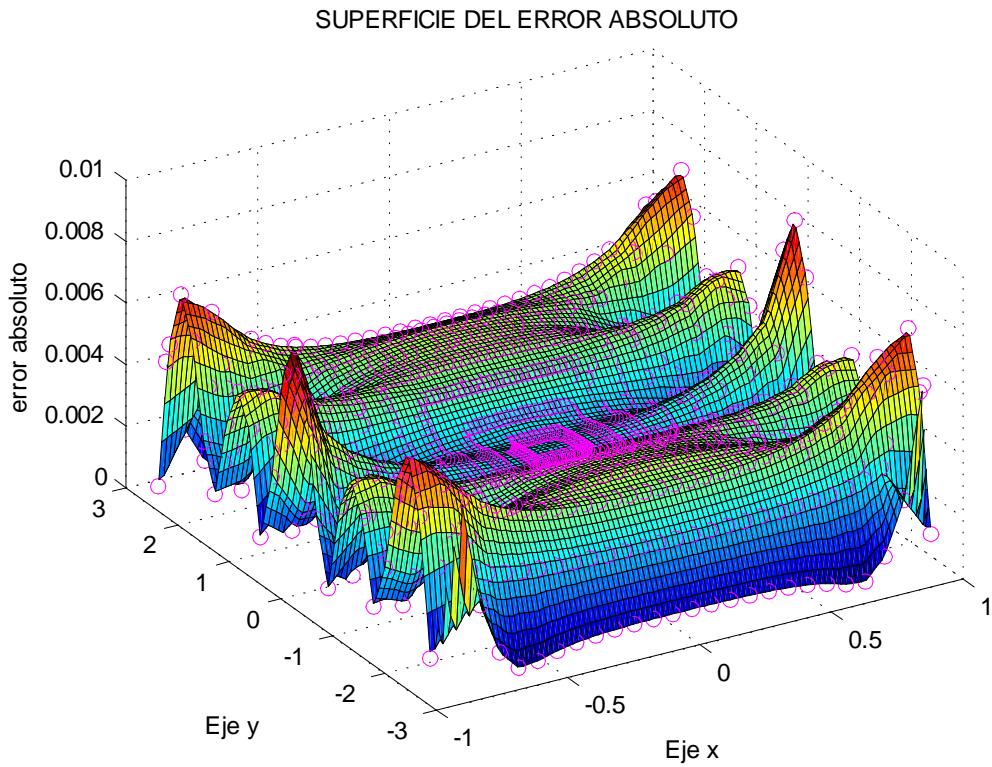


Figura 16: Gráfica de la superficie del error absoluto para el ejemplo 2, inciso b. $N = 120$ términos. $f. d.=1.1$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

los valores correspondientes de N , y realizar un ajuste de curva exponencial; se obtuvo:

$$\text{error abs. máx} = 1.5e^{-0.04537N} \quad (33)$$

con un coeficiente de determinación R cuadrado ajustado con valor 0.9997.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 1.1 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 900$, donde el orden de magnitud del error absoluto fue del orden de 10^{-13} , el nivel de error ya no disminuye necesariamente.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento

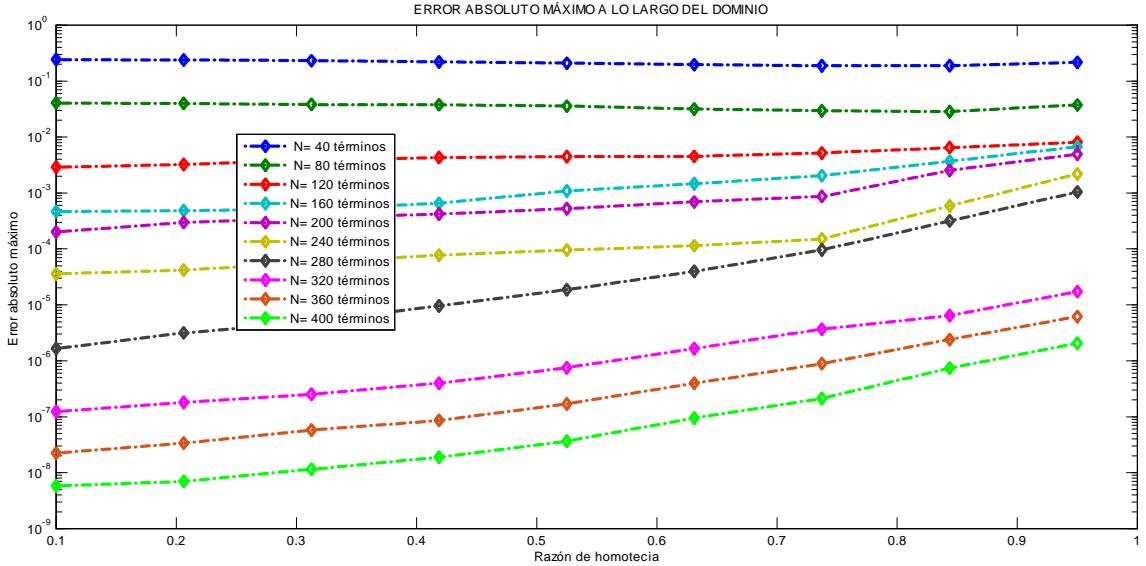


Figura 17: Gráficas del error absoluto máximo, para diversos valores de N, para el ejemplo 2, inciso b. f. d.=1.1, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.1.

de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 4 se presentan los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas homotéticas con razón de homotecia 0.1, para diferentes valores de N y de $f.d.$.

Se puede observar que para los valores menores de N (primeros renglones de la tabla), el error absoluto máximo disminuye en dos o tres órdenes de magnitud en cada paso de alejamiento de la frontera virtual con respecto a la frontera real; sin embargo, para los valores mayores de N este efecto ya no se presenta, incluso el error se puede incrementar.

N	Factor de distancia (f. d.)		
	1.1	1.5	1.9
40	0.24414	0.00041	0.00041
80	0.04098	2.01444e-05	4.45078e-08
120	0.00289	3.25008e-07	9.73447e-09
160	0.00046	1.75616e-07	9.09901e-10
200	0.00020	3.18215e-09	1.70214e-11
240	3.62004e-05	1.65674e-10	4.54293e-13
280	1.66221e-06	1.88398e-11	2.16640e-13
320	1.25279e-07	9.28682e-13	8.73565e-14
360	2.24616e-08	4.14129e-14	1.06884e-13
400	5.83601e-09	2.93551e-14	1.86831e-13

Cuadro 4: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 2, inciso b. Las distancias mínimas entre la frontera real y la frontera real son: 0.1 para f. d.=1.1; 0.5 para f. d.=1.5 y 0.9 para f. d.=1.9.

4.4. Problemas externos

4.4.1. Ejemplo 3

$$\begin{cases} \Delta u + k^2 u = 0, & k \in \mathbb{R}, (x, y) \in \Omega \\ u(x, y) = H_0^{(1)}(k\sqrt{x^2 + y^2}), & (x, y) \in \partial\Omega \\ \frac{\partial u}{\partial r} - iku = o\left(\frac{1}{\sqrt{r}}\right), \text{ cuando } r \rightarrow +\infty. \end{cases}$$

Donde $H_0^{(1)}$ representa la función de Hankel del primer tipo, de orden cero. La última de las condiciones anteriores es la *condición de radiación de Sommerfeld*, donde $r^2 = x^2 + y^2$, $x = r \cos \theta$, $y = r \sin \theta$ y o representa la notación asintótica *o pequeña*.

En este ejemplo se considera un dominio $\Omega = \{(x, y) \in \mathbb{R}^2 : \frac{x^2}{25} + y^2 > 1\}$, cuyo complemento es una elipse con semieje horizontal de longitud *cinco* y con longitud del semieje vertical igual a *uno*.

Para la validación y el cálculo de los errores absolutos se utilizó la

Solución exacta: $u(x, y) = H_0^{(1)}(k\sqrt{x^2 + y^2})$

con el valor $k = \sqrt{2}$.

En la implementación numérica las fuentes discretas fueron colocadas sobre una frontera virtual elíptica (figura 18), utilizando un factor de distancia de 0.8 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor mínimo de N utilizado fue 10, el máximo fue $N = 100$ con un tamaño de paso igual a

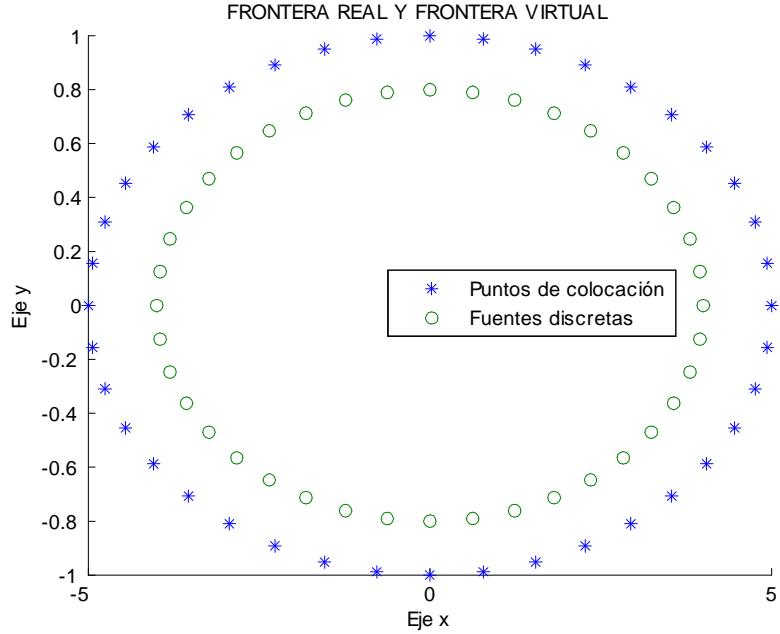


Figura 18: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 3. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

diez. En la figura 19 se muestra la parte real de la solución exacta y la parte real de la solución aproximada para dos valores de N , observándose la mejoría en la aproximación al incrementarse el valor de tal parámetro.

En la figura 20 se muestra la gráfica de la superficie del error absoluto sobre el dominio Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio; principalmente en la cercanía de las extensiones longitudinales del eje menor de la elipse que es complemento del dominio de estudio.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consúltese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 21 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, excepto para los valores más grandes de N , en los que el error aumenta conforme los puntos de prueba

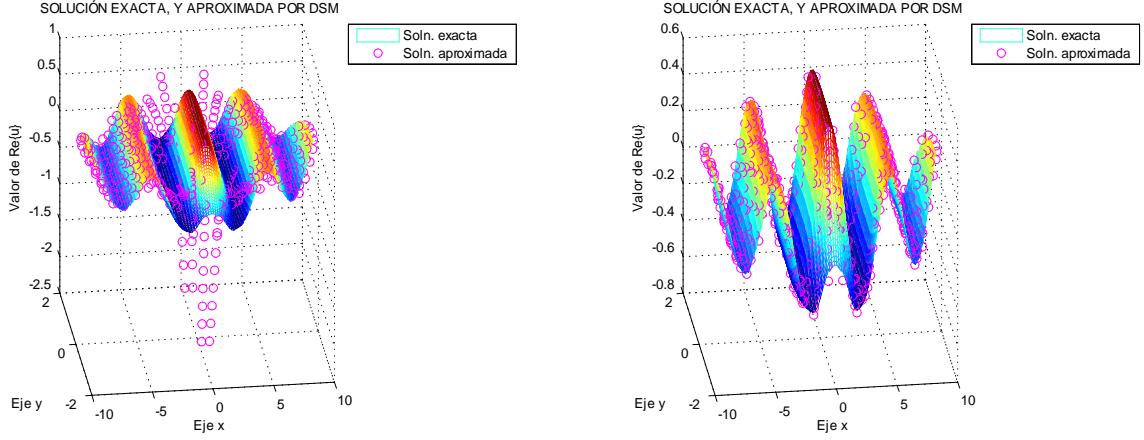


Figura 19: Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 3. En la gráfica de la izquierda se utilizaron $N = 10$ términos; en la gráfica de la derecha se utilizaron $N = 30$ términos. $f. d.=0.8$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

se encuentran más cercanos a la frontera del dominio. De la separación entre las sucesivas gráficas no es claro que se pueda inferir que el error disminuya de forma exponencial, en función de N . Sin embargo, al considerarse los valores de error absoluto máximo obtenidos sobre las curvas con razón de homotecia igual a 1.9 y los valores correspondientes de N , y realizar un ajuste de curva exponencial; se obtuvo:

$$\text{error abs. máx} = 35.48e^{-0.2753N} \quad (34)$$

con un coeficiente de determinación R cuadrado ajustado con valor 0.9998.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 0.8 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 110$, donde el orden de magnitud del error absoluto fue del orden de 10^{-4} , el nivel de error ya no disminuye necesariamente.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 5 se presentan los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas

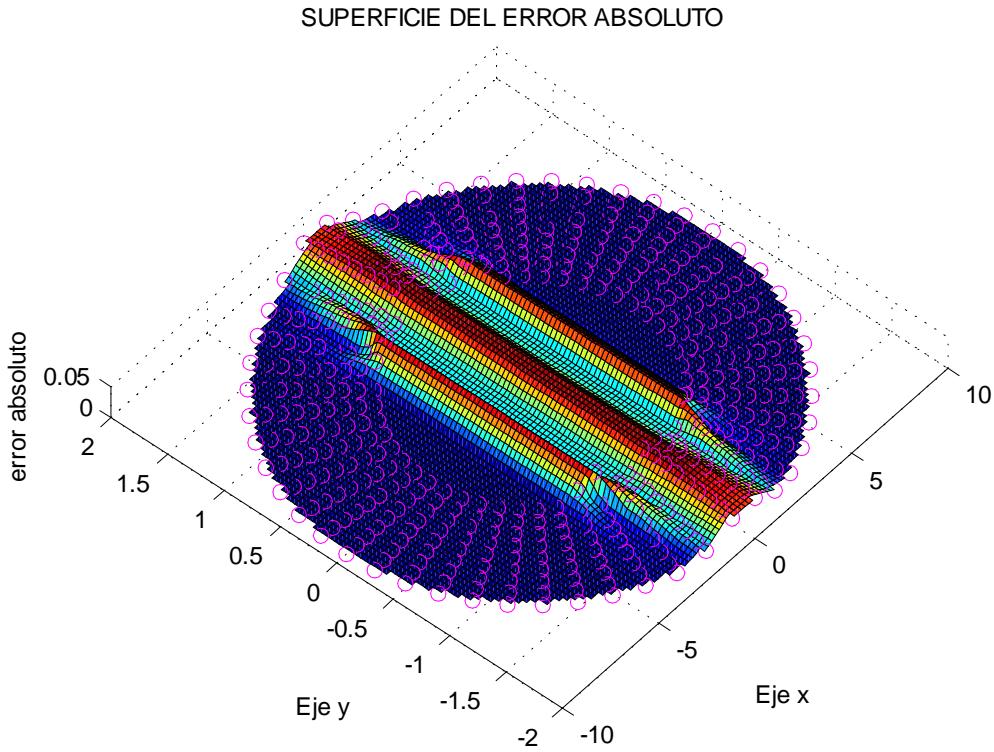


Figura 20: Gráfica de la superficie del error absoluto para el ejemplo 3. $N = 30$ términos. $f.d.=0.8$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

homotéticas con razón de homotecia 1.9, para diferentes valores de N y de $f.d.$.

Se puede observar que para los valores menores de N (primeros renglones de la tabla), el error absoluto máximo disminuye en dos o tres órdenes de magnitud en cada paso de alejamiento de la frontera virtual con respecto a la frontera real; para los valores mayores de N este efecto se presenta incluso de manera más pronunciada, dado que se observaron mejoras en el nivel de error de aproximadamente seis órdenes de magnitud.

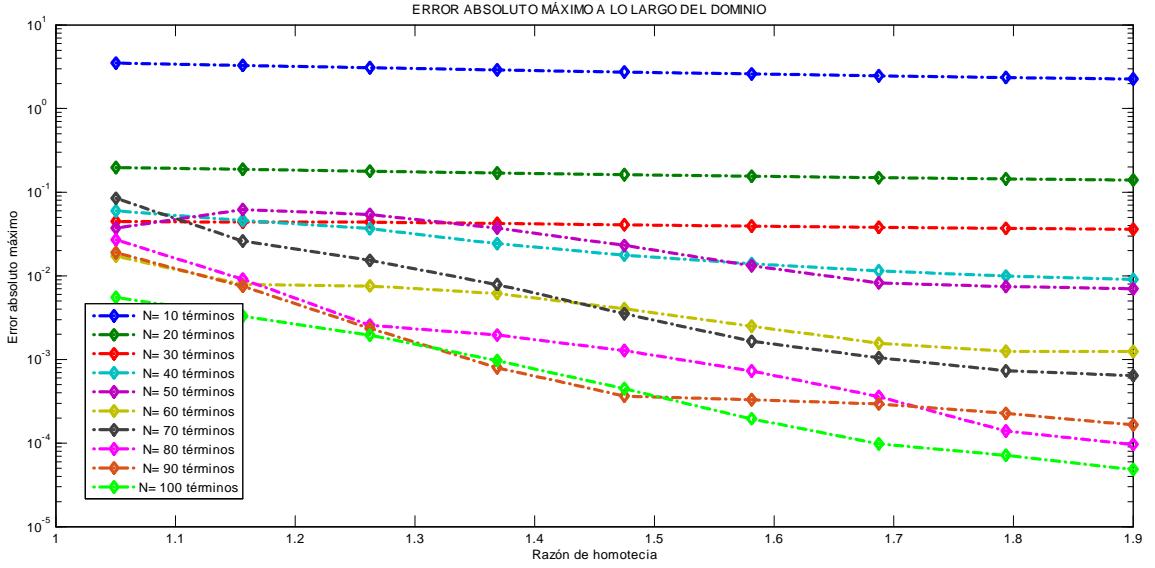


Figura 21: Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 3. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

4.4.2. Ejemplo 4

$$\left\{ \begin{array}{ll} \Delta u + k^2 u = 0, & k \in \mathbb{R}, (x, y) \in \Omega \\ u(x, y) = H_0^{(1)}(k\sqrt{1+y^2}) + H_0^{(1)}(k\sqrt{1.21+y^2}), & x = -1, -3 < y < 3 \\ u(x, y) = H_0^{(1)}(k\sqrt{1+y^2}) + H_0^{(1)}(k\sqrt{0.81+y^2}), & x = 1, -3 < y < 3 \\ u(x, y) = H_0^{(1)}(k\sqrt{x^2+9}) + H_0^{(1)}(k\sqrt{(x-0.1)^2+9}), & y \in \{-3, 3\}, -1 < x < 1 \\ \frac{\partial u}{\partial r} - iku = o\left(\frac{1}{\sqrt{r}}\right), \text{ cuando } r \rightarrow +\infty. \end{array} \right.$$

Donde $H_0^{(1)}$ representa la función de Hankel del primer tipo, de orden cero. La última de las condiciones anteriores es la *condición de radiación de Sommerfeld*, donde $r^2 = x^2 + y^2$, $x = r \cos \theta$, $y = r \sin \theta$ y o representa la notación asintótica *o pequeña*.

En este ejemplo se considera un dominio $\Omega = \{(x, y) \in \mathbb{R}^2 : -1 \leq x \leq 1, -3 \leq y \leq 3\}^c$, cuyo complemento es un rectángulo con base de longitud *dos* y con longitud de la altura igual a *seis*. Este problema se abordará en dos formas, presentadas como incisos 4a) y 4b).

4a) Rectangular externo.

Para la validación y el cálculo de los errores absolutos se utilizó la

$$\text{Solución exacta: } u(x, y) = H_0^{(1)}(k\sqrt{x^2+y^2}) + H_0^{(1)}(k\sqrt{(x-0.1)^2+y^2})$$

N	Factor de distancia (f. d.)		
	0.8	0.5	0.2
10	2.26123	0.07386	0.00059
20	0.14052	0.00292	6.54538e-08
30	0.03617	0.00021	1.17935e-09
40	0.00912	1.80279e-06	5.03366e-10
50	0.00699	1.21970e-06	3.38675e-10
60	0.00124	1.24330e-06	8.19874e-10
70	0.00064	2.86809e-05	4.13493e-11
80	9.67414e-05	4.73607e-05	6.56613e-12
90	0.00016	1.05272e-05	3.41528e-11
100	4.84981e-05	2.32953e-06	9.73658e-12

Cuadro 5: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 3. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.

con el valor $k = \sqrt{2}$.

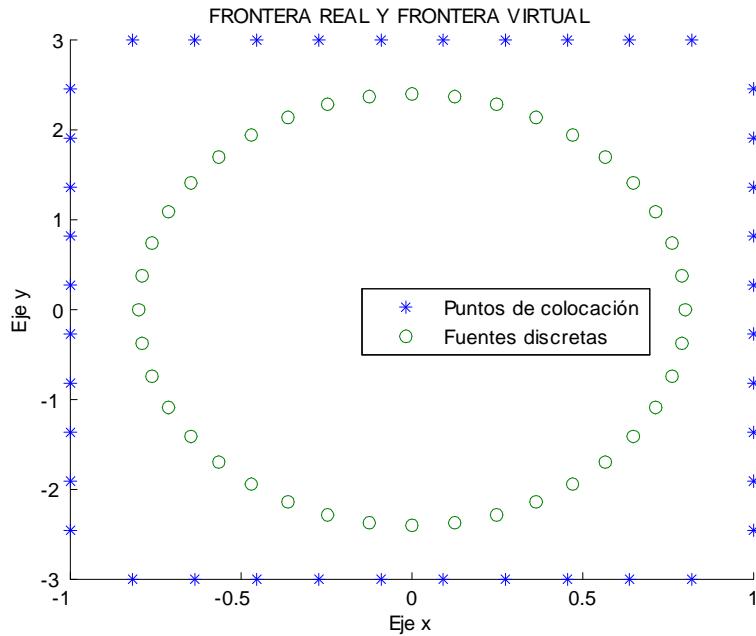


Figura 22: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 4, inciso a. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

En la implementación numérica las fuentes discretas fueron colocadas sobre una frontera virtual elíptica (figura 22), utilizando un factor de distancia de 0.8 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor

mínimo de N utilizado fue 12, el máximo fue $N = 84$ con un tamaño de paso igual a ocho. En la figura 23 se muestra la parte real de la solución exacta y la parte real de la solución aproximada para dos valores de N , observándose la mejoría en la aproximación al incrementarse el valor de tal parámetro.

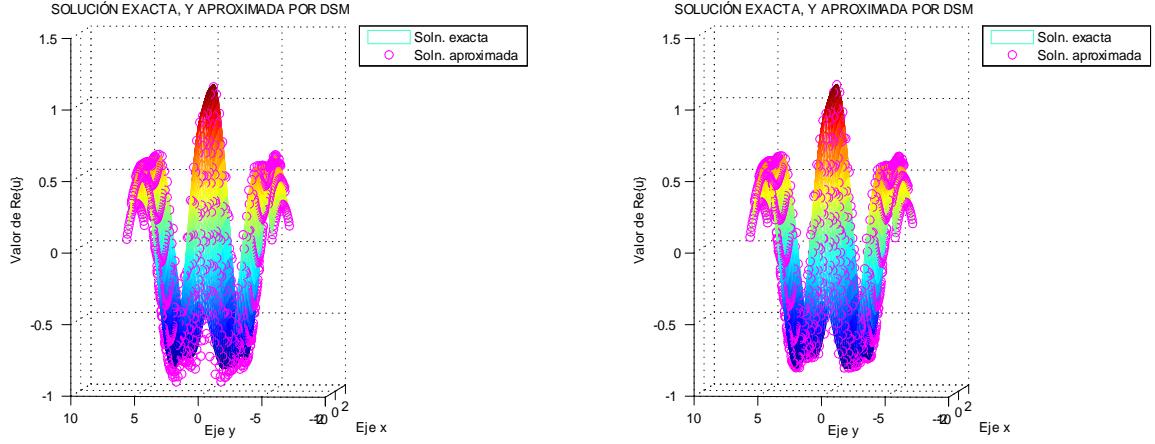


Figura 23: Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 4, inciso a. En la gráfica de la izquierda se utilizaron $N = 12$ términos; en la gráfica de la derecha se utilizaron $N = 28$ términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

En la figura 24 se muestra la gráfica de la superficie del error absoluto sobre el dominio Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio; principalmente en la cercanía de las extensiones longitudinales del eje de simetría paralelo al eje x del rectángulo que es complemento del dominio en estudio.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consúltese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 25 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, excepto para los valores más grandes de N , en los que el error aumenta conforme los puntos de prueba se encuentran más cercanos a la frontera del dominio. De la separación entre las sucesivas gráficas no es claro que se pueda inferir que el error disminuya de forma exponencial, en

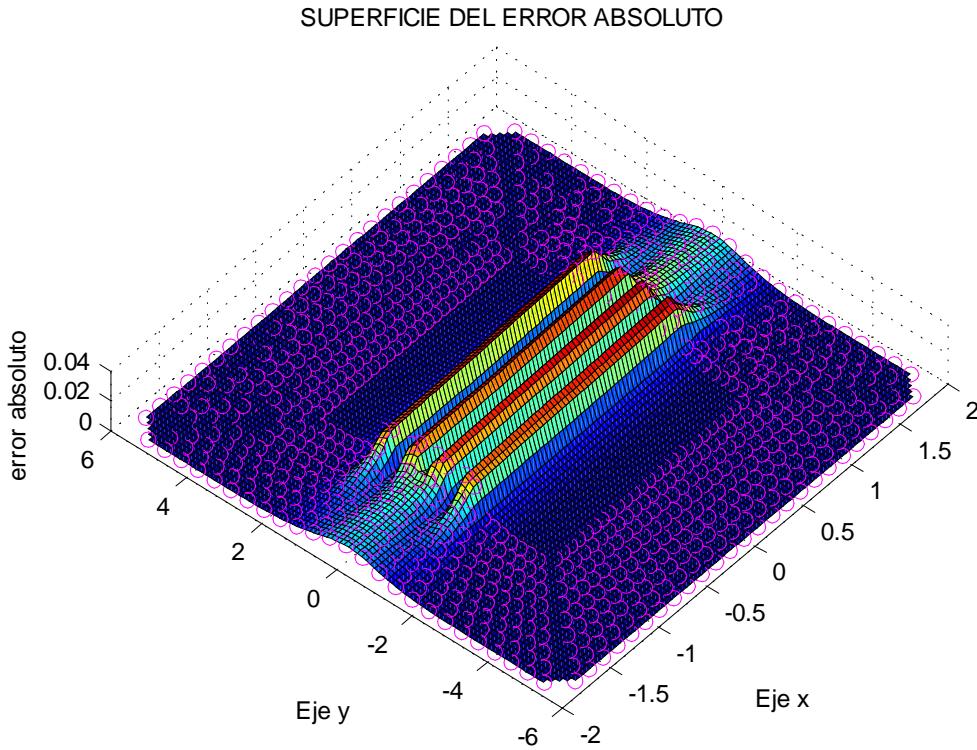


Figura 24: Gráfica de la superficie del error absoluto para el ejemplo 4, inciso a. $N = 28$ términos. $f. d.=0.8$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

función de N . Sin embargo, al considerarse los valores de error absoluto máximo obtenidos sobre las curvas con razón de homotecia igual a 1.9 y los valores correspondientes de N , y realizar un ajuste de curva exponencial; se obtuvo:

$$\text{error abs. máx} = 2.624e^{-0.183N} \quad (35)$$

con un coeficiente de determinación R cuadrado ajustado con valor 0.9999.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 0.8 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 96$, donde el orden de magnitud del error absoluto fue del orden de 10^{-5} , el nivel de error ya no disminuye necesariamente.

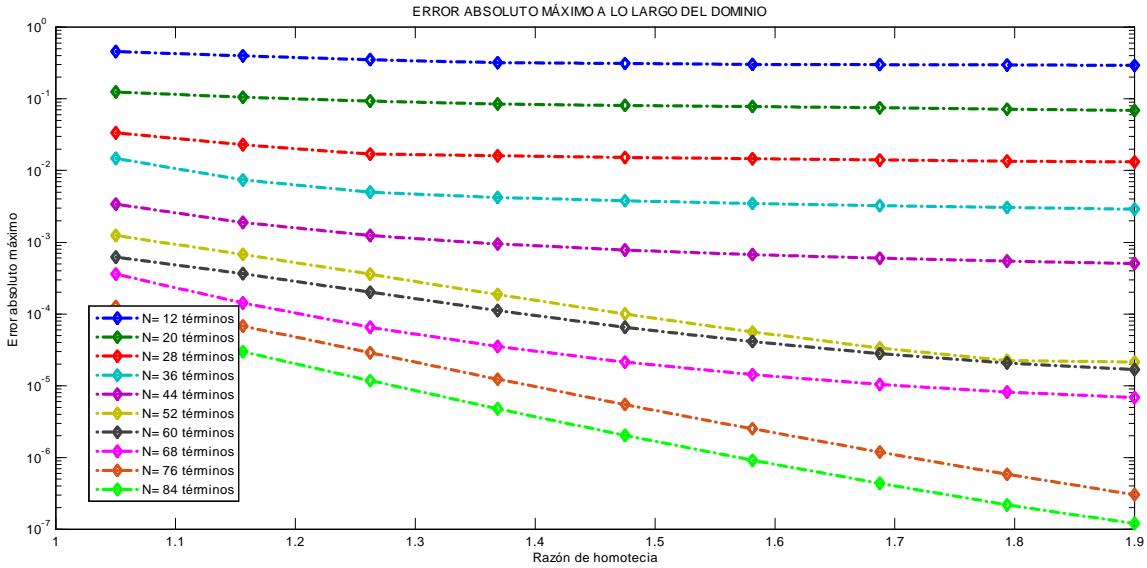


Figura 25: Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 4, inciso a. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 6 se presentan los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas homotéticas con razón de homotecia 1.9, para diferentes valores de N y de *f.d.*.

Se puede observar que para los valores menores de N (primeros renglones de la tabla), el error absoluto máximo disminuye en dos o tres órdenes de magnitud en cada paso de alejamiento de la frontera virtual con respecto a la frontera real; para los valores mayores de N este efecto se presenta incluso de manera más pronunciada, dado que se observaron mejoras en el nivel de error de aproximadamente cinco órdenes de magnitud.

N	Factor de distancia (f. d.)		
	0.8	0.5	0.2
12	0.29183	0.00662	6.30035e-06
20	0.06882	0.00011	1.76001e-08
28	0.01318	1.50608e-06	6.24924e-11
36	0.00289	6.50408e-08	5.36159e-11
44	0.00050	2.04515e-09	3.57656e-11
52	2.14768e-05	7.63688e-09	1.99602e-12
60	1.68870e-05	1.88247e-07	1.32924e-12
68	6.86860e-06	3.87598e-07	1.66803e-12
76	3.04081e-07	5.87654e-08	8.75085e-13
84	1.20256e-07	7.31551e-08	9.81476e-13

Cuadro 6: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 4, inciso a. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.

4b) Rectangular externo.

Para la validación y el cálculo de los errores absolutos se utilizó la

$$\text{Solución exacta: } u(x, y) = H_0^{(1)}(k\sqrt{x^2 + y^2}) + H_0^{(1)}(k\sqrt{(x - 0.1)^2 + y^2})$$

con el valor $k = \sqrt{2}$.

En la implementación numérica las fuentes discretas fueron colocadas, en este caso, sobre una frontera virtual rectangular (figura 26). Nótese que se trata del mismo problema presentado en 4a), con la diferencia de que la frontera virtual es distinta, esto se hizo con el propósito de investigar si la el grado de suavidad de la misma influye en el nivel de error obtenido. Se utilizó un factor de distancia de 0.8 para la frontera virtual (para información del parámetro factor de distancia, consúltese la sección 4.1). El valor mínimo de N utilizado fue 8, el máximo fue $N = 80$ con un tamaño de paso igual a ocho. En la figura 27 se muestra la parte real de la solución exacta y la parte real de la solución aproximada para dos valores de N , observándose la mejoría en la aproximación al incrementarse el valor de tal parámetro.

En la figura 28 se muestra la gráfica de la superficie del error absoluto sobre el dominio Ω . Se puede observar que las regiones donde el error es mayor son aquéllas más cercanas a la frontera del dominio; principalmente en la cercanía de las extensiones longitudinales del eje de simetría paralelo al eje x del rectángulo que es complemento del dominio en estudio. En comparación a lo observado en el inciso a, el comportamiento cualitativo del error absoluto

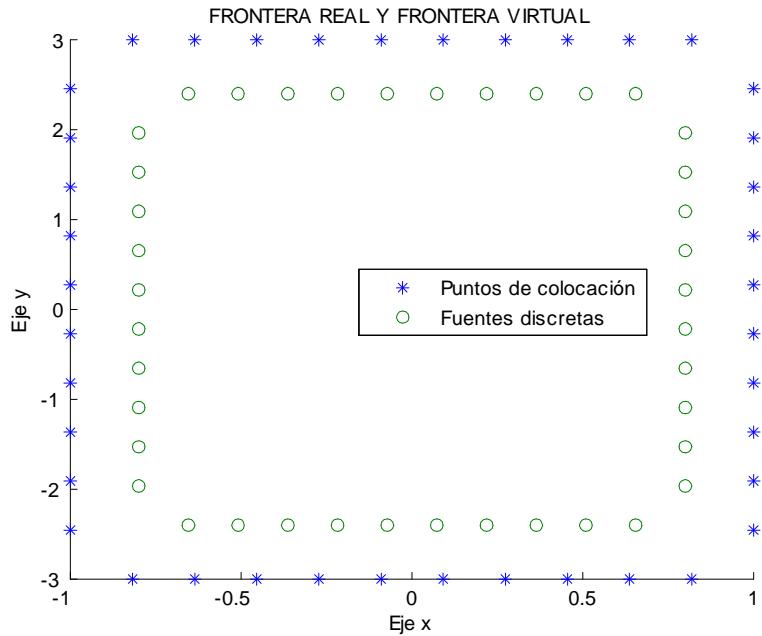


Figura 26: Esquema de la distribución de los puntos de colocación y de las fuentes discretas para el ejemplo 4, inciso b. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

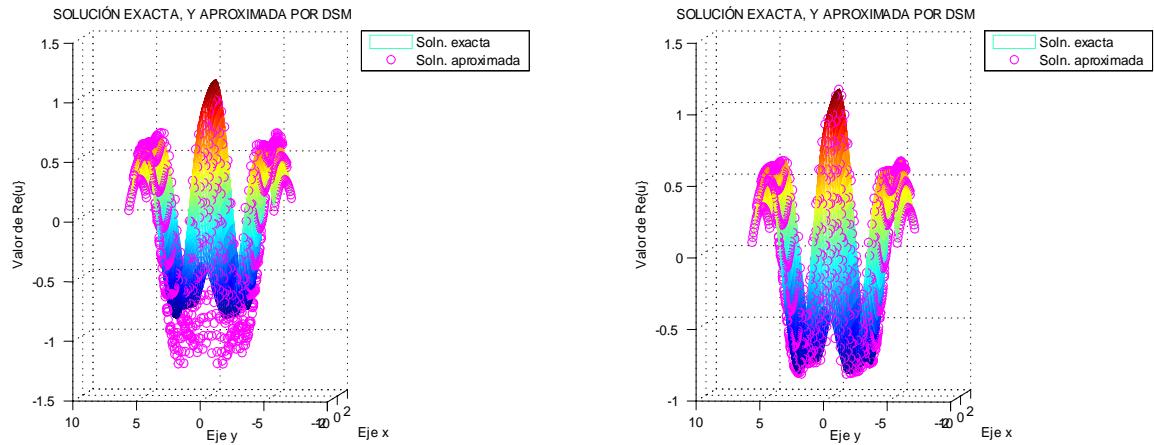


Figura 27: Gráficas de la superficie de la parte real de la solución exacta y de los valores de la parte real de la solución aproximada en los puntos de prueba para el ejemplo 4, inciso b. En la gráfica de la izquierda se utilizaron $N = 8$ términos; en la gráfica de la derecha se utilizaron $N = 24$ términos. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

es muy similar, salvo que en este caso se dio un incremento en la magnitud general del error.

Para un estudio más detallado del error se eligieron puntos de prueba sobre curvas

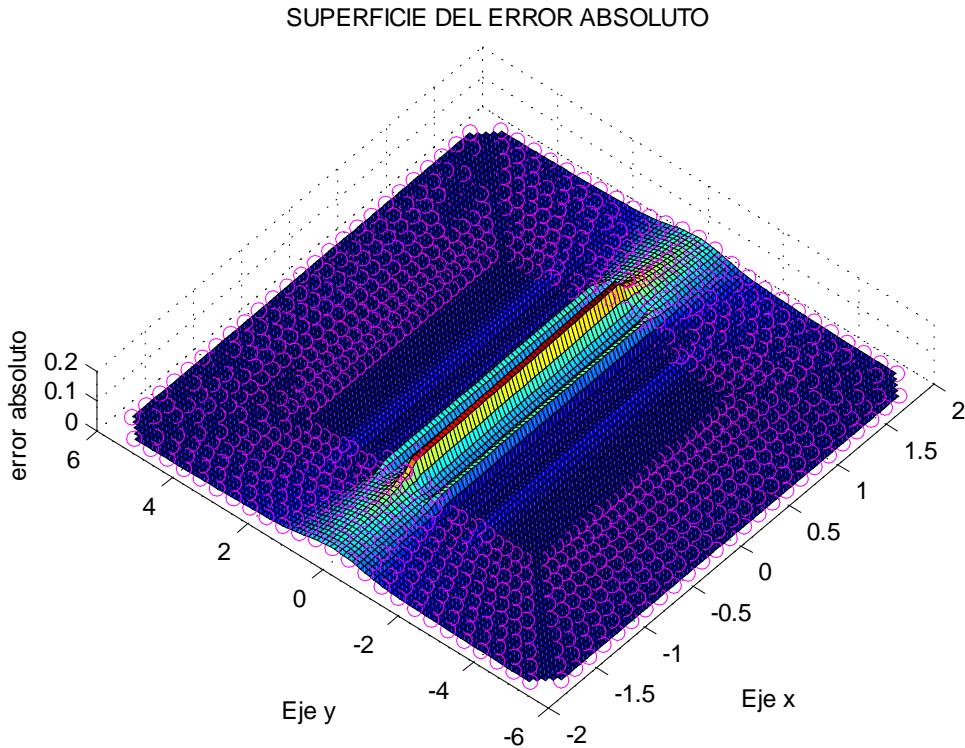


Figura 28: Gráfica de la superficie del error absoluto para el ejemplo 4, inciso b. $N = 24$ términos. $f. d.=0.8$, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

homotéticas a la frontera del dominio (para información acerca de los puntos de prueba, consultese la sección 4.2). Se calculó el error absoluto máximo obtenido en cada una de dichas curvas. En la figura 29 se puede observar el comportamiento del error absoluto máximo a lo largo del interior del dominio, para varios valores de N . Se puede observar que el error absoluto máximo mantiene un comportamiento estable a lo largo del dominio, excepto para los valores más grandes de N , en los que el error se incrementa conforme los puntos de prueba se encuentran más cercanos a la frontera del dominio. De la separación entre las sucesivas gráficas se puede inferir que el error disminuye de forma exponencial, en función de N . Para apoyar esta última inferencia, se consideraron los valores de error absoluto máximo obtenidos sobre las curvas con razón de homotecia igual a 1.9 y los valores correspondientes de N , y se realizó un ajuste de curva exponencial; obteniéndose:

$$\text{error abs. máx} = 3.961e^{-0.1952N} \quad (36)$$

con un coeficiente de determinación R cuadrado ajustado con valor 0.9991.

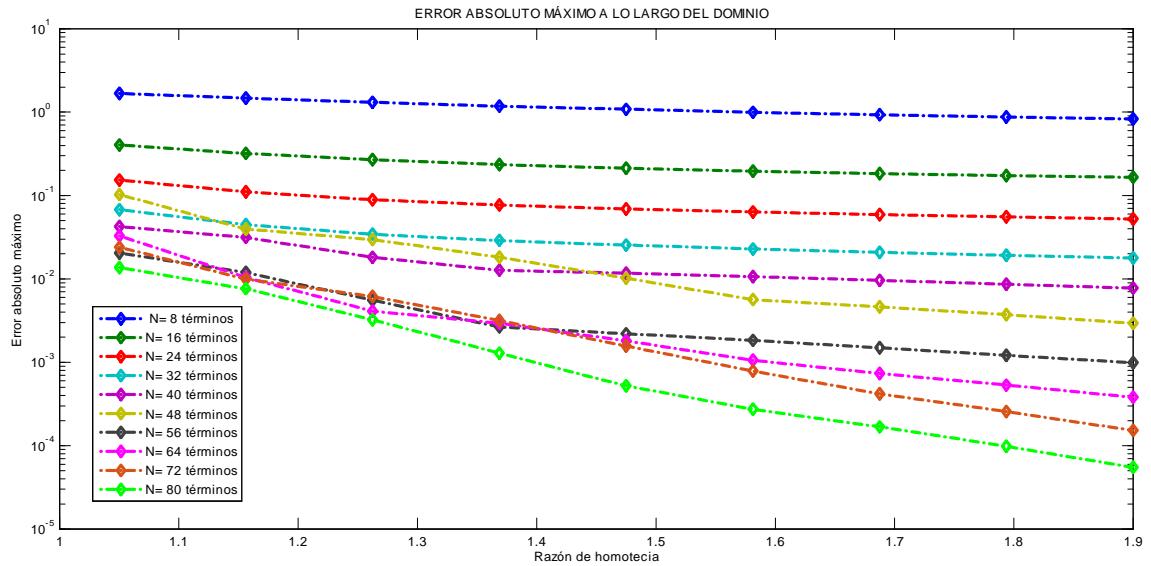


Figura 29: Gráficas del error absoluto máximo, para diversos valores de N , para el ejemplo 4, inciso b. f. d.=0.8, lo que implica una distancia mínima entre la frontera real y la frontera virtual igual a 0.2.

Posteriormente, manteniendo el factor de distancia de alejamiento de la frontera virtual en 0.8 se experimentó aumentando los valores de N , con el fin de observar hasta qué punto se mantiene la disminución del error. Se observó que para valores mayores de $N = 88$, donde el orden de magnitud del error absoluto fue del orden de 10^{-5} , el nivel de error ya no disminuye necesariamente.

Finalmente se exploró el efecto que se presenta el variar el parámetro *factor de distancia* (*f. d.*) que determina el alejamiento de la frontera virtual, y en consecuencia el alejamiento de las fuentes discretas, respecto a la frontera del dominio. En el cuadro 7 se presentan los valores de error absoluto máximo en los puntos de prueba ubicados sobre las curvas homotéticas con razón de homotecia 1.9, para diferentes valores de N y de *f.d.*

Se puede observar que para los valores menores de N (primeros renglones de la tabla), el error absoluto máximo disminuye en uno, dos o tres órdenes de magnitud en cada paso de

N	Factor de distancia (f. d.)		
	0.8	0.5	0.2
8	0.83213	0.07046	0.00234
16	0.16611	0.00758	3.38195e-06
24	0.05253	0.00055	1.21608e-08
32	0.01785	7.50192e-05	3.79579e-11
40	0.00779	7.21131e-06	9.18203e-11
48	0.00295	1.11294e-06	4.83145e-11
56	0.00099	8.08728e-07	4.15711e-11
64	0.00038	2.58229e-07	1.35575e-11
72	0.00015	1.15276e-07	2.85507e-12
80	5.51124e-05	1.60432e-07	8.75242e-12

Cuadro 7: Errores absolutos máximos para diferentes valores de N y de f. d. para el ejemplo 4, inciso b. Las distancias mínimas entre la frontera real y la frontera real son: 0.2 para f. d.=0.8; 0.5 para f. d.=0.5 y 0.8 para f. d.=0.2.

alejamiento de la frontera virtual con respecto a la frontera real; para los valores mayores de N este efecto se presenta incluso de manera más pronunciada, dado que se observaron mejoras en el nivel de error de aproximadamente cinco órdenes de magnitud.

Capítulo V

CONCLUSIONES

A partir de los resultados obtenidos en los experimentos numéricos que se presentan en el capítulo de *ejemplos de prueba*, se puede concluir lo siguiente:

1. El error disminuye al incrementar el valor de N . También disminuye al alejar la frontera virtual de la frontera real, confirmándose los resultados de la literatura.
2. El error presenta un comportamiento más estable (se observa menor variabilidad en sus valores) conforme los puntos de prueba están más alejados de la frontera del dominio Ω .
3. Se observa que el error tiende a propagarse a lo largo de los ejes de menor longitud de los objetos dispersores, o bien de las extensiones de dichos ejes. En los dominios rectangulares el error muestra valores mayores en los puntos de prueba cercanos a las esquinas.
4. En general, se observa que el error disminuye de forma exponencial al incrementarse el valor de N .
5. Respecto al punto anterior: el único caso donde no se observa disminución exponencial del error fue en el problema interno para dominio rectangular con frontera virtual circular.
6. Se confirma el hecho ya consignado en la literatura que afirma la gran sensibilidad del error a la distancia entre la frontera virtual y la frontera real.
7. Respecto al punto anterior: en esta investigación se observa que para los problemas del tipo interno, la sensibilidad disminuye conforme aumenta el valor de N . Para los problemas externos la sensibilidad aumenta al incrementarse el valor de N .

8. Con base en los puntos 1 y 5: se sugiere, si se usan valores relativamente bajos de N , entonces incrementar la distancia entre la frontera real y la frontera virtual, a manera de compensación en el nivel de error.
9. Para los problemas internos se observó que el valor de N puede incrementarse hasta valores mayores, con respecto a los problemas externos, antes de que el cómputo deje de dar resultados confiables; por lo que en los problemas externos se sugiere no usar valores grandes de N , pero aumentar la distancia entre la frontera real y la virtual, para aumentar la exactitud. El nivel de error observado es mayor para los problemas externos.
10. En los problemas internos con dominio rectangular se observa que el nivel de error es mayor al usar una frontera virtual circular; mientras que en los problemas externos rectangulares el nivel de error es mayor al usar una frontera virtual rectangular.

Bibliografía

- Balakrishnan, K. & Ramachandran, P. A. (2000). The method of fundamental solutions for linear diffusion-reaction equations. *Math. Comput. Modelling*, (31), 221–237.
- Barnett, A. H. & Betcke, T. (2008). Stability and convergence of the method of fundamental solutions for helmholtz problems on analytic domains. *J. Comput.Phys.*, (227), 7003–7026.
- Bogomolny, A. (1985). Fundamental solutions method for elliptic boundary value problems. *SIAM Journal of Numerical Analysis*, 22, 644–669.
- Chazarain, J. & Piriou, A. (1982). *Introduction to the Theory of Linear Partial Differential Equations*. Studies in Mathematics and its Applications. The Netherlands: North-Holland Publishing Company.
- Chen, C. S. et al. (2009). The method of the fundamental solutions and its modifications for electromagnetic field problems. *Computer Assisted Mechanics and Engineering Sciences*, 16, 21–33.
- Colton, D. & Kress, R. (2013). *Integral Equations Methods in Scattering Theory*. SIAM, reprint of 1983 original edition.
- Colton., D. & Kress, R. (2013). *Inverse Acoustic and Electromagnetic Scattering Theory*. Springer, 3 edition.
- Deville, R., Godefroy, G., & Zizler, V. (1993). *Smoothness and renormings in Banach spaces*. Number 64 in Pitman Monographs and Surveys in Pure and Applied Mathematics. Longman Scientific & Technical.
- Doicu, A., Eremin, Y. A., & Wriedt, T. (2000). *Acoustic and Electromagnetic Scattering Analysis Using Discrete Sources*. USA: Elsevier Science.
- Edwards, R. E. (1994). *Functional Analysis*. USA: Dover.
- Eremin, Y. A., Stover, J. C., & Orlov, N. V. (1999). Modeling scatter from silicon wafer features based on discrete sources method. *Optical Engineering*, 38(8).
- Eremin, Y. A. & Sveshnikov, A. G. (1993). Discrete sources method in electromagnetic

- scattering problems. *Electromagnetics*, 13(2), 203–215.
- Eremin, Y. A. & Wriedt, T. (2004). Discrete sources method model for evanescent waves scattering analysis. *Journal of Quantitative Spectroscopy & Radiative Transfer*, (89), 53 – 65.
- Evans, L. C. (1998). *Partial Differential Equations*. Graduate Studies in Mathematics. USA: American Mathematical Society.
- Fairweather, G. & Karageorghis, A. (1998). The method of fundamental solutions for elliptic boundary value problems. *Advances in Computational Mathematics*, (9), 69 – 95.
- Golberg, M. & Chen, C. (1998). *The Method of Fundamental Solutions for Potential, Helmholtz and Diffusion Problems*. In: M.A. Golberg, eds., *Boundary Integral Methods – Numerical and Mathematical Aspects*, pp. 103-176. Computational Mechanics Publications.
- Heise, U. (1976). Numerical properties of integral equations in which the given boundary values and the sought solutions are defined on different curves. *Computers & Structures*, 8, 199–205.
- Karapetyants, A. & Kravchenko, V. (2022). *Methods of Mathematical Physics*. Switzerland: Birkhauser, 1st edition.
- Khmelnitskaya, K. V., Kravchenko, V. V., & Rabinovich, V. S. (2003). Quaternionic fundamental solutions for electromagnetic scattering problems and application. *Journal for Analysis and its Applications*, 22(1), 147–166.
- Kirillov, A. A. & Gvishiani, A. D. (1981). *Theorems and problems in Functional Analysis*. USA: Springer-Verlag.
- Kreyszig, E. (1978). *Introductory Functional Analysis with Applications*. USA: John Wiley & Sons.
- Kupradze, V. D. (1967). On the approximate solution of problems in mathematical physics. *Russian Mathematical Surveys*, 22(2), 58 – 108.
- Kupradze, V. D. & Aleksidze, M. A. (1963). An approximate method of solving certain boundary-value problems. *Soobsc. Akad. Nauk Gruzin, SSR* 30, 529–536.
- Mathews, J. H. & Fink, K. D. (1999). *Numerical Methods using MATLAB*. USA: Prentice

- Hall, 3rd edition.
- McLean, W. (2000). *Strongly Elliptic Systems and Boundary Integral Equations*. Cambridge University Press, 1 edition.
- Moiola, A. (2020). *Scattering of time-harmonic acoustic waves: Helmholtz equation, boundary integral equations and BEM*. Department of Mathematics, University of Pavia.
- Munkres, J. R. (2000). *Topology*. USA: Prentice Hall, 2nd edition.
- Pinchover, Y. & Rubinstein, J. (2005). *An Introduction to Partial Differential Equations*. USA: Cambridge University Press, 1st edition.
- Polyanin, A. D. & Nazaikinskii, V. E. (2016). *Handbook of Linear Partial Differential Equations for Engineers and Scientists*. CRC Press, 2 edition.
- Renardy, M. & Rogers, R. C. (2004). *An Introduction to Partial Differential Equations*. Texts in applied Mathematics. USA: Springer, 2nd edition.
- Rudin, W. (1973). *Functional Analysis*. USA: McGraw-Hill Book Company.
- Sayas, F. J. (2006). *Introduction to the boundary element method. A case study: the Helmholtz equation*. Universidad de Concepcion (Chile).
- Smyrlis, Y. S. (2009). Applicability and applications of the method of fundamental solutions. *Mathematics of Computation*, 78(267), 1399 – 1434.
- Tsai, C. C., Lin, Y. C., Young, D. L., & Atluri, S. N. (2006). Investigations on the accuracy and condition number for the method of fundamental solutions. *CMES: Computer Modeling in Engineering & Sciences*, 16(2), 103 – 114.
- Vladimirov, V. S. (1971). *Equations of Mathematical Physics*. USA: Marcel Dekker, Inc.
- Weinstock, B. M. (1973). Uniform approximation by solutions of elliptic equations. *Proceedings of the American Mathematical Society*, 41(2), 513 – 517.
- Wriedt, T. & Eremin, Y., Eds. (2018). *The Generalized Multipole Technique for Light Scattering*. Springer, 1st edition.

Apéndice A

CÓDIGOS DE LAS FUNCIONES GENERALES

A.1. *cboundary.m*

```
function [ ab,or ] = cboundary( a,b,n,t1,t2 )
%CBOUNDARY Calcula puntos sobre un arco de ELIPSE.
% Los argumentos de entrada de la función son: a>0,b>0, la longitud del
% semieje horizontal y del semieje vertical, respectivamente, de
% la elipse; n es el número de puntos que queremos calcular,
% n>=4; t1 y t2 son los valores inicial y final del parámetro,
% respectivamente, que nos permiten establecer el segmento de
% elipse considerado.
% Los argumentos de salida son: ab, or que son los vectores que guardan
% las abscisas y ordenadas, respectivamente, de los puntos calculados.
% NOTA: la función utiliza x=a*cos(t), y=b*sin(t) como ecuaciones
% paramétricas de la elipse.

%Validación.
if ~isscalar(a)||~isscalar(b)||~(b>0)||~(a>0)||t1<0||t1>2*pi||t2<0||t2>2*pi...
    ||n~=fix(n)||n<4
    error(['a y b deben ser escalares mayores a cero; n debe ser un entero',...
        ' al menos igual a 4; t1 y t2 deben tener valores en el intervalo',...
        '[0,2*pi].']);
end

%La función calculará n puntos sobre el arco de elipse definido por los
%valores del parámetro (t1 y t2) así como por los valores de a y b.

%Se procede al cálculo de los puntos sobre el arco de elipse.

%se inicializa el vector que contendrá las abscisas de los puntos calculados.
ab=[];
%se inicializa el vector que contendrá las ordenadas de los puntos calculados.
or=[];

for i=1:n
    if t1==0 && t2==2*pi || t1==2*pi && t2==0
        xc=a*cos((i-1)*(t2-t1)/n); %calculando la abscisa.
        yc=b*sin((i-1)*(t2-t1)/n); %calculando la ordenada.
    else
        xc=a*cos(t1+i*(t2-t1)/(n+1)); %calculando la abscisa.
        yc=b*sin(t1+i*(t2-t1)/(n+1)); %calculando la ordenada.
    end

    ab=[ab xc]; %guardando la abscisa calculada.
    or=[or yc]; %guardando la ordenada calculada.
end

end %fin del programa.
```

A.2. pboundary.m

```
function [ ab,or ] = pboundary( x,y,n,c )
%PBOUNDARY Calcula puntos sobre una trayectoria POLIGONAL.
% Los argumentos de entrada de la función son: x,y que son los vectores que
% contienen las abscisas y ordenadas, respectivamente, de los vértices
% del polígono, dichos vectores deben ser de igual longitud (mayor o igual
% a 3), una vez que el usuario determina cuál es el primer vértice, se debe
% elegir una dirección (sentido horario o antihorario) de recorrido del
% polígono y siguiendo esa dirección es el orden en que deben ser
% ingresados los demás vértices hasta llegar al último de ellos;
% n es el número de puntos que queremos calcular, n debe ser un entero al
% menos igual a la longitud de los vectores x y y; c nos permite indicar
% si queremos que la trayectoria poligonal se cierre entre el primer y el
% último vértice, si c=0 la trayectoria queda abierta, si c toma
% cualquier otro valor, la trayectoria se cierra.
% Los argumentos de salida son: ab, or que son los vectores que guardan
% las abscisas y ordenadas, respectivamente, de los puntos calculados.

if length(x) ~=length(y)||length(x)<3||n~=fix(n)||n<length(x)      %Validación.
    error(['El número de elementos del vector x y del vector y deben ser',...
           'iguales, dicha longitud debe ser al menos igual a 3; n debe ser',...
           'un entero al menos igual al tamaño del vector x.']);
end

%La función calculará n/(número de vértices) puntos sobre cada una de las
%aristas de la trayectoria poligonal cerrada, o bien tantos puntos como la
%parte entera del resultado de dicha operación. Se llamará j la variable
%que guardará el número de puntos calculados sobre cada arista y ari es la
%variable que guardará el número de aristas.
%La función calculará n/(número de vértices - 1) puntos sobre cada una de
%las aristas de la trayectoria poligonal abierta, o bien tantos puntos como
%la parte entera del resultado de dicha operación. Se llamará j la variable
%que guardará el número de puntos calculados sobre cada arista y ari es la
%variable que guardará el número de aristas.

if c~=0
    ari=length(x);
    j=fix(n/ari);

else
    ari=length(x)-1;
    j=fix(n/ari);

end

%Se procede al cálculo de los puntos sobre la trayectoria.

ab=[]; %se inicializa el vector que contendrá las abscisas de los puntos
        %calculados.
or=[]; %se inicializa el vector que contendrá las ordenadas de los puntos
        %calculados.
for i=1:ari %bucle que controla el avance sobre la trayectoria poligonal.
    for k=1:j %bucle que controla el avance sobre una arista.
```

```

if i~=ari || c==0
    xc=x(i)+(k/(j+1))*(x(i+1)-x(i));      %calculando la abscisa.
    yc=y(i)+(k/(j+1))*(y(i+1)-y(i));      %calculando la ordenada.
else
    xc=x(i)+(k/(j+1))*(x(1)-x(i));      %calculando la abscisa.
    yc=y(i)+(k/(j+1))*(y(1)-y(i));      %calculando la ordenada.
end
ab=[ab xc]; %guardando la abscisa calculada.
or=[or yc]; %guardando la ordenada calculada.
end
end

end %fin del programa.

```

Published with MATLAB® R2014a

A.3. *sol_aproximada.m*

```
function [ sol ] = sol_aproximada( k,coef,X1,X2,Y1,Y2 )  
%SOL_APROXIMADA Evalúa la aproximación obtenida en el DSM.  
% Los argumentos de entrada son: k, que es el número de onda; el vector  
% coef que contiene los coeficientes del DSM; X1 y X2 son los vectores que  
% contienen las abscisas y ordenadas, respectivamente, de los puntos de  
% prueba. Y1 y Y2 contienen las abscisas y ordenadas, respectivamente,  
% de las fuentes discretas.  
% El único argumento de salida es: sol, que es un vector que guarda los  
% valores obtenidos en las evaluaciones.  
  
N1=length(X1);  
N=length(Y1);  
sol=zeros(1,N1);  
xa=zeros(1,N);  
ya=zeros(1,N);  
  
%Se procede a la evaluación  
for i=1:N1  
    xa(1,:)=X1(i);  
    ya(1,:)=X2(i);  
    d=sqrt((xa-Y1).^2+(ya-Y2).^2);  
    eval=sum(coef.* (besselh(0,k*d)).'); %en este punto se ejecuta la  
                                              %evaluación.  
    sol(i)=eval;  
end  
  
end %fin del programa.
```

Published with MATLAB® R2014a

Apéndice B

CÓDIGOS DE LOS SCRIPTS Y FUNCIONES PARA LOS EJEMPLOS

B.1. Ejemplo 1

```
%{
EJEMPLO 1
    Eliptico interno. Frontera virtual eliptica.
    Dominio: {(x,y) en R2: x^2/25+y^2<1}
    Problema:
        { delta(u)+2u=0,   (x,y) en el dominio
          u(x,y)=cos xcos(sqrt(1-x^2/25)), (x,y) en la frontera
        Solución exacta: u(x,y)=cos xcos y
    }

%***** INICIALIZANDO *****
clc      %limpiando ventana de comandos
clear all %limpiando el Workspace
close all %cerrando todas las figuras

k=sqrt(2); %número de onda
R=5;       %longitud del semieje horizontal
R1=1;      %longitud del semieje vertical
%N=20;     %cantidad de términos deseados en la aproximación
fd=1+ 0.1; %factor de distancia deseado para la frontera virtual.
            %Ajustar el segundo sumando, de manera que fd > 1
N_inic=50; %valor inicial para N
incr=50;   %incremento deseado para N
N_fin=500;  %valor final para N

N1=50;      %cantidad de puntos de prueba en cada distancia
fd_min=0.1; %factor de distancia mínimo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                    %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores relativos máximos a lo largo del interior
                            %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores absolutos máximos a lo largo del interior
                            %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
%***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

    cont=cont+1;
    [X1,X2]=cboundary( R,R1,N,0,2*pi ); %puntos de colocación (frontera real)
    [Y1,Y2]=cboundary( R*fd,R1*fd,N,0,2*pi );%fuentes discretas (frontera virtual)

    if cont==3
        figure
        plot(X1,X2,'*',Y1,Y2,'o');
    end
}
```

```

%axis square;
title('FRONTERA REAL Y FRONTERA VIRTUAL');
xlabel('Eje x');
ylabel('Eje y');
legend('Puntos de colocación','Fuentes discretas','Location','Best');
box off
end

***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej1(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

***** VALIDACIÓN Y GRÁFICAS *****
x=[]; %en este vector se guardan las abscisas de los puntos de prueba
y=[]; %en este vector se guardan las ordenadas de los puntos de prueba
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(fd_min,0.95,9)
    [Z1,Z2]=cboundary( R*r,R1*r,N1,0,2*pi ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej1(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)./abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');
    mesh(xq,yq,vq);
    hold on
    plot3(x,y,z_aprox,'mo');
    %axis square
    legend('Soln. exacta','Soln. aproximada');
    title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
    xlabel('Eje x');
    ylabel('Eje y');
    zlabel('Valor de u');

```

```

        hold off
    end

end

% Graficando el error relativo máximo
figure
rl=linspace(fd_min,0.95,9);
semilogy(rl,Matriz_err_r_max,'-.d','LineWidth',2.5);
grid on
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

% Graficando el error absoluto máximo
figure
rl=linspace(fd_min,0.95,9);
semilogy(rl,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

%Graficando la superficie de error relativo
C=Matriz_errores_r{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

%Graficando la superficie de error absoluto
C=Matriz_errores_a{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error absoluto');
hold off

```

```
%fin del script
```

Published with MATLAB® R2014a

```
function [ M,B ] = Matriz_extendida_ej1( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ1 Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej1(X1(i),X2(i));
end
end %fin del programa.
```

Published with MATLAB® R2014a

```
function [ r ] = sol_exacta_ej1( x,y )
%SOL_EXACTA_EJ1 Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

    %Evaluación de la solución exacta.

r=cos(x).*cos(y);      %opción de solución exacta

end %fin del programa.
```

Published with MATLAB® R2014a

B.2. Ejemplo 2

B.2.1. 2a)

```
EJEMPLO 2a
Rectangular interno. Frontera virtual circular.
Dominio: {(x,y)en R2:-1<x<1,-3<y<3}
Problema:
{delta(u)+2u=0,           (x,y) en el dominio
u(x,y)=(1-y/(i*sqrt(2)))e^{iy*sqrt(2)}-sinh(iy*sqrt(2))/2,x en {-1,1}, -3<y<3
u(x,y)=(x^2-3/(i*sqrt(2)))e^{i*3sqrt(2)}-sinh(i*3sqrt(2))/2,      y=3, -1<x<1
u(x,y)=(x^2+3/(i*sqrt(2)))e^{-i*3sqrt(2)}-sinh(-i*3sqrt(2))/2,   y=-3, -1<x<1
Solución exacta: u(x,y)=(x^2-y/(i*sqrt(2)))e^{iy*sqrt(2)}-sinh(iy*sqrt(2))/2
%}

***** INICIALIZANDO *****
clc      %limpiando ventana de comandos
clear all %limpiando el Workspace
close all %cerrando todas las figuras

k=sqrt(2); %número de onda
a=2;      %longitud de la base del rectángulo
b=6;      %longitud de la altura del rectángulo
%N=20;     %cantidad de términos deseados en la aproximación
fd=sqrt(a^2+b^2)/a+ 0.19;    %factor de distancia deseado para la
                             %frontera virtual. Ajustar el segundo sumando
N_inic=40; %valor inicial para N, debe ser múltiplo de 4
incr=40;   %incremento deseado para N, debe ser múltiplo de 4
N_fin=400; %valor final para N, debe ser múltiplo de 4

N1=100;    %cantidad de puntos de prueba en cada distancia
fd_min=0.1; %factor de distancia mínimo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                     %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                             %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                             %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                             %errores relativos máximos a lo largo del interior
                             %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                             %errores absolutos máximos a lo largo del interior
                             %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

cont=cont+1;
ab_v=[a/2 -a/2 -a/2 a/2];    %abscisas de los vértices
or_v=[b/2 b/2 -b/2 -b/2];    %ordenadas de los vértices
[X1,X2]=pboundary( ab_v,or_v,N,1 ); %puntos de colocación (frontera real)
[Y1,Y2]=cboundary( a/2*fd,a/2*fd,N,0,2*pi );%fuentes discretas (frontera virtual)
```

```

if cont==3
    figure
    plot(X1,X2,'*',Y1,Y2,'o');
    axis square;
    title('FRONTERA REAL Y FRONTERA VIRTUAL');
    xlabel('Eje x');
    ylabel('Eje y');
    legend('Puntos de colocación','Fuentes discretas','Location','Best');
    box off
end

%***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej2a(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

%***** VALIDACIÓN Y GRÁFICAS *****
x=[];
y=[];
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(fd_min,0.95,9)
    [Z1,Z2]=pboundary( ab_v*r,or_v*r,N1,1 ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej2a(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)./abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');
    mesh(xq,yq,vq);
    hold on
    plot3(x,y,z_aprox,'mo');
    %axis square
    legend('Soln. exacta','Soln. aproximada');

```

```

        title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
        xlabel('Eje x');
        ylabel('Eje y');
        zlabel('Valor de Re\{u\}');
        hold off
    end

    end

% Graficando el error relativo máximo
figure
rl=linspace(fd_min,0.95,9);
semilogy(rl,Matriz_err_r_max,'-.d','LineWidth',2.5);
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

% Graficando el error absoluto máximo
figure
rl=linspace(fd_min,0.95,9);
semilogy(rl,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

%Graficando la superficie de error relativo
C=Matriz_errores_r{3};
v=C(:,');
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

%Graficando la superficie de error absoluto
C=Matriz_errores_a{3};
v=C(:,';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');

```

```

zlabel('error absoluto');
hold off

%fin del script

```

Published with MATLAB® R2014a

```

function [ M,B ] = Matriz_extendida_ej2a( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ2A Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej2a(X1(i),X2(i));
end
end %fin del programa.

```

Published with MATLAB® R2014a

```

function [ r ] = sol_exacta_ej2a( x,y )
%SOL_EXACTA_EJ2A Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

    %Evaluación de la solución exacta.

r=(x.^2-y/(i*sqrt(2))).*exp(sqrt(2)*i*y)-1/2*sinh(sqrt(2)*i*y);
%opción de solución exacta

end %fin del programa.

```

Published with MATLAB® R2014a

B.2.2. 2b)

```

%{
EJEMPLO 2b
    Rectangular interno. Frontera virtual rectangular.
    Dominio: {(x,y) en R2:-1<x<1,-3<y<3}
    Problema:
        {delta(u)+2u=0,           (x,y) en el dominio
        u(x,y)=(1-y/(i*sqrt(2)))e^{iy*sqrt(2)}-sinh(iy*sqrt(2))/2, x en {-1,1}, -3<y<3
        u(x,y)=(x^2-3/(i*sqrt(2)))e^{i*3sqrt(2)}-sinh(i*3sqrt(2))/2,      y=3, -1<x<1
        u(x,y)=(x^2+3/(i*sqrt(2)))e^{-i*3sqrt(2)}-sinh(-i*3sqrt(2))/2,   y=-3, -1<x<1
        Solución exacta: u(x,y)=(x^2-y/(i*sqrt(2)))e^{iy*sqrt(2)}-sinh(iy*sqrt(2))/2
    %}

***** INICIALIZANDO *****
clc      %limpiando ventana de comandos
clear all %limpiando el Workspace
close all %cerrando todas las figuras

k=sqrt(2); %número de onda
a=2;      %longitud de la base del rectángulo
b=6;      %longitud de la altura del rectángulo
%N=20;     %cantidad de términos deseados en la aproximación
fd=1+ 0.1; %factor de distancia deseado para la frontera virtual.
            %Ajustar el segundo sumando, de manera que fd > 1
N_inic=40; %valor inicial para N, debe ser múltiplo de 4
incr=40;   %incremento deseado para N, debe ser múltiplo de 4
N_fin=400; %valor final para N, debe ser múltiplo de 4

N1=100;    %cantidad de puntos de prueba en cada distancia
fd_min=0.1; %factor de distancia mínimo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                    %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores relativos máximos a lo largo del interior
                            %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores absolutos máximos a lo largo del interior
                            %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

    cont=cont+1;
    ab_v=[a/2 -a/2 -a/2 a/2];    %abscisas de los vértices
    or_v=[b/2 b/2 -b/2 -b/2];    %ordenadas de los vértices
    [X1,X2]=pboundary( ab_v,or_v,N,1 ); %puntos de colocación (frontera real)
    [Y1,Y2]=pboundary( ab_v*fd,or_v*fd,N,1 );%fuentes discretas (frontera virtual)

```

```

if cont==3
    figure
    plot(X1,X2,'*',Y1,Y2,'o');
    axis square;
    title('FRONTERA REAL Y FRONTERA VIRTUAL');
    xlabel('Eje x');
    ylabel('Eje y');
    legend('Puntos de colocación','Fuentes discretas','Location','Best');
    box off
end

%***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej2b(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

%***** VALIDACIÓN Y GRÁFICAS *****
x=[];
y=[];
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(fd_min,0.95,9)
    [Z1,Z2]=pboundary( ab_v*r,or_v*r,N1,1 ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej2b(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)./abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');
    mesh(xq,yq,vq);
    hold on
    plot3(x,y,z_aprox,'mo');
    %axis square
    legend('Soln. exacta','Soln. aproximada');

```

```

    title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
    xlabel('Eje x');
    ylabel('Eje y');
    zlabel('Valor de Re\{u\}');
    hold off
end

end

% Graficando el error relativo máximo
figure
r1=linspace(fd_min,0.95,9);
semilogy(r1,Matriz_err_r_max,'-.d','LineWidth',2.5);
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

% Graficando el error absoluto máximo
figure
r1=linspace(fd_min,0.95,9);
semilogy(r1,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

%Graficando la superficie de error relativo
C=Matriz_errores_r{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

%Graficando la superficie de error absoluto
C=Matriz_errores_a{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');

```

```

zlabel('error absoluto');
hold off

%fin del script

```

Published with MATLAB® R2014a

```

function [ M,B ] = Matriz_extendida_ej2b( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ2B Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej2b(X1(i),X2(i));
end
end %fin del programa.

```

Published with MATLAB® R2014a

```

function [ r ] = sol_exacta_ej2b( x,y )
%SOL_EXACTA_EJ2B Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

%Evaluación de la solución exacta.

r=(x.^2-y/(i*sqrt(2))).*exp(sqrt(2)*i*y)-1/2*sinh(sqrt(2)*i*y);
%opción de solución exacta

end %fin del programa.

```

Published with MATLAB® R2014a

B.3. Ejemplo 3

```
%{
EJEMPLO 3
    Elíptico externo. Frontera virtual elíptica.
    Dominio: {(x,y) en R2: x^2/25+y^2>1}
    Problema:
        (delta(u)+2u=0, (x,y) en el dominio
        u(x,y)=besselh(0,sqrt(2)*sqrt((24/25)x^2+1)), (x,y) en la frontera
        du/dr-iku=o(1/sqrt(r)), cuando r->+Infinito. (Cond. de Sommerfeld)
        Solución exacta: u(x,y)=besselh(0,sqrt(2)*sqrt(x.^2+y.^2))
}

***** INICIALIZANDO *****
clc      %limpiando ventana de comandos
clear all %limpiando el Workspace
close all %cerrando todas las figuras

k=sqrt(2); %número de onda
R=5;        %longitud del semieje horizontal
R1=1;       %longitud del semieje vertical
%N=20;      %cantidad de términos deseados en la aproximación
fd=1- 0.2;  %factor de distancia deseado para la frontera virtual.
            %Ajustar el segundo sumando de manera que fd < 1
N_inic=10;  %valor inicial para N
incr=10;    %incremento deseado para N
N_fin=100;   %valor final para N

N1=50;      %cantidad de puntos de prueba en cada distancia
fd_max=1.9; %factor de distancia máximo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                    %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores relativos máximos a lo largo del interior
                            %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores absolutos máximos a lo largo del interior
                            %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

cont=cont+1;
[X1,X2]=cboundary( R,R1,N,0,2*pi ); %puntos de colocación (frontera real)
[Y1,Y2]=cboundary( R*fd,R1*fd,N,0,2*pi );%fuentes discretas (frontera virtual)

if cont==3
    figure

```

```

plot(X1,X2,'*',Y1,Y2,'o');
%axis square;
title('FRONTERA REAL Y FRONTERA VIRTUAL');
xlabel('Eje x');
ylabel('Eje y');
legend('Puntos de colocación','Fuentes discretas','Location','Best');
box off
end

%***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej3(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

%***** VALIDACIÓN Y GRÁFICAS *****
x=[];
y=[];
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(1.05,fd_max,9)
    [Z1,Z2]=cboundary( R*r,R1*r,N1,0,2*pi ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej3(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)./abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');
    mesh(xq,yq,vq);
    hold on
    plot3(x,y,z_aprox,'mo');
    %axis square
    legend('Soln. exacta','Soln. aproximada');
    title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
    xlabel('Eje x');
    ylabel('Eje y');

```

```

% Graficando el error relativo máximo
figure
rl=linspace(1.05,fd_max,9);
semilogy(rl,Matriz_err_r_max,'-.d','LineWidth',2.5);
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

% Graficando el error absoluto máximo
figure
rl=linspace(1.05,fd_max,9);
semilogy(rl,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

% Graficando la superficie de error relativo
C=Matriz_errores_r{3}';
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

% Graficando la superficie de error absoluto
C=Matriz_errores_a{3}';
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error absoluto');
hold off

% fin del script

```

Published with MATLAB® R2014a

```

function [ M,B ] = Matriz_extendida_ej3( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ3 Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej3(X1(i),X2(i));
end
end %fin del programa.

```

Published with MATLAB® R2014a

```
function [ r ] = sol_exacta_ej3( x,y )
%SOL_EXACTA_EJ3 Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

%Evaluación de la solución exacta.

r=besselh(0,sqrt(2)*sqrt(x.^2+y.^2)); %opción de solución exacta
%opción de solución exacta

end %fin del programa.
```

Published with MATLAB® R2014a

B.4. Ejemplo 4

B.4.1. 4a)

```
%{
EJEMPLO 4a
    Rectangular externo. Frontera virtual elíptica.
    Dominio: {(x,y)en R2:-1<=x<=1, -3<=y<=3}^c
    Problema:
        (delta(u)+2u=0,           (x,y) en el dominio
        u(x,y)=besselh(0,sqrt(2)*sqrt(1+y^2))+...
                    besselh(0,sqrt(2)*sqrt(1.21+y^2)),   x=-1,-3<=y<3
        u(x,y)=besselh(0,sqrt(2)*sqrt(1+y^2))+...
                    besselh(0,sqrt(2)*sqrt(0.81+y^2)),   x=1,-3<=y<3
        u(x,y)=besselh(0,sqrt(2)*sqrt(x^2+9))+...
                    besselh(0,sqrt(2)*sqrt((x-0.1)^2+9)),   y en {-3,3}, -1<x<1
        du/dr-iku=o(1/sqrt(r)), cuando r->+Infinito. (Cond. de Sommerfeld)
    Solución exacta: u(x,y)=besselh(0,sqrt(2)*sqrt(x^2+y^2))+...
                                besselh(0,sqrt(2)*sqrt((x-0.1)^2+y^2))
}

%***** INICIALIZANDO *****
clc      %limpiando ventana de comandos
clear all %limpiando el Workspace
close all %cerrando todas las figuras

k=sqrt(2); %número de onda
a=2;      %longitud de la base del rectángulo
b=6;      %longitud de la altura del rectángulo
%N=20;    %cantidad de términos deseados en la aproximación
fd=1- 0.2; %factor de distancia deseado para la frontera virtual.
            %Ajustar el segundo sumando de manera que fd < 1
N_inic=12; %valor inicial para N, debe ser múltiplo de 4
incr=8;    %incremento deseado para N, debe ser múltiplo de 4
N_fin=84;  %valor final para N, debe ser múltiplo de 4

N1=100;    %cantidad de puntos de prueba en cada distancia
fd_max=1.9; %factor de distancia máximo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                    %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                            %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores relativos máximos a lo largo del interior
                            %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores absolutos máximos a lo largo del interior
                            %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
%***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

```

```

cont=cont+1;
ab_v=[a/2 -a/2 -a/2 a/2]; %abscisas de los vértices
or_v=[b/2 b/2 -b/2 -b/2]; %ordenadas de los vértices
[X1,X2]=pboundary( ab_v,or_v,N,1 ); %puntos de colocación (frontera real)
[Y1,Y2]=cboundary( a/2*fd,b/2*fd,N,0,2*pi );%fuentes discretas (frontera virtual

if cont==3
    figure
    plot(X1,X2,'*',Y1,Y2,'o');
    %axis square;
    title('FRONTERA REAL Y FRONTERA VIRTUAL');
    xlabel('Eje x');
    ylabel('Eje y');
    legend('Puntos de colocación','Fuentes discretas','Location','Best');
    box off
end

%***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej4a(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

%***** VALIDACIÓN Y GRÁFICAS *****
x=[];
y=[];
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(1.05,fd_max,9)
    [Z1,Z2]=pboundary( ab_v*r,or_v*r,N1,1 ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej4a(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)./abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');

```

```

mesh(xq,yq,vq);
hold on
plot3(x,y,z_aprox,'mo');
%axis square
legend('Soln. exacta','Soln. aproximada');
title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
xlabel('Eje x');
ylabel('Eje y');
zlabel('Valor de Re\{u\}');
hold off
end

end

% Graficando el error relativo máximo
figure
r1=linspace(1.05,fd_max,9);
semilogy(r1,Matriz_err_r_max,'-.d','LineWidth',2.5);
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

% Graficando el error absoluto máximo
figure
r1=linspace(1.05,fd_max,9);
semilogy(r1,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)),'N= %d términos','Location','Best');

%Graficando la superficie de error relativo
C=Matriz_errores_r{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

%Graficando la superficie de error absoluto
C=Matriz_errores_a{3};
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on

```

```

plot3(x,y,v, 'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error absoluto');
hold off

%fin del script

```

Published with MATLAB® R2014a

```

function [ M,B ] = Matriz_extendida_ej4a( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ4A Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej4a(X1(i),X2(i));
end
end %fin del programa.

```

Published with MATLAB® R2014a

```

function [ r ] = sol_exacta_ej4a( x,y )
%SOL_EXACTA_EJ4A Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

%Evaluación de la solución exacta.

r=besselh(0,sqrt(2)*sqrt(x.^2+y.^2))+...
    besselh(0,sqrt(2)*sqrt((x-0.1).^2+y.^2)); %opción de solución exacta
%opción de solución exacta

end %fin del programa.

```

Published with MATLAB® R2014a

B.4.2. 4b)

```
%{
EJEMPLO 4b
    Rectangular externo. Frontera virtual rectangular.
    Dominio: {(x,y)en R2:-1<=x<=1,-3<=y<=3}^c
    Problema:
        {delta(u)+2u=0,           (x,y) en el dominio
        u(x,y)=besselh(0,sqrt(2)*sqrt(1+y^2))+...
            besselh(0,sqrt(2)*sqrt(1.21+y^2)),      x=-1,-3<=y<3
        u(x,y)=besselh(0,sqrt(2)*sqrt(1+y^2))+...
            besselh(0,sqrt(2)*sqrt(0.81+y^2)),      x=1,-3<=y<3
        u(x,y)=besselh(0,sqrt(2)*sqrt(x^2+9))+...
            besselh(0,sqrt(2)*sqrt((x-0.1)^2+9)),   y en {-3,3}, -1<x<1
        du/dr-iku=o(1/sqrt(r)), cuando r->+Infinito. (Cond. de Sommerfeld)
    Solución exacta: u(x,y)=besselh(0,sqrt(2)*sqrt(x^2+y^2))+...
                    besselh(0,sqrt(2)*sqrt((x-0.1)^2+y^2))
    %}

%***** INICIALIZANDO *****
clc          %limpiando ventana de comandos
clear all    %limpiando el Workspace
close all    %cerrando todas las figuras

k=sqrt(2); %número de onda
a=2;         %longitud de la base del rectángulo
b=6;         %longitud de la altura del rectángulo
%N=20;        %cantidad de términos deseados en la aproximación
fd=1- 0.2;  %factor de distancia deseado para la frontera virtual.
             %Ajustar el segundo sumando de manera que fd < 1
N_inic=8;  %valor inicial para N, debe ser múltiplo de 4
incr=8;    %incremento deseado para N, debe ser múltiplo de 4
N_fin=80;  %valor final para N, debe ser múltiplo de 4

N1=100;     %cantidad de puntos de prueba en cada distancia
fd_max=1.9; %factor de distancia máximo para los puntos de prueba

cont=0;
n_cond=zeros(1,3); %vector donde se guardará el no. de condición
                     %de las diversas iteraciones.
Matriz_errores_r=cell(1,3); %matriz de celdas donde se guardarán las
                             %matrices de errores relativos en cada iteración.
Matriz_errores_a=cell(1,3); %matriz de celdas donde se guardarán las
                             %matrices de errores absolutos en cada iteración.
Matriz_err_r_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores relativos máximos a lo largo del interior
                            %del dominio, en cada iteración.
Matriz_err_a_max=zeros(3,9); %matriz que guardará en sus renglones los
                            %errores absolutos máximos a lo largo del interior
                            %del dominio, en cada iteración.

for N= N_inic:incr:N_fin
%***** CALCULANDO PUNTOS DE COLOCACIÓN Y FUENTES DISCRETAS *****

```

```

cont=cont+1;
ab_v=[a/2 -a/2 -a/2 a/2]; %abscisas de los vértices
or_v=[b/2 b/2 -b/2 -b/2]; %ordenadas de los vértices
[X1,X2]=pboundary( ab_v,or_v,N,1 ); %puntos de colocación (frontera real)
[Y1,Y2]=pboundary( ab_v*fd,or_v*fd,N,1 );%fuentes discretas (frontera virtual)

if cont==3
    figure
    plot(X1,X2,'*',Y1,Y2,'o');
    %axis square;
    title('FRONTERA REAL Y FRONTERA VIRTUAL');
    xlabel('Eje x');
    ylabel('Eje y');
    legend('Puntos de colocación','Fuentes discretas','Location','Best');
    box off
end

%***** EJECUTANDO EL DSM Y CALCULANDO EL NÚMERO DE CONDICIÓN *****
tic
[M,B]=Matriz_extendida_ej4b(k,X1,X2,Y1,Y2);
coef=M\B;
toc
n_cond(cont)=cond(M);

%***** VALIDACIÓN Y GRÁFICAS *****
x=[];
y=[];
z_aprox=[];
z_exact=[];
error_r=[];
error_a [];

for r =linspace(1.05,fd_max,9)
    [Z1,Z2]=pboundary( ab_v*r,or_v*r,N1,1 ); %generando puntos de prueba
    sol_aprox =sol_aproximada(k,coef,Z1,Z2,Y1,Y2);
    sol_exact =sol_exacta_ej4b(Z1,Z2);
    error_a=[error_a; abs(sol_exact-sol_aprox)];
    error_r=[error_r; abs(sol_exact-sol_aprox)/abs(sol_exact)];
    x=[x Z1]; %guardando puntos de prueba
    y=[y Z2];
    z_aprox=[z_aprox real(sol_aprox)];
    z_exact=[z_exact real(sol_exact)];
end
Matriz_errores_r(cont)={error_r};
Matriz_err_r_max(cont,:)=(max(error_r,[],2))';
Matriz_errores_a(cont)={error_a};
Matriz_err_a_max(cont,:)=(max(error_a,[],2))';

% Graficando la solución exacta y la aproximada
if cont==1||cont==3
    figure
    [xq,yq]=meshgrid(linspace(min(x),max(x),100),linspace(min(y),max(y),100));
    vq=griddata(x,y,z_exact,xq,yq,'cubic');

```

```

mesh(xq,yq,vq);
hold on
plot3(x,y,z_aprox,'mo');
%axis square
legend('Soln. exacta','Soln. aproximada');
title('SOLUCIÓN EXACTA, Y APROXIMADA POR DSM');
xlabel('Eje x');
ylabel('Eje y');
zlabel('Valor de Re\{u\}');
hold off
end

end

% Graficando el error relativo máximo
figure
rl=linspace(1.05,fd_max,9);
semilogy(rl,Matriz_err_r_max,'-.d','LineWidth',2.5);
title('ERROR RELATIVO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error relativo máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

% Graficando el error absoluto máximo
figure
rl=linspace(1.05,fd_max,9);
semilogy(rl,Matriz_err_a_max,'-.d','LineWidth',2.5);
title('ERROR ABSOLUTO MÁXIMO A LO LARGO DEL DOMINIO');
xlabel('Razón de homotecia');
ylabel('Error absoluto máximo');
legend(num2str((N_inic:incr:N_fin)', 'N= %d términos'), 'Location', 'Best');

%Graficando la superficie de error relativo
C=Matriz_errores_r{3}';
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on
plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR RELATIVO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error relativo');
hold off

%Graficando la superficie de error absoluto
C=Matriz_errores_a{3}';
v=C(:)';
vq=griddata(x,y,v,xq,yq,'linear');
figure
surf(xq,yq,vq);
hold on

```

```

plot3(x,y,v,'om');
%axis square
title('SUPERFICIE DEL ERROR ABSOLUTO');
xlabel('Eje x');
ylabel('Eje y');
zlabel('error absoluto');
hold off

%fin del script

```

Published with MATLAB® R2014a

```

function [ M,B ] = Matriz_extendida_ej4b( k,X1,X2,Y1,Y2 )
%MATRIZ_EXTENDIDA_EJ4B Calcula la matriz extendida para el DSM.
% Los argumentos de entrada son: k (número de onda); X1 y X2, vectores
% que contienen las abscisas y ordenadas, respectivamente, de los puntos
% de colocación; los vectores Y1 y Y2 contienen las abscisas y ordenadas,
% respectivamente, de las fuentes discretas.
% Los argumentos de salida son M y B. M es la matriz del sistema y
% B contiene los valores en la frontera.

N=length(Y1);
M=zeros(N,N);
B=zeros(N,1);
%Se procede al cálculo de M y de B
for i=1:N
    for j=1:N
        d=sqrt((X1(i)-Y1(j)).^2+(X2(i)-Y2(j)).^2);
        v=besselh(0,k*d);
        M(i,j) = v;
    end
    B(i)=sol_exacta_ej4b(X1(i),X2(i));
end
end %fin del programa.

```

Published with MATLAB® R2014a

```

function [ r ] = sol_exacta_ej4b( x,y )
%SOL_EXACTA_EJ4B Evalúa la solución exacta para el DSM.
% Los argumentos de entrada son x y y que son los vectores que contienen
% las abscisas y ordenadas, respectivamente, de los puntos donde se
% evaluará la soln. exacta. El argumento de salida es el vector r que
% contiene los valores obtenidos en la evaluación.

%Evaluación de la solución exacta.

r=besselh(0,sqrt(2)*sqrt(x.^2+y.^2))+...
    besselh(0,sqrt(2)*sqrt((x-0.1).^2+y.^2)); %opción de solución exacta

end %fin del programa.

```

Published with MATLAB® R2014a

Apéndice C

NOTACIÓN

Se presenta a continuación una lista con los principales símbolos utilizados en esta tesis.

SÍMBOLO CONCEPTO REPRESENTADO

$\phi(x, s)$	solución fundamental
\mathcal{L}	operador diferencial parcial
u	solución de una ecuación diferencial parcial
Ω	dominio en estudio
$supp \psi(x)$	soporte de la función ψ
s	coordenadas de la ubicación de una fuente puntual
$\partial\Omega$	frontera del dominio Ω ; frontera real
$\partial\Omega'$	frontera virtual
$\{s_j\}_{j=1}^M$	conjunto de las ubicaciones de las fuentes puntuales
$\{x_k\}_{k=1}^N$	conjunto de puntos de colocación para un problema dado
$u_{M,N}(x, c)$	solución aproximada para un problema dado
$c, \{c_j\}_{j=1}^N$	coeficientes por determinar en el DSM
$f(x_k)$	condición de frontera evaluada en el punto x_k
M	cantidad de fuentes puntuales
N	cantidad de puntos de colocación
B	operador diferencial asociado con la función f (cond. de frontera)
Ω^c	complemento del conjunto Ω , relativo a \mathbb{R}^n
X'	espacio dual del espacio vectorial X
x_c	centroide del dominio en estudio
$f. d.$	factor que determina el alejamiento de $\partial\Omega'$ con respecto a Ω
R	parámetro de reescalamiento
Δ	operador diferencial de Laplace