Sociedad Mexicana de Computación Científica
y sus Aplicaciones

Boletín

Sociedad Mexicana de Computación Científica y sus Aplicaciones

Año IX - Número 9
Diciembre 2023

El Boletín Sociedad Mexicana de Computación Científica y sus Aplicaciones publica artículos de investigación originales y de alta calidad en las áreas de matemáticas aplicadas y computación científica, así como artículos de difusión científica. Todos los artículos son sometidos a una revisión por expertos en estas áreas de instituciones nacionales e internacionales.

El Boletín Sociedad Mexicana de Computación Científica y sus Aplicaciones A. C. (SMCCA), Año IX, No. 9, Diciembre 2023, es una publicación oficial anual editada por la Sociedad Mexicana de Computación Científica y sus Aplicaciones A. C., calle Luis Horacio Salinas, 545, Col. Valle de Morelos, Saltillo, Coahuila, C.P. 25013, Tel. (844) 133 5647, www.smcca.org.mx.

Editor responsable: Gerardo Tinoco Guerrero. Reserva de Derechos al Uso Exclusivo No. 04 - 2017 - 103114330600 - 203, ISSN: 2594-0457, ambos otorgados por el Instituto Nacional de Derechos de Autor. Responsable de la última actualización de este Número, Gerardo Tinoco Guerrero, Avenida Francisco J. Mújica S/N, Ciudad Universitaria, Edificio B, Morelia, Michoacán, C.P. 58030, fecha de la última modificación: 3 de noviembre de 2025.

Suscripciones al Boletín
smcca@smcca.org.mx
https://www.scipedia.com/sj/smcca

Índice general

Carta de Bienvenida

La Sociedad Mexicana de Computación Científica y sus Aplicaciones, A.C. (SMCCA) y el Comité Editorial, les damos una cordial bienvenida a la novena edición del Boletín electrónico anual de la SMCCA, el cual tiene como objetivo mantenerlos informados de las actividades realizadas por la SMCCA y sus asociados. En el Boletín se publican noticias, eventos, artículos de divulgación y de investigación de alto nivel en el área de las Matemáticas Aplicadas y Cómputo Científico, así como resúmenes de las mejores tesis de Licenciatura en Matemáticas Aplicadas.

En esta novena edición del boletín se presentan: una breve semblanza de la XXXI Escuela Nacional de Optimización y Análisis Numérico, llevada a cabo en conjunto con la Unidad Cuernavaca del Instituto de Matemáticas de la UNAM y la Facultad de Contaduría, Administración e Informática y el Centro de Investigación en Ciencias de la Universidad Autónoma del Estado de Morelos, en modalidad híbrida del 26 al 30 de junio del presente año, en la Ciudad de Cuernavaca, Morelos; e incluye dos artículos de investigación por solicitud, y tres artículos de divulgación, dos de ellos por solicitud y el otro corresponde a la ganadora de la vigésima primera edición del premio Mixbaal.

La SMCCA agradecerá que ante el interés que surja en los lectores en los temas que se presenten en nuestra publicación, éstos se conviertan en usuarios asiduos, así como en miembros activos de nuestra Sociedad. La información del registro de membresías a la SMCCA la pueden consultar en el Módulo de Registro de nuestra página .

Justino Alavez Ramírez
Presidente
Sociedad Mexicana de Computación Científica y sus Aplicaciones, A.C.

Reseña de la XXXI Escuela Nacional de Optimización y Análisis Numérico

Justino Alavez Ramírez,
Rina Betzabeth Ojeda Castañeda,
Gilberto Calvillo Vives

Del 26 al 30 de junio de 2023 se celebró con gran éxito la XXXI edición de la Escuela Nacional de Optimización y Análisis Numérico (ENOAN 2023). Se realizó conjuntamente con la Unidad Cuernavaca del Instituto de Matemáticas (UCIM) de la UNAM y la Facultad de Contaduría, Administración e Informática (FCAeI) y el Centro de Investigación en Ciencias (CInC) de la Universidad Autónoma del Estado de Morelos, en modalidad híbrida en la Ciudad de Cuernavaca, Morelos.

El comité organizador estuvo constituido por el Comité Local formado por profesores-investigadores, administradores y técnicos de la UCIM, de la FCAeI y del CInC de la Universidad Autónoma del Estado de Morelos, y el Comité Nacional conformado por profesores-investigadores miembros en activo de la Sociedad Mexicana de Computación Científica y sus Aplicaciones (SMCCA).

Como en anteriores ediciones, las actividades académicas realizadas en esta ENOAN 2023 comprendieron: 10 cursos cortos, 6 conferencias plenarias y 29 conferencias invitadas a cargo de reconocidos investigadores tanto nacionales como internacionales, 15 trabajos de investigación en formato de cartel y 32 en formato de

comunicación oral agrupados en 6 sesiones temáticas; adicionalmente se tuvieron tres sesiones especiales: la sesión del "IV Mini-Simposium de Medicina y Matemáticas (Cáncer, Dengue, Neuro, Obesidad y Diabetes)", la sesión de "Finanzas" y la sesión del "II Foro Conjunto de Sociedades Científicas (SMM, SMCCA, MEX-SIAM, AME y SMIO)". Al ser un evento híbrido, todas las actividades tanto presenciales como virtuales se trasmitieron en línea vía Zoom y por https://www.facebook.com/FCAeI, salvo la exposición de los carteles que se realizó en modalidad presencial. Las actividades virtuales también se proyectaron en la sala para los participantes presenciales.

En cuanto a los 10 cursos cortos, estos fueron de diferentes niveles: básico, intermedio y avanzado, y fueron impartidos por profesores-investigadores, de diferentes Instituciones de Educación Superior y Centros de Investigación del País. Se ofrecieron para alumnos de licenciatura y posgrado, así como a profesores y profesionales interesados, de distintas instituciones educativas del país y del extranjero, y con tópicos específicos, los cuales generalmente no son abordados con regularidad en sus instituciones. El nombre de los cursos impartidos, así como la asistencia promedio en cada uno de ellos se presenta en la tabla 0.1.

Tabla 0.1: Cursos impartidos en la XXXI ENOAN.

CURSOS NIVEL BÁSICO:	**ASISTENCIA PROMEDIO**
Introducción a las Finanzas. Dr. Gilberto Calvillo Vives, UCIM-UNAM.	34
Desafía tu mente y resuelve sudokus usando programación de restricciones Dr. Jonás Velasco Álvarez, CIMAT, Aguascalientes.	15
Curso de "finanzas personales". Dra. Luz Stella Vallejo Trujillo, Valle-INTEP, Colombia.	19
CURSOS NIVEL INTERMEDIO:	**ASISTENCIA PROMEDIO**
Análisis de redes sociales mediante técnicas de inteligencia artificial. Dr. José Alberto Hernández Aguilar, FCAeI-UAEM.	21
Plataformas de información y análisis financiero y bursátil para el análisis e investigación en el área de mercados financieros. Dr. Rogelio Ladrón de Guevara Cortés, UV.	12
Cómputo Científico con Python. Dr. Gerardo Tinoco Guerrero, UMSNH.	11
Introducción a las redes neuronales y aprendizaje profundo. Dra. Lorena Díaz González, UAEM. M.C. Alida Esmeralda Zarate Jiménez, UAEM. M.C. Oscar Alejandro Uscanga Junco, UAEM. M.O.C.A. Edna Cruz Flores, UAEM.	34
CURSOS NIVEL AVANZADO:	**ASISTENCIA PROMEDIO**
Métodos Variacionales para Determinación de Parámetros en Ecuaciones Diferenciales. Dr. Lorenzo Héctor Juárez Valencia, UAM-Iztapalapa.	29
Introducción al Deep Learning para Finanzas. Dr. José Alberto Guzmán Torres, UMSNH.	45
Diferencias finitas de alto orden de precisión para aproximar derivadas. Dr. Reymundo Ariel Itzá Balam, CONAHCYT-CIMAT Unidad Mérida, Dr. Miguel Ángel Uh Zapata, CONAHCYT-CIMAT Unidad Mérida,	26

En cuanto a las 6 conferencias plenarias, la conferencia inaugural "Diego Bricio", estuvo a cargo del reconocido Profesor-Investigador, el Dr. Luis Javier Álvarez Noguera, investigador del Laboratorio de Simulación de la Unidad Cuernavaca del Instituto de Matemáticas UNAM, la conferencia "Cátedra Humberto

Madrid" estuvo a cargo del reconocido Profesor-Investigador, el Dr. Jesús López Estrada, de la Facultad de Ciencias de la Universidad Nacional Autónoma de México y las cuatro conferencias restantes estuvieron a cargo de los reconocidos profesores investigadores: el Dr. Antonio Fernando Sarmiento Galán de la UCIM, la Dra. María Victoria Chávez Hernández de FCFM-UANL, el Dr. Enrique Covarrubias Jaramillo, Director de Estrategia y Economía de Actinver, y el Dr. Benito M. Chen Charpentier, de la University of Texas at Arlington. El título y resumen de las conferencias se puede consultar en la página de la sociedad en la liga http://www.smcca.org.mx/Productos_y_Resultados_2023.

La presentación de trabajos en forma de cartel se realizó a través de la exposición de 15 carteles sometidos y aprobados por la comisión de evaluación de trabajos, uno en modalidad virtual expuesto en la red social https://www.facebook.com/SMCCA.org.mx y 14 en modalidad presencial. Actualmente se puede acceder a todos ellos en http://www.smcca.org.mx/Productos_y_Resultados_2023. La presentación de los 32 trabajos en forma de comunicación oral, agrupados en 6 sesiones temáticas, se llevó a cabo en modalidad híbrida y transmitidas a través de las aulas virtuales de la plataforma Zoom. Los títulos y resúmenes de los trabajos, así como la información de los ponentes y la grabación de la ponencias se pueden consultar en http://www.smcca.org.mx/Productos_y_Resultados_2023.

Las conferencias invitadas de la Escuela, estuvieron a cargo de el Dr. José Carlos Gómez Larrañaga, Director de la Unidad Mérida del CIMAT, el Dr. Aubin Arroyo Camacho, Jefe de la Unidad Cuernavaca del Instituto de Matemáticas UNAM, el Dr. Caleb Erubiel Andrade Sernas, investigador de Kantar México, el Dr. Gerardo Hernández Dueñas, investigador de la Unidad Juriquilla del Instituto de Matemáticas UNAM, el Dr. Mario Alberto Abarca Sotelo, investigador de Kantar México, la Dra. Elizabeth Santiago del Angel, investigadora del INER, la Dra. Lorena Díaz González, investigadora del CInC de la UAEM y el Dr. José Jacobo Oliveros Oliveros, profesor investigador de la FCFM-BUAP. El título y resumen de las conferencias así como las grabaciones de las mismas se puede consultar en la página de la sociedad en http://www.smcca.org.mx/Productos_y_Resultados_2023.

En cuanto a la sesión especial "IV Mini-Simposium de Medicina y Matemáticas", coordinada por el Dr. Gilberto Calvillo Vives y el Dr. Jesús López Estrada, se contó con la participación de el Dr. Rogelio Danis Lozano, Director del CRISP/INSP de Tapachula, Chiapas, la Dra. Graciela María de los Dolores González Farías, investigadora del CIMAT Monterrey, la Dra. Adriana Monroy Guzmán, investigadora del Hospital General de México de la UNAM, el Dr. Cruz Vargas de León, investigador de la División de Investigación del Hospital Juárez de México, el Dr. Juan Alberto Nader Kawachi, investigador del Servicio de Neurología, Médica Sur, y el Dr. Marco Arieli Herrera Valdez, profesor investigador de la Facultad de Ciencias de la UNAM. El programa de esta sesión se puede consultar en http://www.smcca.org.mx/Productos_y_Resultados_2023.

En la sesión especial de "Finanzas", coordinada por el Dr. Erick Treviño Aguilar y el Dr. Gilberto Calvillo Vives, se contó con la participación de la Dra. Myriam Cisneros Molina del INFONAVIT, el Dr. Carlos Rodríguez Contreras del IIMAS-UNAM, el Dr. Daniel Hernández Hernández del CIMAT, el Dr. José Carlos Méndez de la Torre del SAP, el Dr. Carlos Alberto Torres Martínez de la UACM, la Dra. Dulce Rocío Garnica Jácome, Gerente de Investigación Crédito al Consumo y Productivo (COPPEL), el Dr. Carlos Segovia González de la Unidad Oaxaca del Instituto de Matemáticas UNAM, el Lic. Yeudiel Lara Moreno del IIMAS-UNAM, y la conferencia plenaria de el Dr. Enrique Covarrubias Jaramillo, Director de Estrategia y Economía de Actinver. Los títulos y resúmenes de las conferencias así como las grabaciones de las mismas se pueden consultar en la página http://www.smcca.org.mx/Productos_y_Resultados_2023.

En la sesión especial de "II Foro Conjunto de Sociedades (SMCCA, SMM, SMIO, AME, MEX-SIAM)", coordinada por el Dr. Justino Alavez Ramírez y la Dra. Rina Betzabeth Ojeda Castañeda de la SMCCA, se contó con la participación de conferencistas de cinco sociedades: la Dra. Nancy Maribel Arratia Martínez de la Sociedad Matemática Mexicana, el Dr. Rafael Bernardo Carmona Benítez de la Sociedad Mexicana de Investigación de Operaciones, la Dra. Graciela del Socorro Herrera Zamarrón de SIAM Sección México, el Dr. José Andrés Christen García de la Asociación Mexicana de Estadística, y la conferencia plenaria de el Dr. Benito M. Chen Charpentier de la SMCCA. Los títulos y resúmenes de las conferencias así como las grabaciones de las mismas se pueden consultar en la página http://www.smcca.org.mx/Productos_y_Resultados_2023.

Ganadores del vigésimo primer Premio Mixbaal

Desde 2002, dentro del marco de la ENOAN, se instituyó el "Premio Mixbaal a la Mejor Tesis de Licenciatura en Matemáticas Aplicadas", cuya convocatoria está dirigida a los egresados de carreras de matemáticas y áreas afines. Los ganadores de la vigésima primera edición de este premio, evaluado en la ENOAN 2023, son los siguientes:

Ganadora del premio Mixbaal

- **Vanessa Itzel Soulé Flores**
  Lic. en Matemáticas
  Facultad de Ciencias, Universidad Nacional Autónoma de México
- **Trabajo:** Influencia de las comorbilidades y factores socioeconómicos en el riesgo de fallecer por COVID-19 en México.
- **Director:** Dr. Carlos Erwin Rodríguez Hernández-Vela.

Mención Honorífica

- **Román Alberto Vélez Jiménez**
  Lic. en Matemáticas Aplicadas
  Instituto Tecnológico Autónomo de México
- **Trabajo:** Apuestas Óptimas.
- **Director:** Dr. Alfredo Garbuno Íñigo.

Los ganadores expusieron su trabajo en la ENOAN 2023 como ponentes invitados.

Concurso de Carteles

Continuando con la tradición de las anteriores ediciones de la ENOAN, se realizó el concurso de carteles, en el cual una vez que los participantes exponen ante el jurado y el público su trabajo, el jurado lleva a cabo una evaluación para decidir la premiación. En esta ocasión, al ser una edición híbrida (presencial y virtual), el procedimiento que se estableció fue que los participantes en modalidad presencial expusieron su cartel ante el jurado y el público en forma oral; los participantes en modalidad virtual enviaron su cartel en formato PDF junto con un video grabado de 10 minutos exponiendo su trabajo, mismos que están expuestos en la red social https://www.facebook.com/SMCCA.org.mx. El jurado, integrado por el Dr. Justino Alavez Ramírez (Coordinador), el Dr. Erik Treviño Aguilar y el Dr. Federico Alonso Pecina, revisó cada uno de los videos y los carteles en formato PDF de los participantes en modalidad virtual y analizó la exposición de los participantes en modalidad presencial, y con base en la Convocatoria emitida para tal fin y bajo los Criterios de Evaluación: Calidad Científica, Impacto Visual, Creatividad y Originalidad, los Miembros del Jurado revisaron reservada y libremente la totalidad de los carteles presentados, dictaminando lo siguiente:

Mejor Cartel Nivel Licenciatura	
Título	**Autor(es)-Institución**
Modelo financiero con análisis de retardos y su comportamiento caótico.	Jesús Salinas Gutiérrez Marcos Ángel González Olvera Universidad Autónoma de la Ciudad de México
Mejor Cartel Nivel Maestría	
Título	**Autor(es)-Institución**
Análisis de la dinámica de un modelo SIS con migración en dos regiones.	Itzayana Yisely Madrigal Estrada Universidad Juárez Autónoma de Tabasco
Mejor Cartel Nivel Doctorado	
Título	**Autor(es)-Institución**
Análisis de un modelo para el Problema de Colapso de Colmenas con influencia de un pesticida.	Erika Fabiola Rivero Esquivel María de Lourdes Esteva Peralta Jesús López Estrada Universidad Nacional Autónoma de México

Cobertura de las actividades

De acuerdo al registro oficial del Certificado de Inscripción a la ENOAN 2023, se contó con la asistencia de 182 personas, de las cuales 58 (32 %) pertenecían al género femenino, 122 (67 %) al género masculino y 2 (1 %) seleccionaron la opción de "Prefiero no contestar", éstos resultados se muestran en la gráfica 0.1.

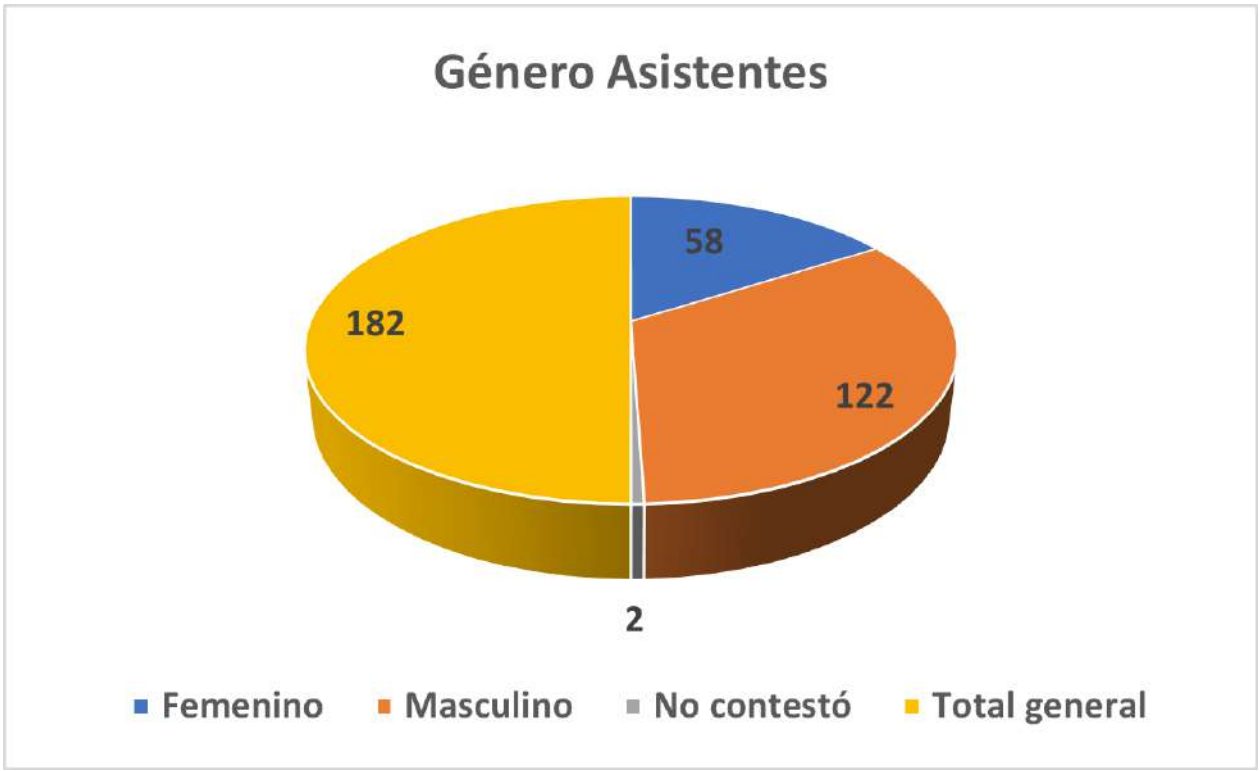


Figura 0.1: Gráfica de número de asistentes por género a la XXXI ENOAN.

Los intervalos de edades que se establecieron en la cédula de registro fueron de los 18 años a más de 64 años, que eran las edades probables de la audiencia para este tipo de evento (jóvenes, adultos y adultos mayores), y los resultados que se obtuvieron con respecto a la edad se presentan en la tabla 0.2 y figura 0.2. Se puede observar en la tabla 0.2 que el mayor número de asistentes se dieron en el rango de 18 a 24 años que fueron 60 (33 %), seguido por el rango de 25 a 34 años que fueron 44 (24.17 %), estando el menor número de asistentes en las edades adultas de 55 a más de 64 años con 20 (11 %). Es importante resaltar que hubo una alta participación de los jóvenes que desde sus primeros semestres de la licenciatura estén decidiendo participar en la ENOAN.

Tabla 0.2: Número y porcentaje de asistentes al evento por intervalo de edades.

Intervalos de edades	**Número de asistentes**	**Porcentaje de asistentes**
18 a 24	60	32,97 %
25 a 34	44	24,17 %
35 a 44	34	18,68 %
45 a 54	17	9,34 %
55 a 64	7	3,85 %
> 64	13	7,14 %
No contestó	7	3,85 %
Total	**182**	**100** %

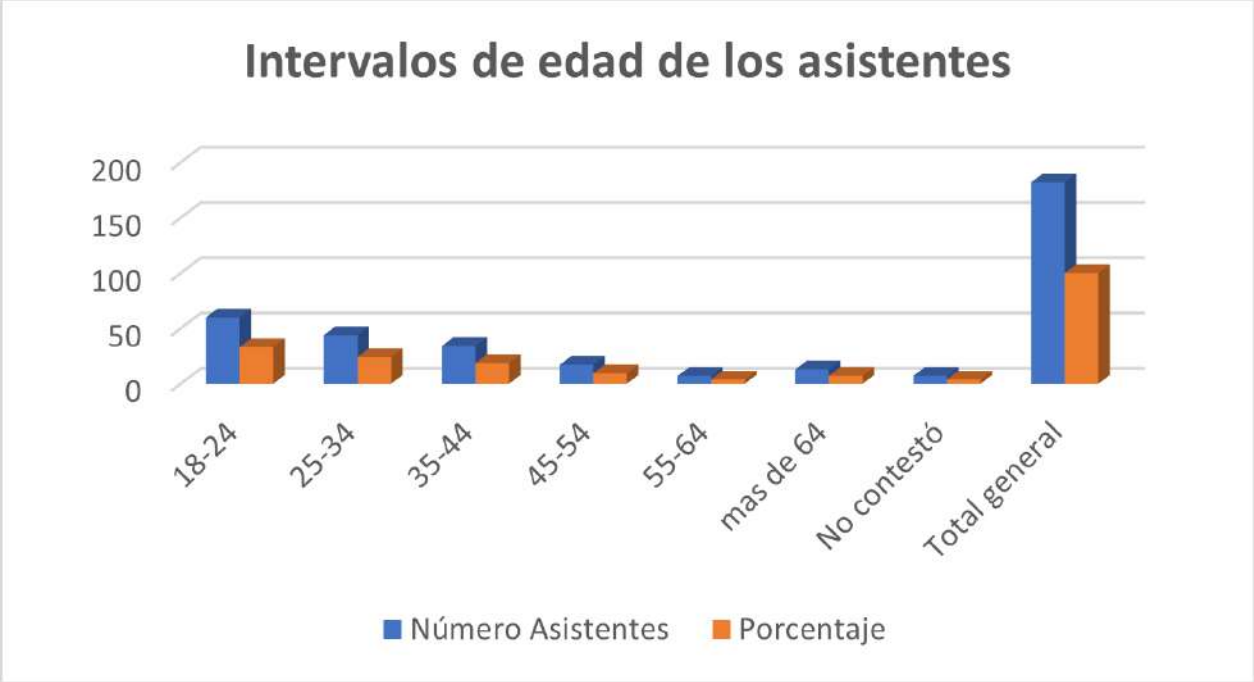


Figura 0.2: Gráfica de intervalos de edades y los porcentajes respectivos de los participantes.

En la tabla 0.3 y figura 0.3 se muestran las frecuencias cruzadas edad-género, que permite ver la relación existente entre estas dos variables. Tanto la tabla como la gráfica muestran que en todos los rangos de edad de los 18 a los 64 años y más, se tuvo un mayor porcentaje de asistencia de hombres que de mujeres, habiendo una diferencia menor de participación entre los géneros en los intervalos de edades de 25 a 34, 55 a 64 y más de 64.

A continuación se muestra en la tabla 0.4 y figura 0.4, los porcentajes de asistentes en cada una de las dos modalidades (presencial y virtual) y en la tabla 0.5 y figura 0.5 se presenta una gráfica cruzada de porcentaje de asistentes que muestra la relación entre el género y la modalidad seleccionada.

Tabla 0.3: Frecuencia cruzada Edad-Género de los asistentes.

Edad Género	**Intervalos de Edades**							
Género	**18-24**	**25-34**	**35-44**	**45-54**	**55-64**	**Más de 64**	**No contestó**	**Total**
Femenino	19	18	10	3	3	3	2	58
Porcentaje	10.4 %	10 %	5.5 %	1.7 %	1.7 %	1.7 %	1 %	32 %
Masculino	41	25	24	14	4	9	5	122
Porcentaje	22.5 %	13.7 %	13.2 %	7.7 %	2.2 %	5 %	2.7 %	67 %
No contestó	0	1	0	0	0	1	0	2
Porcentaje	0 %	0.5 %	0 %	0 %	0 %	0.5 %	0 %	1 %
Total	**60**	**44**	**34**	**17**	**7**	**13**	**7**	**182**
Porcentaje	**32.9 %**	**24.2 %**	**18.7 %**	**9.4 %**	**3.9 %**	**7.2 %**	**3.7 %**	**100 %**

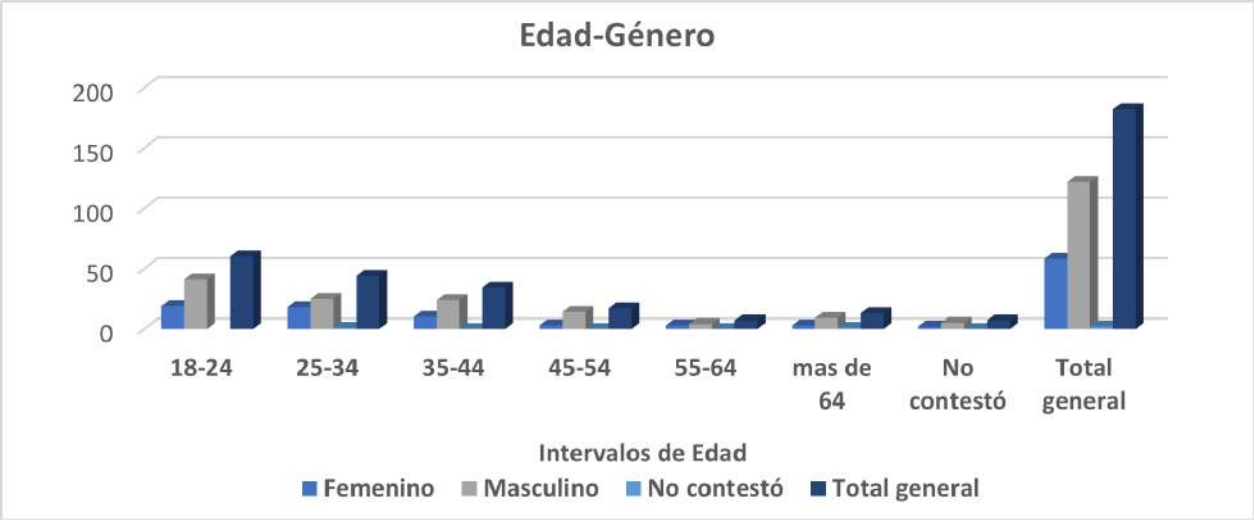


Figura 0.3: Gráfica relación entre Género y Edad por intervalo de edades.

Tabla 0.4: Número y porcentaje de asistentes en cada modalidad.

Modalidad	**Número**	**Porcentaje**
Presencial	129	70.9 %
Virtual	52	28.6 %
No contestó	1	0.5 %
Total	**182**	**100 %**

Como se puede visualizar tanto en la tabla 0.4 como en la gráfica 0.4, que en esta edición de la ENOAN 2023, hubo una mayor participación en modalidad presencial (129, 70.9 %) que en la virtual (52, 28.6 %), solamente una persona "No contestó" (1, 0.5 %). El incremento en la modalidad presencial en esta edición se debió a la disminución de las restricciones en cuanto a movilidad e interrelación de las personas que fueron establecidas durante la pandemia del COVID-19 y a la vuelta a la "normalidad". Tanto la tabla 0.5 como la gráfica 0.5 muestran que en ambos géneros, femenino y masculino, se prefirió la participación en la modalidad presencial.

Se obtuvieron otros estadísticos relacionados con características de los asistentes como: país y estado de nacimiento, pertenencia a algún grupo étnico, entre otros. La información completa de las estadísticas de la asistencia se puede consultar en https://www.smcca.org.mx/Boletínes/Estadisticas-Asistencia-ENOAN-2023.pdf.

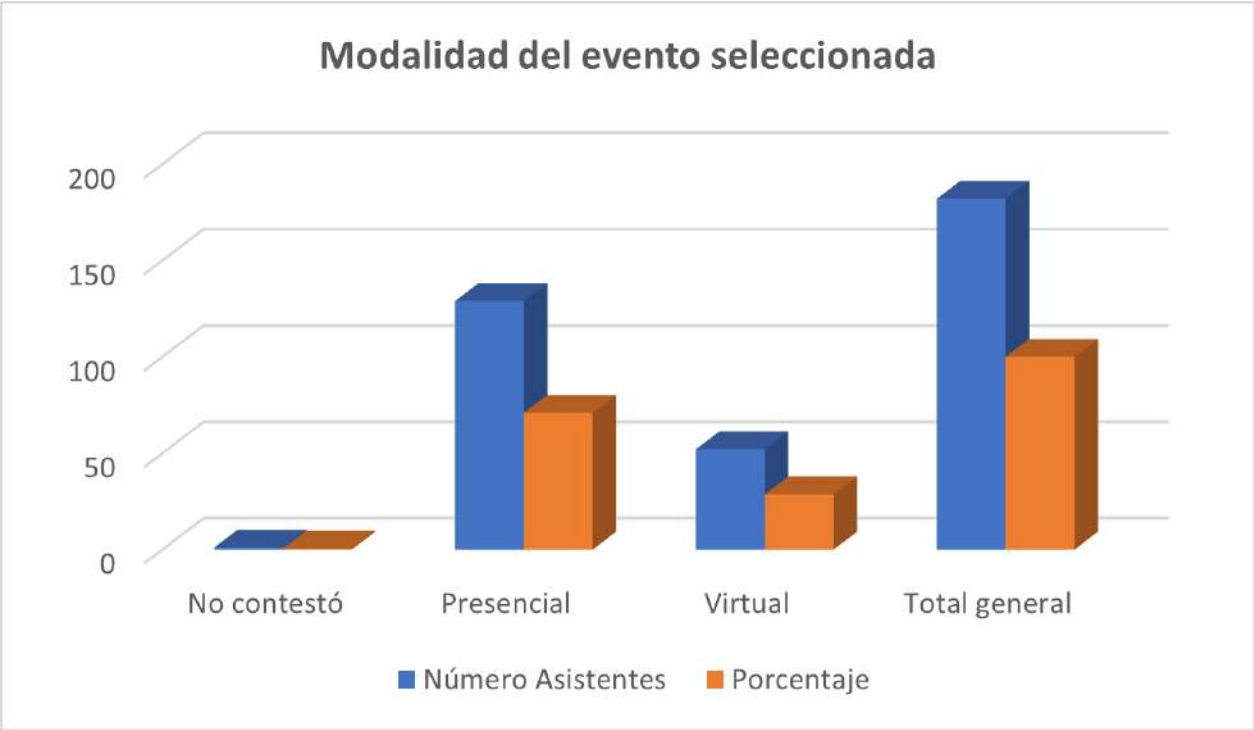


Figura 0.4: Gráfica de número y porcentaje de selección de modalidad de los asistentes.

Tabla 0.5: Frecuencia cruzada Género-Modalidad de los asistentes.

	Género			
Modalidad	**Femenino**	**Masculino**	**No contestó**	**Total**
Presencial	41	87	1	129
Porcentaje	22.6 %	47.8 %	0.5 %	70.9 %
Virtual	17	34	1	52
Porcentaje	9.4 %	18.7 %	0.5 %	28.6 %
No contestó	0	1	0	1
Porcentaje	0 %	0.5 %	0 %	0.5 %
Total	**58**	**122**	**2**	**182**
Porcentaje	**32 %**	**67 %**	**1 %**	**100 %**

Consideraciones finales

Por último es de gran importancia señalar que el gran esfuerzo de trabajo realizado tanto por el Comité Local (Unidad Cuernavaca del Instituto de Matemáticas UNAM, Facultad de Contaduría, Administración e Informática y el Centro de Investigación en Ciencias de la Universidad Autónoma del Estado de Morelos) como el Comité Nacional (aglutinados dentro de la SMCCA) en la organización, y contando con el importante apoyo financiero de Instituciones, Dependencias y Centros de Investigación como: CONAHCYT, la Sociedad Mexicana de Computación Científica y sus Aplicaciones, A.C., la Unidad Cuernavaca del Instituto de Matemáticas UNAM, la Facultad de Contaduría, Administración e Informática de la Universidad Autónoma del Estado de Morelos, la Universidad Autónoma Metropolitana Unidad Iztapalapa, la Universidad Michoacana de San Nicolás de Hidalgo, la SIAM Sección México (MexSIAM), la Sociedad Matemática Mexicana (SMM), la Sociedad Mexicana de Investigación de Operaciones (SMIO) y la Sociedad Mexicana de Estadística (AME); permitió obtener un conjunto de resultados a beneficio de una comunidad científica conformada por alumnos, profesores, investigadores y profesionales interesados en la Computación Científica y las Matemáticas Aplicadas, que incidieron en indicadores de impacto como los que se presentan en la tabla 0.6.

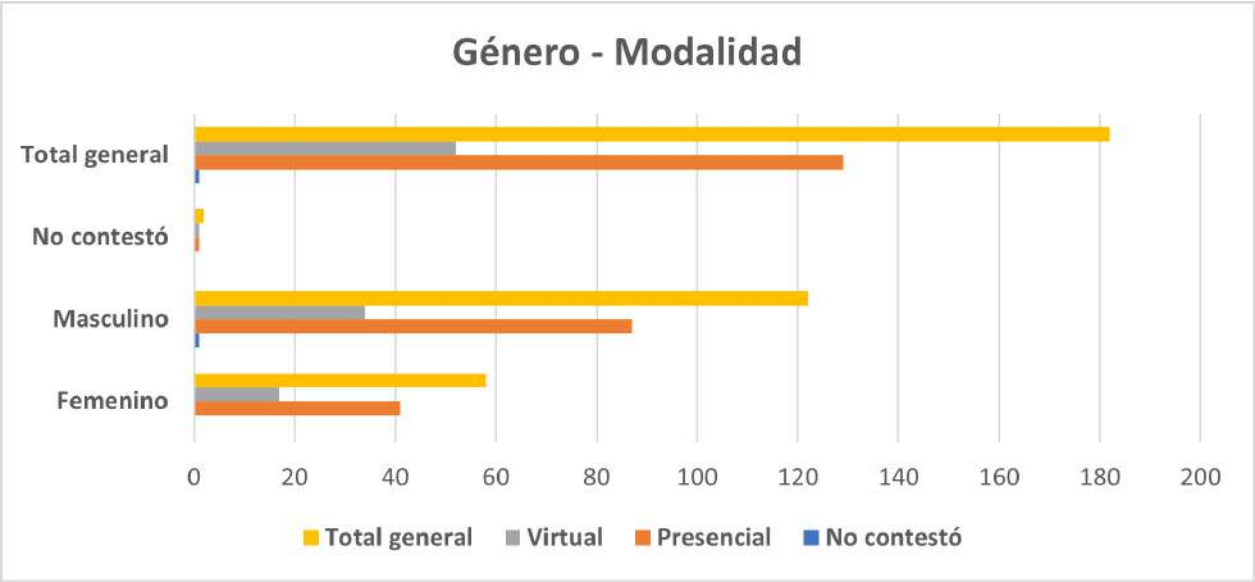


Figura 0.5: Gráfica cruzada Género-Modalidad de los asistentes al evento.

Tabla 0.6: Indicadores de impacto.

Indicador	**Cantidad**
Total de asistentes registrados:	119
Programas Académicos beneficiados por el evento:	28
Cuerpos Académicos o Grupos de Investigación beneficiados:	15
Cursos cortos impartidos:	10
Conferencias (Diego Bricio, Humberto Madrid, Plenarias e Invitadas):	37
Ponencias por solicitud:	32
Carteles expuestos:	15
Estudiantes beneficiados:	109
Investigadores y Docentes beneficiados:	73
Mujeres beneficiadas:	58
Hombres beneficiados:	122
Número de Instituciones Nacionales participantes:	41
Número de Instituciones Internacionales participantes:	9
Integrantes del Comité Nacional:	9
Integrantes del Comité Local:	8
Personal de Apoyo Sede:	9
Auditorio para inauguración, conferencias plenarias e invitadas:	2
Salas de cómputo equipados con cañón proyector para cursos:	4
Salones equipados con cañón proyector y pizarrón para ponencias:	3
Mamparas para carteles:	7
Licencias de uso de aulas virtuales de la plataforma ZOOM:	2

Finalmente, se muestran en la figura 0.6 algunas imágenes de las actividades desarrolladas por los participantes de la ENOAN 2023.

Figura 0.6: Algunas imágenes de las actividades de la ENOAN 2023.

Artículos

General foundations of high-order immersed interface methods to solve interface problems

Heidy Escamilla Puc[1], Reymundo Itzá Balam[2,3], and Miguel Uh Zapata[*2,3]

[1]Facultad de Matemáticas, Universidad Autónoma de Yucatán; Anillo Periférico Norte, Tablaje Cat. 13615, Colonia Chuburná Hidalgo Inn, Mérida, México
[2]Consejo Nacional de Humanidades, Ciencias y Tecnologías, CONAHCYT, México
[3]Centro de Investigación en Matemáticas A. C., CIMAT Unidad Mérida, México

Abstract

This work forms the foundation for addressing high-order immersed interface methods to solve interface problems and enables us to conduct in-depth examination of this theory. Here, we focus on the introduction a fourth-order finite-difference formulation to approximate the second-order derivative of discontinuous functions. The approach is based on the combination of a high-order implicit formulation and the immersed interface method. The idea is to modify the standard schemes by introducing additional contribution terms based on jump conditions. These contributions are calculated only at grid points where the stencil intersects with the interface. Here, we discuss the issues of implementing the one-dimensional Poisson equation and the heat conduction equation with discontinuous solutions as a three-point stencil for each grid point on the computational domain. In both cases, the resulting discretization approach yields a tridiagonal linear system with matrix coefficients identical to those employed for smooth solutions. We present several numerical experiments to verify the feasibility and accuracy of the method. Thus, this high-order method provides an attractive numerical framework that can efficiently lead to the solution to more complex problems.

Keywords: Immersed interface method; Implicit finite difference; Fourth-order accuracy; Poisson equation; Heat conduction equation.

1 Introduction

High-order numerical solutions to differential equations arising from discontinuous solutions find extensive utility across various research domains [5, 26, 16, 13, 27, 28]. In the case of smooth solutions,

*angeluh@cimat.mx

the standard central finite-difference method requires a significant number of grid points to achieve a high level of accuracy in its numerical results. As a result, over the past few decades, several schemes have been developed to obtain fourth- and sixth-order finite-difference methods [15, 35, 21, 30, 34], including those one based on the implicit finite-difference (IFD) formulation [29, 11, 12, 33].

On the other hand, although several methods have been proposed to address discontinuous problems [26, 24, 18, 23, 20, 2, 10], the immersed interface method (IIM) [14, 31, 1, 25] stands out as a highly accurate option that requires minimal adjustments to the standard finite-difference formulation. However, these methods typically achieve second-order accuracy. For instance, there are limited implementations of a few third-, fourth- and sixth-order IIMs available for solving Poisson equations with discontinuous solutions [9, 8, 17, 37, 36, 7, 22, 4].

This paper focuses on the basic ideas of combining the IFD and IIM to achieve high-order approximations for second-order derivatives of both continuous and discontinuous real-valued functions. The IFD scheme offers a highly accurate numerical method [3, 19], while the IIM handles discontinuities through minimal adjustments made exclusively at grid points where the stencil intersects the interface [32, 6], yielding additional terms known as jump contributions. The resulting method will be named as implicit finite-difference immersed interface method (IFD-IIM).

We illustrate the implementation of a global fourth-order IFD-IIM approach with two examples: the one-dimensional Poisson equation for static cases and the heat conduction equation with a fixed interface for time-evolving scenarios. Our proposed method offers several advantages. Notably, the resulting tridiagonal matrix coefficient of the finite-difference scheme remains the same as those for smooth solutions, with the additional terms arising from the jumps located in the right-hand side vector. Consequently, our algorithm is straightforward to implement, employing the efficient Thomas' algorithm.

We have organized our study as follows. In Section 2, we introduce a fourth-order IFD method capable of handling second-order derivatives, both in smooth and discontinuous scenarios. Section 3 demonstrates the application of this implicit scheme in approximating solutions to the one-dimensional Poisson equation. Section 4 shows the combination of the IFD-IIM with the Crank-Nicolson method to solve the heat conduction equation. Sections 5 and 6 provide a series of numerical examples to illustrate the algorithm's accuracy for both equations. Lastly, Section 7 offers our conclusions and outlines directions for future research.

2 IFD formulation with discontinuities

In this section, we outline the key attributes of the IFD formulation to achieve a global fourth-order method, demonstrating how the scheme can be adapted for addressing discontinuous problems through the utilization of the IIM.

We approximate the numerical solution on the domain $[\mathfrak{a}, \mathfrak{b}]$ that is divided into N sub-intervals using the points x_i, as follows

$$x_i = \mathfrak{a} + (i-1)h, \qquad i = 1, 2, \ldots, N, N+1, \tag{1}$$

where the grid size is given by $h = (\mathfrak{b} - \mathfrak{a})/N$. We employ U_i and $u_i = u(x_i)$ to denote the approximate and exact solution at the ith-point of the grid, respectively. Here, the interface is at $x = x_\alpha$ located between grid points x_I and x_{I+1} as $x_I \leq x_\alpha < x_{I+1}$, see Fig. 1. The distances of the closest grid points to the interface are defined as $h_L = x_I - x_\alpha$ and $h_R = x_{I+1} - x_\alpha$. Note that h_R and h_L are positive and negative values, respectively.

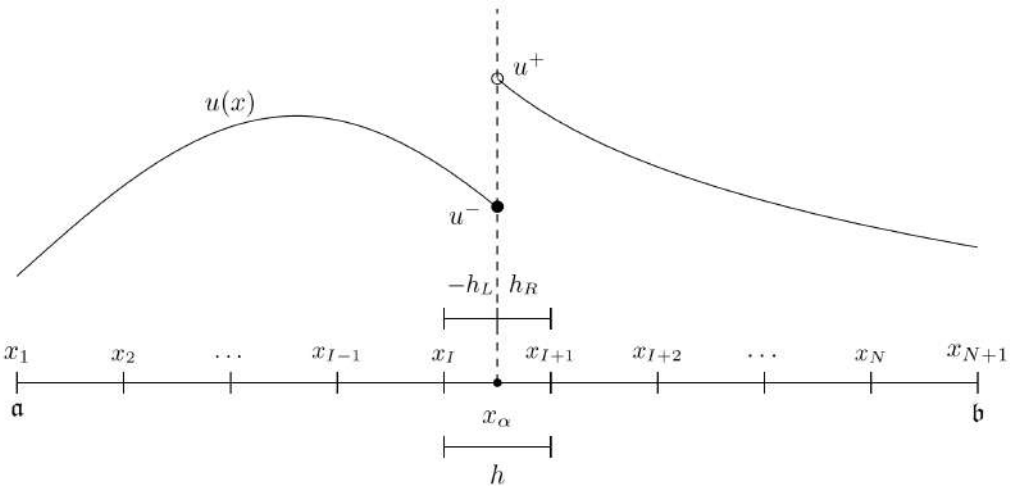


Figure 1: Example of a discontinuous function u with an interface located between the points x_I and x_{I+1}.

On the other hand, the jump for u at x_α is defined as

$$[u] = u^+ - u^-, \quad \text{where} \quad u^+ = u(x_\alpha^+) = \lim_{x \to x_\alpha^+} u(x), \quad \text{and} \quad u^- = u(x_\alpha^-) = \lim_{x \to x_\alpha^-} u(x).$$

We employ a similar definition for the jumps such as the ones of the right-hand side and the derivatives of u.

In this paper, we designate x_I and x_{I+1} as *irregular points*, while considering the rest as *regular points*. This classification holds significant importance as distinct schemes are applied to each category, as presented in the following theorems.

2.1 Second-order derivative approximation for regular and irregular grid points

The following two Theorems state the main results to approximate the second-order derivative using high-order schemes for regular and irregular points.

Theorem 2.1. ***Regular points*** *[3, 19]. Let us consider a real-valued function u with an interface x_α such that $x_I \leq x_\alpha < x_{I+1}$. Then u_{xx} can be approximated by a fourth-order IFD scheme*

$$\mathfrak{D}^2 u_{xx}\big|_i = \delta^2 u\big|_i + O(h^4), \quad i \neq I, I+1, \tag{2}$$

where

$$\mathfrak{D}^2(\cdot) := (\cdot) + bh^2(\cdot)_{xx}, \tag{3}$$

$b = 1/12$, and the central finite-difference formula δ^2 is given by

$$\delta^2 u\big|_i := \frac{1}{h^2}\left\{u_{i-1} - 2u_i + u_{i+1}\right\}. \tag{4}$$

Proof. From Taylor series expansions and under some simplifications, the second-order derivative at any regular point x_i can be written in terms of the centered finite-difference operator, as follows

$$u_{xx}|_i = \frac{1}{h^2}\left\{u_{i-1} - 2u_i + u_{i+1}\right\} - \frac{1}{12}h^2\, u_{xxxx}|_i + O(h^4),$$

Previous equation holds due to the solution is smooth on a neighborhood around x_i. Thus, we obtain a fourth-order IFD using a stencil of three nodes by moving the fourth-order term to the left-hand side. We get

$$u_{xx}|_i + \frac{1}{12}h^2\,(u_{xx})_{xx}|_i = \frac{1}{h^2}\left\{u_{i-1} - 2u_i + u_{i+1}\right\} + O(h^4).$$

Finally, the proof is completed by using the definition of operator $\mathfrak{D}^2$, δ^2, and b. □

It is important to remark that formula (2) is applicable exclusively for regular points. In order to address this limitation for the two irregular points near the interface, we introduce a modified IFD scheme using the IIM, specifically tailored to handle discontinuous solutions. Furthermore, instead of having a fourth-order local truncation error for the irregular points, we proceed as other IIMs [14, 31, 1, 37, 4, 22] by taking one order lower at these points. We will numerically show that the global order of convergence can be still $O(h^4)$ even if the local truncation error at $i = I$ and $i = I+1$ is $O(h^3)$.

Theorem 2.2. ***Irregular points*** *[12]. Let us consider the known jump conditions*

$$[u], \quad [u_x], \quad [u_{xx}], \quad [u_{xxx}], \quad [u_{xxxx}], \tag{5}$$

at x_α such that $x_I \leq x_\alpha < x_{I+1}$. Then u_{xx} can be approximated at x_I and x_{I+1} by the IFD-IIM given by

$$\mathfrak{D}^2 u_{xx}\,|_i = \delta^2 u\big|_i + C_i + O(h^3), \quad i = I, I+1, \tag{6}$$

where $\mathfrak{D}^2$ and δ^2 are defined in (3) *and* (4)*, respectively, and*

$$C_i = \begin{cases} -\dfrac{1}{h^2}\left\{[u] + h_R\,[u_x] + \dfrac{h_R^2}{2}\,[u_{xx}] + b\left(2h_R^3\,[u_{xxx}] + \dfrac{h_R^4}{2}\,[u_{xxxx}]\right)\right\}, & i = I, \\ \dfrac{1}{h^2}\left\{[u] + h_L\,[u_x] + \dfrac{h_L^2}{2}\,[u_{xx}] + b\left(2h_L^3\,[u_{xxx}] + \dfrac{h_L^4}{2}\,[u_{xxxx}]\right)\right\}, & i = I+1, \end{cases} \tag{7}$$

and where $b = 1/12$.

Proof. We obtain a third-order scheme for at x_I and x_{I+1} following similar ideas as the ones developed for the generalized Taylor expansion proposed by Xu & Wang [32] and the IIM for elliptic interface problems with straight interfaces proposed by Feng & Li [6]. The idea is to consider extended smooth solutions of u such that we can apply the standard central scheme to x_I and x_{I+1}. For instance, a function based on the original left solution is defined as

$$u_\ell(x) = \begin{cases} u(x), & x \leq x_\alpha, \\ u^- + h_R u_x^- + \dfrac{h_R^2}{2} u_{xx}^- + \dfrac{h_R^3}{6} u_{xxx}^- + \dfrac{h_R^4}{24} u_{xxxx}^-, & x_\alpha < x, \end{cases}$$

Using Taylor series expansions of u_{I+1} around x_α, the definition of jumps (5), and some simplification yield

$$\delta^2 u\big|_I = \delta^2 u_\ell\big|_I + \frac{1}{h^2}\left([u] + h_R [u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{6}[u_{xxx}] + \frac{h_R^4}{24}[u_{xxxx}] \right) + O(h^3).$$

Thus,

$$\begin{aligned} \delta^2 u\big|_I &= (u_\ell)_{xx}|_I + bh^2 \, (u_\ell)_{xxxx}|_I \\ &\quad + \frac{1}{h^2}\left([u] + h_R [u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{6}[u_{xxx}] + \frac{h_R^4}{24}[u_{xxxx}] \right) + O(h^3). \end{aligned}$$

Finally, we get (6) which complete the proof. The same procedure can be applied for the proof at x_{I+1}. We refer to the reader to the work of Itza Balam and Uh Zapata [12] for more details. □

Remark. *If $b = 0$, then Theorem 2.1 yields the standard second-order finite-difference method for regular points given by*

$$u_{xx}|_i = \delta^2 u\big|_i + O(h^2), \quad i \neq I, I+1, \tag{8}$$

and Theorem 2.1 results in an IIM of first-order for irregular points $i = I, I+1$ as follows

$$u_{xx}|_i = \delta^2 u\big|_i + c_i + O(h), \quad i = I, I+1, \tag{9}$$

where

$$c_i = \begin{cases} -\dfrac{1}{h^2}\left([u] + h_R [u_x] + \dfrac{h_R^2}{2}[u_{xx}] \right), & i = I, \\ \dfrac{1}{h^2}\left([u] + h_L [u_x] + \dfrac{h_L^2}{2}[u_{xx}] \right), & i = I+1. \end{cases} \tag{10}$$

Note that, in this case, we only require to explicitly know jump conditions $[u]$, $[u_x]$ and $[u_{xx}]$.

2.2 Implicit approximation for a real-valued function

Before applying previous results to approximate differential equations, it would be useful to express the operator $\mathfrak{D}^2 u$ using finite differences for a real-valued function u rather than its second-order derivative u_{xx}. The finite-difference formula is obtained by approximating $u_{xx}|$ with the central finite differences, as presented in (8)-(10) for regular and irregular points, respectively.

For regular points ($i \neq I, I+1$), it follows from equation (8):

$$\mathfrak{D}^2 u\big|_i \;=\; u_i + bh^2 u_{xx}\big|_i \;=\; u_i + bh^2\left\{\delta^2 u\big|_i + O(h^2)\right\} \;=\; bu_{i-1} + (1-2b)u_i + bu_{i+1} + O(h^4).$$

Thus, if we define as $\mathfrak{d}^2$ the resulting finite-difference

$$\mathfrak{d}^2 u\big|_i \;:=\; bu_{i-1} + (1-2b)u_i + bu_{i+1}, \tag{11}$$

then, we have that

$$\mathfrak{D}^2 u\big|_i = \mathfrak{d}^2 u\big|_i + O(h^4), \quad i \neq I, I+1. \tag{12}$$

However, we still have the same issue to overcome for a discontinuous function u. We remark that this second-order derivative of u is already multiplied for h^2. Consequently, we can use the IIM technique where the contribution term is only first-order accurate to u; thus, we still have a local $O(h^3)$ to keep a global fourth-order accurate method. Then for irregular points, the IIM applied for this term follows

$$\mathfrak{D}^2 u\big|_i = \mathfrak{d}^2 u\big|_i + (c_u)_i + O(h^3), \quad i = I, I+1, \tag{13}$$

where $\mathfrak{d}^2$ is given by finite difference (11) and

$$(c_u)_i = \begin{cases} -b\left([u] + h_R\,[u_x] + \dfrac{h_R^2}{2}\,[u_{xx}]\right), & i = I, \\ \;\; b\left([u] + h_L\,[u_x] + \dfrac{h_L^2}{2}\,[u_{xx}]\right), & i = I+1. \end{cases} \tag{14}$$

Finally, for regular and irregular points, we have high-order finite-difference approximations of the implicit operator $\mathfrak{D}^2$ applied to a real-valued function u as in (12) and (13), and to its second-order derivative u_{xx} as in (2) and (6). Now, we proceed to implement them to the solution of differential equations, as presented in the following two sections.

3 Poisson equation

In this section, we develop a fourth-order finite difference scheme for the Poisson equation. Let us consider u and f as the solution of the problem and known right-hand side function, respectively.

Thus the interface problem is given by

$$\begin{aligned} \Delta u(\boldsymbol{x}) &= f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega \setminus \Gamma, \quad \Omega = \Omega^+ \cup \Omega^-, \\ u(\boldsymbol{x}) &= g(\boldsymbol{x}), \quad \boldsymbol{x} \in \partial\Omega, \\ [u]_\Gamma &= \mathfrak{p}(\boldsymbol{x}), \quad \boldsymbol{x} \in \Gamma, \\ [u_{\boldsymbol{n}}]_\Gamma &= \mathfrak{q}(\boldsymbol{x}), \quad \boldsymbol{x} \in \Gamma. \end{aligned}$$

We divide Ω in two regions, Ω^+ and Ω^-, separated by an immersed interface Γ. Dirichlet boundary conditions are defined on $\partial\Omega$ through function g. We assume that u and f may have discontinuities at the interface Γ. Thus, we require additional conditions $[u]_\Gamma$ and $[u_{\boldsymbol{n}}]_\Gamma$ known as principal jump conditions. Here, $u_{\boldsymbol{n}}$ is the derivative in the normal direction. Note that the principal jump conditions, $\mathfrak{p}$ and $\mathfrak{q}$, are known functions defined on Γ.

In the context of the general problem, the computational domain can be considered into multiple dimensions. Nevertheless, since the primary objective of this paper is to illustrate the fundamental attributes of the proposed implicit high-order method, we concentrate on investigating the one-dimensional (1D) interface problem as defined by

$$u_{xx}(x) = f(x), \quad x \in (\mathfrak{a}, x_\alpha) \cup (x_\alpha, \mathfrak{b}), \tag{15}$$

$$u(x) = g(x), \quad x \in \{\mathfrak{a}, \mathfrak{b}\}, \tag{16}$$

$$[u] = \mathfrak{p}, \tag{17}$$

$$[u_x] = \mathfrak{q}, \tag{18}$$

Here, u and f can be discontinuous functions at a given fixed point $x = x_\alpha$, and principal jump conditions $[u] = [u]_{x_\alpha}$ and $[u_x] = [u_x]_{x_\alpha}$ are known values at x_α.

3.1 IFD-IIM for the 1D Poisson problem

Let us consider that x_α is located between the adjacent grid points x_I and x_{I+1} as $x_I \leq x_\alpha < x_{I+1}$. If we apply operator $\mathfrak{D}^2(\cdot) = (\cdot) + bh^2(\cdot)_{xx}$ at both sides of (15), then we get

$$\mathfrak{D}^2 u_{xx}\big|_i = \mathfrak{D}^2 f\big|_i, \qquad i = 2, \ldots, N. \tag{19}$$

For $i = 1$ and $i = N+1$, the grid points are at the boundary, thus the Dirichlet boundary condition can be directly applied. Thus, using the IFD scheme (2) and approximation (12) in (19) at x_i, we have that the finite-difference scheme for regular points is given by

$$\delta^2 u\big|_i = \mathfrak{d}^2 f\big|_i + O(h^4), \qquad i \neq I, I+1. \tag{20}$$

Similarly, using formulas (6) and (13), the scheme for irregular points is given by

$$\delta^2 u\big|_i + C_i = \mathfrak{d}^2 f\big|_i + (c_f)_i + O(h^3), \qquad i = I, I+1, \tag{21}$$

where C_i and $(c_f)_i$ are given by (7) and (14), respectively. Thus, combining the results for all grid points, the IFD-IIM for the 1D Poisson equation (15) at x_i is given by

$$\frac{1}{h^2}U_{i-1} - \frac{2}{h^2}U_i + \frac{1}{h^2}U_{i+1} = bf_{i-1} + (1-2b)f_i + bf_{i+1} + \mathfrak{C}_i, \quad i = 2, \ldots, N, \tag{22}$$

where total contribution is $\mathfrak{C}_i = (c_f)_i - C_i$ with

$$C_i = \begin{cases} -\dfrac{1}{h^2}\left([u] + h_R[u_x] + \dfrac{h_R^2}{2}[u_{xx}] + b\left(2h_R^3[u_{xxx}] + \dfrac{h_R^4}{2}[u_{xxxx}]\right)\right), & i = I, \\ \dfrac{1}{h^2}\left([u] + h_L[u_x] + \dfrac{h_L^2}{2}[u_{xx}] + b\left(2h_L^3[u_{xxx}] + \dfrac{h_L^4}{2}[u_{xxxx}]\right)\right), & i = I+1, \\ 0, & \text{else where,} \end{cases} \tag{23}$$

and

$$(c_f)_i = \begin{cases} -b\left([f] + h_R[f_x] + \dfrac{h_R^2}{2}[f_{xx}]\right), & i = I, \\ b\left([f] + h_L[f_x] + \dfrac{h_L^2}{2}[f_{xx}]\right), & i = I+1, \\ 0, & \text{else where.} \end{cases} \tag{24}$$

We remark that scheme (22)-(24) results in an approximation with local truncation error of fourth- and third-order for regular and irregular grid points, respectively. Thus, a global $O(h^4)$ method is expected.

3.2 IFD-IIM and principal jump conditions

Note that $[u]$, $[u_x]$, $[u_{xx}]$, $[u_{xxx}]$, and $[u_{xxxx}]$ must be known to apply the proposal fourth-order IFD-IIM. Thus, it seems that more jump conditions of u rather than the principal jump conditions (17) and (18) are required to have a fourth-order accurate method. However, we can use the Poisson equation (15) to obtain them as follows

$$[u_{xx}] = [f], \qquad [u_{xxx}] = [f_x], \qquad [u_{xxxx}] = [f_{xx}]. \tag{25}$$

Thus, the total jump contribution for the one-dimensional Poisson problem is given by

$$\mathfrak{C}_i = \begin{cases} \dfrac{1}{h^2}\left([u] + h_R[u_x] + \dfrac{h_R^2}{2}[f]\right) \\ \quad -b\left\{[f] + h_R\left(-\dfrac{2h_R^2}{h^2} + 1\right)[f_x] + \dfrac{h_R^2}{2}\left(-\dfrac{h_R^2}{h^2} + 1\right)[f_{xx}]\right\}, & i = I, \\ -\dfrac{1}{h^2}\left([u] + h_L[u_x] + \dfrac{h_L^2}{2}[f]\right) \\ \quad +b\left\{[f] + h_L\left(-\dfrac{2h_L^2}{h^2} + 1\right)[f_x] + \dfrac{h_L^2}{2}\left(-\dfrac{h_L^2}{h^2} + 1\right)[f_{xx}]\right\}, & i = I+1, \\ 0, & \text{else where.} \end{cases} \tag{26}$$

Thus contribution $\mathfrak{C}_i$ depends only on the principal jump conditions, $[u]$ and $[u_x]$, and right-hand side jumps: $[f]$, $[f_x]$, and $[f_{xx}]$. The additional jumps derivatives from the right-hand side can be approximated using the current values of f. In this paper, we will assume that we know them.

Remark. *For the 1D Poisson problem, a global second-order IIM ($b = 0$) only requires to know the principal jump conditions $[u]$, $[u_x]$ and $[f]$.*

Remark. *If $h_R/h = 1$, then $h_L = 0$ and both weight terms next to second-order derivative jump of f are equal to zero in (26). Thus, we do not require to know jump condition $[f_{xx}]$ to obtain a fourth-order method when the interface is located at a grid point.*

4 Heat equation with a fixed interface

For the second differential equation, we consider the heat conduction equation with initial, boundary, and principal jump conditions, as follows

$$\begin{aligned} u_t(\boldsymbol{x},t) &= \Delta u(\boldsymbol{x},t) - f(\boldsymbol{x},t), \quad \boldsymbol{x} \in \Omega \setminus \Gamma \times (0,T], \quad \Omega = \Omega^+ \cup \Omega^-, \\ u(\boldsymbol{x},0) &= u_0(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x},t) &= g(\boldsymbol{x},t), \quad (\boldsymbol{x},t) \in \partial\Omega \times [0,T], \\ [u]_\Gamma &= \mathfrak{p}(\boldsymbol{x},t), \quad \boldsymbol{x} \in \Gamma \times [0,T], \\ [u_{\boldsymbol{n}}]_\Gamma &= \mathfrak{q}(\boldsymbol{x},t), \quad \boldsymbol{x} \in \Gamma \times [0,T], \end{aligned}$$

where the source f and initial value u_0 may be discontinuous or singular across a fixed interface Γ. The interface Γ is immersed in the domain Ω and divides into two parts, Ω^+ and Ω^-. As the Poisson equation, g, $\mathfrak{p}$ and $\mathfrak{q}$ are known functions. This paper only focuses on the one-dimensional problem given by

$$u_t = u_{xx} - f, \quad (x,t) \in (\mathfrak{a}, x_\alpha) \cup (x_\alpha, \mathfrak{b}) \times (0,T], \tag{27}$$

$$u(x,0) = u_0(x), \quad x \in (\mathfrak{a}, \mathfrak{b}), \tag{28}$$

$$u(x,t) = g(x,t), \quad (x,t) \in \{\mathfrak{a}, \mathfrak{b}\} \times [0,T], \tag{29}$$

$$[u] = \mathfrak{p}(t), \quad t \in [0,T], \tag{30}$$

$$[u_x] = \mathfrak{q}(t), \quad t \in [0,T]. \tag{31}$$

where the source f and initial value u_0 may be discontinuous or singular across a fixed interface located at x_α.

4.1 IFD-IIM for the 1D heat conduction problem

Since the interface is fixed and all the quantities are continuous with time, we can approximate the time derivative using the Crank-Nicolson method, as follows

$$u^{n+1} = u^n + \frac{\Delta t}{2}\left\{(u_{xx} - f)^{n+1} + (u_{xx} - f)^n\right\} + O(\Delta t^2). \tag{32}$$

Applying the fourth-order operator (3) to equation (32) for $i = 2, \ldots, N$, yields

$$\mathfrak{D}^2 u\big|_i^{n+1} = \mathfrak{D}^2 u\big|_i^n + \frac{\Delta t}{2}\left\{ \mathfrak{D}^2 u_{xx}\big|_i^{n+1} + \mathfrak{D}^2 u_{xx}\big|_i^n - \mathfrak{D}^2 f\big|_i^{n+1} - \mathfrak{D}^2 f\big|_i^n \right\} + O(\Delta t^2). \tag{33}$$

By employing the IFD method (2) and approximation (12), equation (33) can be approximated for regular points as follows

$$\begin{aligned} \mathfrak{d}^2 u\big|_i^{n+1} &= \mathfrak{d}^2 u\big|_i^n + \frac{\Delta t}{2}\left\{ \delta^2 u\big|_i^{n+1} + \delta^2 u\big|_i^n - \mathfrak{d}^2 f\big|_i^{n+1} - \mathfrak{d}^2 f\big|_i^n \right\} \\ &\quad + O(\Delta t^2) + O(h^4), \qquad i \neq I, I+1. \end{aligned} \tag{34}$$

By using the IFD-IIM (6) and approximation (13), the implicit scheme for irregular points is given by

$$\begin{aligned} \mathfrak{d}^2 u\big|_i^{n+1} &= \mathfrak{d}^2 u\big|_i^n + \frac{\Delta t}{2}\left\{ \delta^2 u\big|_i^{n+1} + \delta^2 u\big|_i^n - \mathfrak{d}^2 f\big|_i^{n+1} - \mathfrak{d}^2 f\big|_i^n \right\} + \mathfrak{C}_i \\ &\quad + O(\Delta t^2) + O(h^3), \qquad i = I, I+1, \end{aligned} \tag{35}$$

where

$$\mathfrak{C}_i = -(c_u)_i^{n+1} + (c_u)_i^n + \frac{\Delta t}{2}\left\{ C_i^{n+1} + C_i^n - (c_f)_i^{n+1} - (c_f)_i^n \right\}. \tag{36}$$

Here, $(c_u)_i$ and $(c_f)_i$ are defined as (14), and C_i is given by (7). Thus, for 1D heat conduction equation (27), the IFD-IIM reads

$$\begin{aligned} & (b-r)U_{i-1}^{n+1} + (1-2b+2r)U_i^{n+1} + (b-r)U_{i+1}^{n+1} \\ = \; & (b+r)U_{i-1}^n + (1-2b-2r)U_i^n + (b+r)U_{i+1}^n \\ & -\frac{\Delta t}{2}\left\{ b f_{i-1}^{n+1} + (1-2b) f_i^{n+1} + b f_{i+1}^{n+1} \right\} \\ & -\frac{\Delta t}{2}\left\{ b f_{i-1}^{n} + (1-2b) f_i^{n} + b f_{i+1}^{n} \right\} + \mathfrak{C}_i, \end{aligned} \tag{37}$$

where $r = \frac{1}{2}\Delta t/h^2$ and $\mathfrak{C}_i$ is given by (36). Here, we have that $\mathfrak{C}_i = 0$ for regular points.

Remark. *The IFD-IIM* (37) *is unconditionally stable and the local truncation error is of* $O(\Delta t^2 + h^4)$ *and* $O(\Delta t^2 + h^3)$ *for regular and irregular grid points, respectively. Thus a global fourth-order method is expected by taking* $\Delta t = O(h^2)$.

Remark. *As the Crank-Nicolson method is implicit in time, the IFD-IIM solves a linear system at each time step. To address this efficiently, we employed Thomas' algorithm, given that the resulting matrix is tridiagonal. Furthermore, the implicit method in space preserves the original structure of this system of equations, thus yielding a higher-order method without compromising the efficiency of the standard scheme.*

Remark. *In addition to accounting for the contributions of the source term f and its derivatives, the finite-difference scheme presented in equation* (37) *requires additional knowledge of the jump conditions for the solution given by $[u_{xx}]$, $[u_{xxx}]$, and $[u_{xxxx}]$. It is worth noting that, although not presented here, there are techniques available for deriving all the necessary jump conditions from the principal jump conditions [12].*

5 Numerical results for the Poisson equation

In this section, we test the IFD-IIM for different examples of the Poisson equation. In the following simulations, we numerically solve the equation for a given right-hand side function and compare it with its analytic solution. In all cases the resulting linear system is solved using the Thomas' algorithm.

The numerical method is tested using three different examples. Example 1 considers a smooth solution to verify the fourth-order implicit method for smooth solutions. Example 2 studies a Poisson equation with a discontinuous solution in a single interface point. Finally, Example 3 presents a discontinuous problem with multiple interface points. A Matlab code of these examples can be download at `https://github.com/CIMATMerida/IFD-IIM`

For the all these examples, the computational domain is the interval $[0, 1]$, and the grid spacing is $h = 1/N$ for different N sub-intervals. The errors are reported using the L_∞-norm and the discrete L_2-norm calculated as

$$\|e\|_\infty = \max_{1\le i\le N+1} |u_i - U_i|, \quad \text{and} \quad \|e\|_2 = \left(\sum_{i=1}^{N+1} (u_i - U_i)^2 \Delta x\right)^{1/2}, \tag{38}$$

respectively, where U_i and u_i corresponds to the numerical and exact solution at x_i, respectively. The estimated order of accuracy is computed as

$$\text{Order} := \frac{\log\left(\|e\|_{N_1} / \|e\|_{N_2}\right)}{\log\left(N_1/N_2\right)}, \tag{39}$$

where N_1 and N_2 indicates the different number of sub-intervals of the corresponding norm.

5.1 Example 1. Poison equation with smooth solution

Example 1 considers the one-dimensional Poisson problem

$$\begin{aligned} u_{xx}(x) &= f(x), \quad x \in (0,1), \\ u(0) &= e^{-\pi/4}, \\ u(1) &= e^{-9\pi/4}, \end{aligned} \tag{40}$$

where the right-hand side in (40) is given by

$$f(x) = 4\pi(16\pi x^2 - 8\pi x + \pi - 2)e^{-4\pi(x-1/4)^2}.$$

The exact solution of problem (40) is an infinitely smooth and differentiable function given by the following expression

$$u(x) = e^{-4\pi(x-1/4)^2}. \tag{41}$$

Note that the function f and Dirichlet boundary conditions are obtained directly from (41). Due to the regularity of the solution, the jump contributions $\mathcal{C}_I$ and $\mathcal{C}_{I+1}$ are equal to zero.

Table 1 presents the convergence analysis of Example 1 for different grid resolutions. If $b = 0$, then we recover the standard central finite-difference method of second-order accuracy. On the other hand, the fourth-order implicit scheme is recovered when $b = 1/12$. Last row of table 1 shows the numerical order calculated by the regression-line slope based on a least squares method (LSM) of the L_∞- and L_2-norm error. A complete analysis of the IFD method for smooth solutions can be found in [29].

Table 1: Convergence analysis of Example 1 testing a Poisson equation with smooth solution.

	Central scheme ($b = 0$)				IFD ($b = 1/12$)			
N	L_∞-norm	Order	L_2-norm	Order	L_∞-norm	Order	L_2-norm	Order
10	7.52e-02	—	2.84e-02	—	9.30e-03	—	3.56e-03	—
20	1.70e-02	2.15	6.52e-03	2.12	5.05e-04	4.20	1.93e-04	4.21
40	4.15e-03	2.03	1.60e-03	2.03	3.06e-05	4.04	1.17e-05	4.04
80	1.03e-03	2.01	3.98e-04	2.01	1.90e-06	4.01	7.27e-07	4.01
160	2.57e-04	2.00	9.95e-05	2.00	1.18e-07	4.00	4.54e-08	4.00
LSM		**2.04**		**2.04**		**4.06**		**4.06**

5.2 Example 2. Poison equation with a single interface

For Example 2, we show the method's capability by solving a single interface problem located at $x = x_\alpha$. Here, we consider the one-dimensional Poisson problem

$$\begin{aligned} u_{xx}(x) &= f(x), \quad x \in (0, x_\alpha) \cup (x_\alpha, 1), \\ u(0) &= 0 \\ u(1) &= -1, \\ [u] &= \cos(\pi x_\alpha) - \sin(\pi x_\alpha), \\ [u_x] &= -\pi(\sin(\pi x_\alpha) + \cos(\pi x_\alpha)), \end{aligned} \tag{42}$$

where the right-hand side in (42) is

$$f(x) = \begin{cases} -\pi^2 \sin(\pi x), & x < x_\alpha, \\ -\pi^2 \cos(\pi x), & x > x_\alpha. \end{cases}$$

The exact solution of problem (42) is a discontinuous function given by the expression

$$u(x) = \begin{cases} \sin(\pi x), & x < x_\alpha, \\ \cos(\pi x), & x > x_\alpha. \end{cases} \tag{43}$$

Note that the right-hand side, Dirichlet boundary conditions, and principal jump conditions are obtained directly from equation (43).

We test two different points: $x_\alpha = 0.4$ and $x_\alpha = 0.63$. For the case $x_\alpha = 0.4$, we always have $h_R/h = 1$ for $N = 10 \times 2^n$ $(n = 0, 1, 2, \dots)$; thus the interface is always located at one grid point of that resolution. In general, for $x_\alpha = 0.63$, we have different h_R/h values for different N numbers. Fig. 2 shows the numerical and exact solution when the interface is located at these two values using $N = 40$. As expected, the exact solution is accurately recovered for both cases.

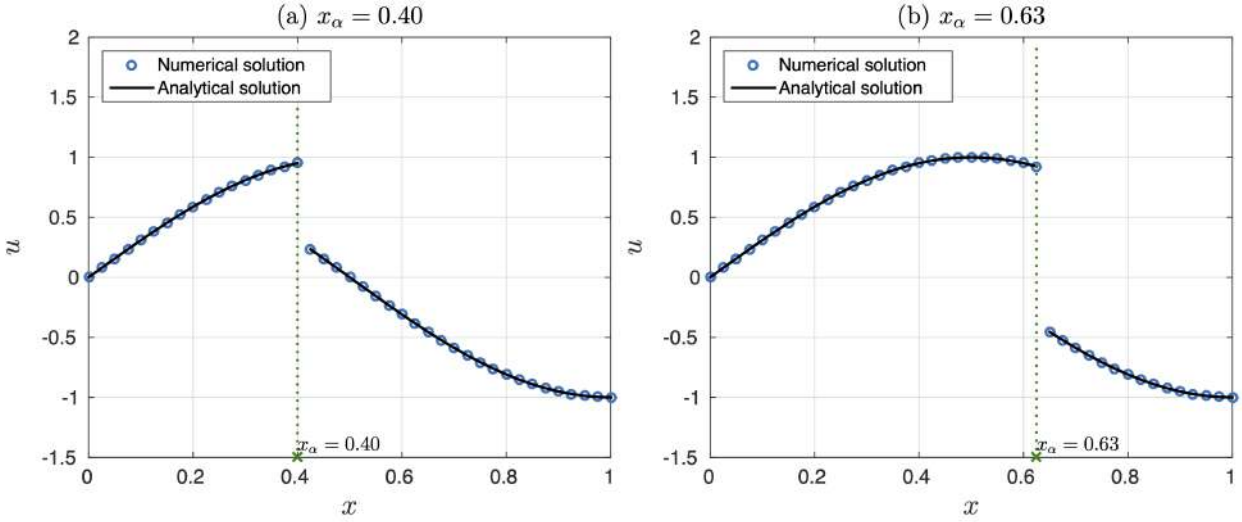


Figure 2: Numerical and exact solution of Example 2 using $N = 40$ using (a) $x_\alpha = 0.4$, and (b) $x_\alpha = 0.63$. Here, the interface is located at one grid point using $N = 40$ and $x_\alpha = 0.4$.

Table 2 shows the convergence analysis for Example 2 for the two x_α values. Observe that a second-order method is recovered for $b = 0$ and a fourth-order method for $b = 1/12$. As expected, the IFD-IIM numerical order does not depend on the location of the interface. However, the magnitude of the error may present minor variations due to the interface position. Fig. 3 shows the error analysis corresponding to interface locations $x_\alpha = 0.40$ and $x_\alpha = 0.63$ for $N = 10, 11, 12, \dots, 320$. The solid black line for this and the following figures shows the numerical order calculated by the regression-line slope based on a least squares method of the L_∞-norm.

Table 2: Convergence analysis of Example 2 using the IFD-IIM.

	IIM ($b = 0$)				IFD-IIM ($b = 1/12$)			
	$x_\alpha = 0.40$		$x_\alpha = 0.63$		$x_\alpha = 0.40$		$x_\alpha = 0.63$	
N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	1.69e-02	—	5.19e-03	—	5.75e-05	—	3.42e-05	—
20	4.21e-03	2.00	1.18e-03	2.13	3.58e-06	4.00	1.97e-06	4.12
40	1.05e-03	2.00	3.07e-04	1.95	2.24e-07	4.00	1.39e-07	3.82
80	2.63e-04	2.00	7.53e-05	2.03	1.40e-08	4.00	7.75e-09	4.16
160	6.57e-05	2.00	1.86e-05	2.01	8.74e-10	4.00	5.37e-10	3.85
LSM		**2.04**		**2.02**		**4.00**		**3.99**

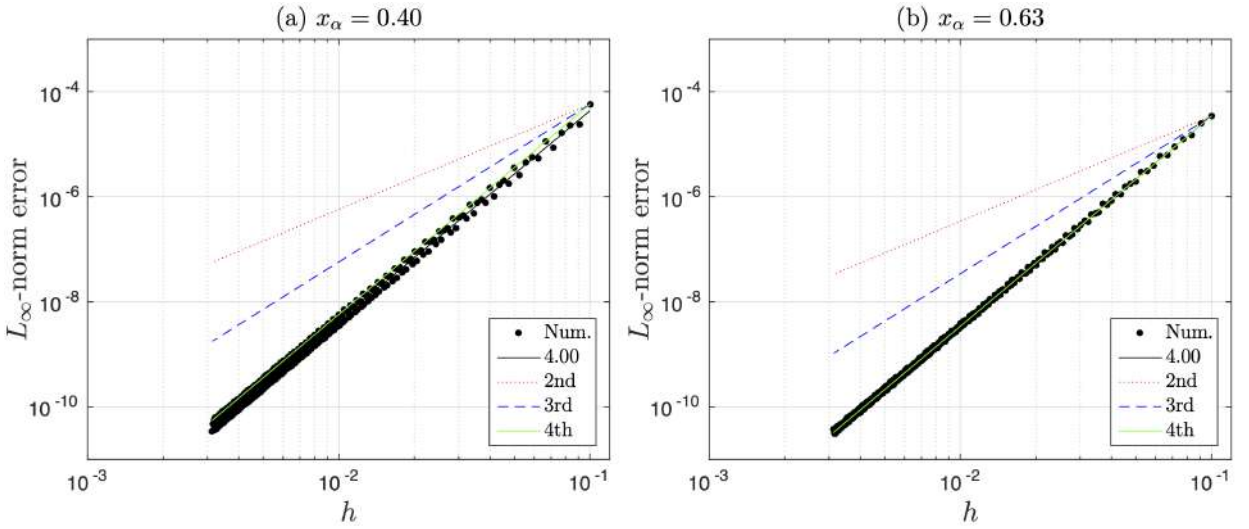


Figure 3: Convergence analysis of Example 2 for $N = 10, \ldots, 320$ using (a) $x_\alpha = 0.40$, and (b) $x_\alpha = 0.63$. The solid black line shows the numerical order 4.00 calculated by the regression-line slope based on a least squares method of the L_∞-norm.

The contribution formula includes jumps $[u]$, $[u_x]$, $[u_{xx}]$, $[u_{xxx}]$, and $[u_{xxxx}]$ to obtain a fourth-order accurate method. Fig. 4 shows that if we add additional jumps of high-order derivatives into $\mathcal{C}$, such as $[u_{xxxxx}] = [f_{xxx}]$, we observe that the error oscillation decreases in comparison with Fig. 3 results. It is expected because the method is $O(h^4)$ for the whole computational domain, including the irregular points. Thus, we can mitigate error oscillations due to interface position by adding high-order jumps.

Now we study the effect of removing jumps in $\mathcal{C}$. Let us consider a contribution term up to the third derivative jump by dropping $[u_{xxxx}] = [f_{xx}]$. Thus, the numerical approximation is only third-order accurate, as shown in Fig. 5 for different interface values. Note that error oscillations may have a clear pattern or a scattered distribution depending on the interface location. In general, the error magnitude perturbations are related to the variations coming from h_R/h and h_L/h values

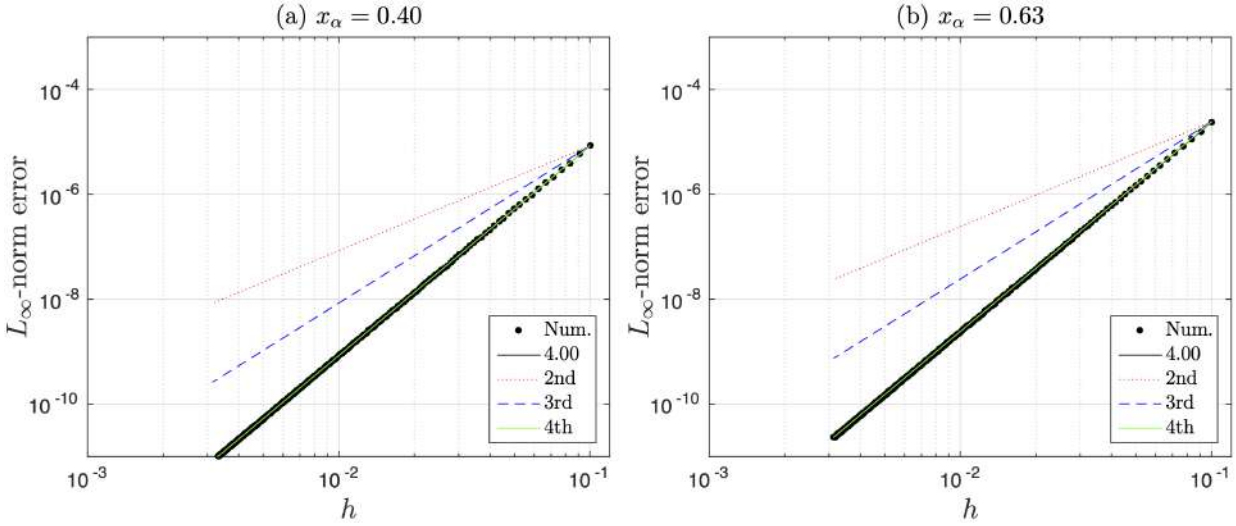


Figure 4: Convergence analysis of Example 2 for $N = 10, \ldots, 320$ using (a) $\alpha = 0.40$, and (b) $\alpha = 0.63$. The contribution includes jumps up to fifth-order ($[u_{xxxxx}] = [f_{xxx}]$).

in the $\mathcal{C}$ formula (26).

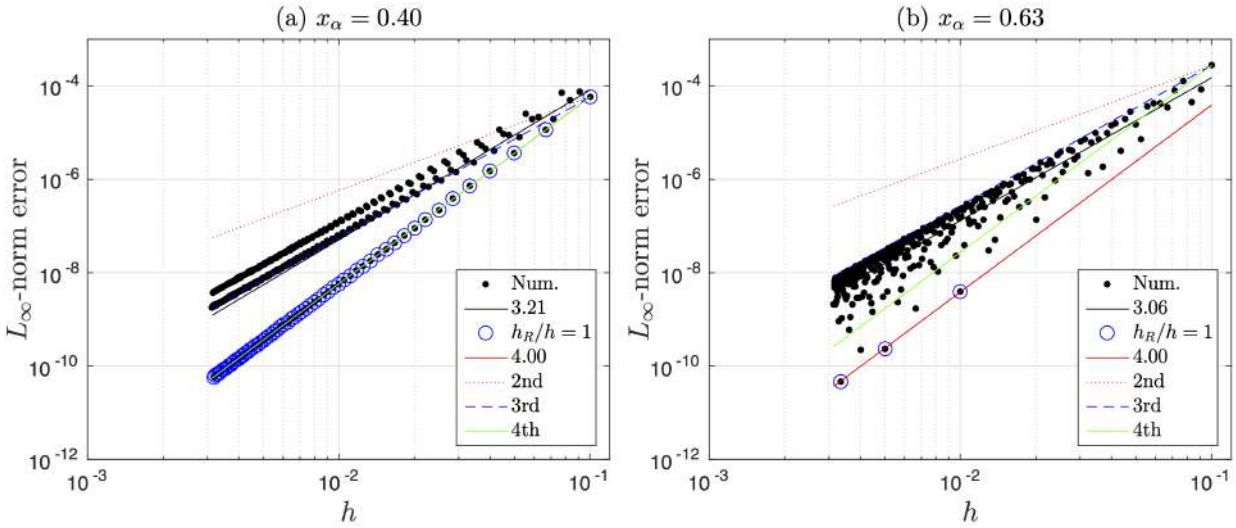


Figure 5: Convergence analysis of Example 2 for $N = 10, \ldots, 320$ using (a) $\alpha = 0.40$, and (b) $\alpha = 0.63$. The contribution includes jumps up to third-order ($[u_{xxx}] = [f_x]$).

Note that there are some points in Fig. 5, marked with circles, that are close to the fourth-order

line. Those circled markers correspond to the values with $h_R/h = 1$: (a) $N = 5, 10, 15, \ldots, 315$ for $x_\alpha = 0.40$, and (b) $N = 100, 200, 300$ for $x_\alpha = 0.63$. Both set of points satisfy that the interface x_α is located at a grid point x_I. In the case of $x_\alpha = 0.4$, we observe that the global order (black points) is close to 3.21; meanwhile, the circled points order is four. Similar behavior is obtained for $x_\alpha = 0.63$. Thus, for a given mesh resolution N, the IFD-IIM is still fourth-order accurate for $h_R/h = 1$, as proven theoretically in Section 3.2.

5.3 Example 3. Poison equation with multiple interface points

Example 3 investigates the numerical scheme capability to solve a multiple interface problem. We only focus on two interface points located at $x = x_{\alpha_1}$ and $x = x_{\alpha_2}$. However, the methodology could be applied for multiple interfaces by doing minor modifications in the implementation. For this problem, the exact solution of the Poisson problem is given by

$$u(x) = \begin{cases} \sin(2\pi x), & x < x_{\alpha_1}, \\ \cos(2\pi x), & x_{\alpha_1} < x < x_{\alpha_2}, \\ \sin(2\pi x), & x_{\alpha_2} < x. \end{cases} \tag{44}$$

The right-hand function, Dirichlet boundary conditions, and principal jump conditions are obtained directly from the exact solution (44). We consider the same computational domain and grid resolution as previous 1D examples. Fig. 6 presents the analytical and numerical solution when the interface is located at $x_{\alpha_1} = 0.3$ and $x_{\alpha_2} = 0.7$ using $N = 40$. This figure also shows the error analysis. As expected, the IFD-IIM is a fourth-order accurate method.

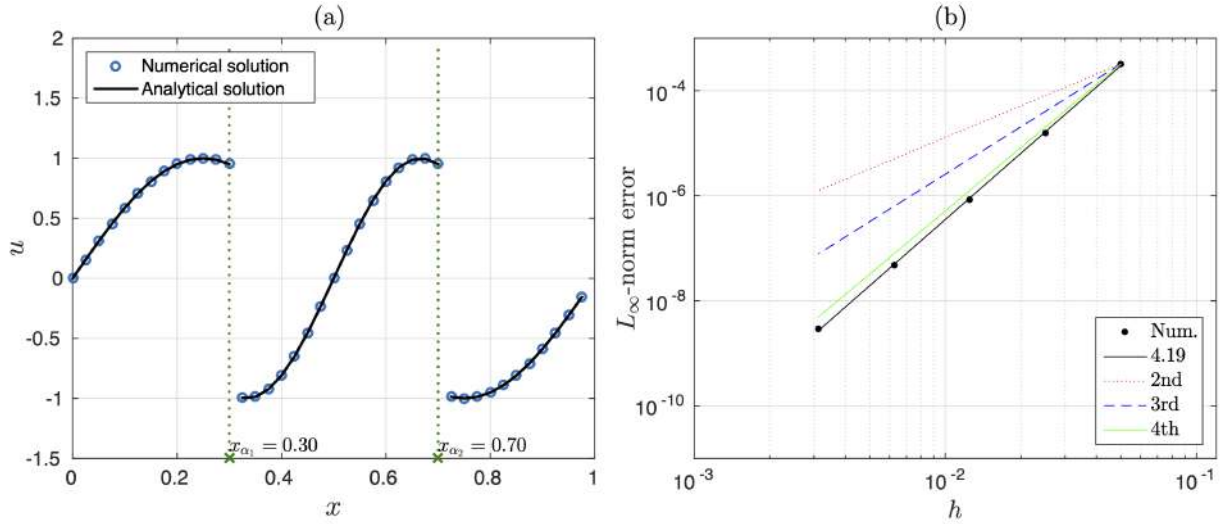


Figure 6: (a) Numerical and exact solution of Example 3 with multiple interfaces using $N = 40$, and (b) convergence error analysis using grid resolutions of $N = 10$, 20, 40, 80 and 160.

6 Numerical results for the Heat equation

This section tests the IFD-IIM for the Heat conduction equation in different scenarios. Example 4 verifies the fourth-order implicit method for smooth solutions in space. Example 5 studies a Heat equation with a discontinuous solution in a single interface point and no source term. Finally, Example 6 presents a general discontinuous problem. For the all these examples, the computational domain is the same interval $[0, 1]$ and space step, h, used for the Poisson examples. Here, the final time is at $T = 0.5$ and the time step is $\Delta t = \frac{1}{4}h^2$. The error and estimated order of accuracy are reported using (38) and (39), respectively, at the final step approximation.

6.1 Example 4. Heat equation with smooth solution

This example considers the heat conduction problem

$$\begin{aligned} u_t(x,t) &= u_{xx}(x,t) - f(x,t), & (x,t) \in (0,1) \times (0, \tfrac{1}{2}], \\ u(x,0) &= \sin\left(\tfrac{3}{2}\pi x\right), & x \in (0,1), \\ u(0,t) &= 0, & t \in [0, \tfrac{1}{2}], \\ u(1,t) &= -e^{-t}, & t \in [0, \tfrac{1}{2}], \end{aligned} \tag{45}$$

where f in (45) is given by

$$f(x,t) = \left(1 - \frac{9}{4}\pi^2\right) \sin\left(\frac{3}{2}\pi x\right) e^{-t}.$$

This example is constructed so that the exact solution is

$$u(x,t) = \sin\left(\frac{3}{2}\pi x\right) e^{-t}. \tag{46}$$

Note that the source term, f, initial condition and boundary conditions are obtained from the exact solution (46). The convergence analysis of this example is presented in Table 3 with the final time being $T = 0.5$. As expected, the high-order methods reach their corresponding order of accuracy.

Table 3: Convergence analysis of Example 4 testing a heat equation with smooth solution.

	Central scheme ($b = 0$)				IFD ($b = 1/12$)			
N	L_∞-norm	Order	L_2-norm	Order	L_∞-norm	Order	L_2-norm	Order
10	1.66e-02	—	1.07e-02	—	1.84e-04	—	1.18e-04	—
20	4.12e-03	2.01	2.65e-03	2.01	1.14e-05	4.01	7.34e-06	4.01
40	1.03e-03	2.00	6.60e-04	2.00	7.15e-07	4.00	4.58e-07	4.00
80	2.58e-04	2.00	1.65e-04	2.00	4.47e-08	4.00	2.86e-08	4.00
160	6.44e-05	2.00	4.12e-05	2.00	2.79e-09	4.00	1.79e-09	4.00
LSM		**2.00**		**2.00**		**4.00**		**4.00**

6.2 Example 5. Heat equation with discontinuous solution

In this example, the heat equation with discontinuous solution is given by

$$\begin{aligned}
u_t(x,t) &= u_{xx}(x,t), && (x,t) \in (0,x_\alpha) \cup (x_\alpha,1) \times (0,\tfrac{1}{2}], \\
u(x,0) &= \begin{cases} \sin(x), & x < x_\alpha, \\ \cos(x), & x > x_\alpha, \end{cases} && x \in (0,x_\alpha) \cup (x_\alpha,1), \\
u(0,t) &= 0, && t \in [0,\tfrac{1}{2}], \\
u(1,t) &= \cos(1)e^{-t}, && t \in [0,\tfrac{1}{2}], \\
[u](t) &= (\cos(x_\alpha) - \sin(x_\alpha))\, e^{-t}, && t \in [0,\tfrac{1}{2}], \\
[u_x](t) &= -(\sin(x_\alpha) + \cos(x_\alpha))\, e^{-t}, && t \in [0,\tfrac{1}{2}].
\end{aligned} \tag{47}$$

The exact solution u is defined as

$$u(x,t) = \begin{cases} \sin(x)e^{-t}, & x < x_\alpha, \\ \cos(x)e^{-t}, & x > x_\alpha. \end{cases} \tag{48}$$

Similar to previous examples, we use $x_\alpha = 0.4$ and $x_\alpha = 0.63$. Note that the source term f of this problem is $f \equiv 0$. In this example, both the function u and their derivatives have jumps, and these jumps vary in time. The boundary condition, initial condition, and all jumps are specified by u.

The figure of numerical solution and absolute error plot using $N = 40$ for different time stages are shown in Fig. 7. On the other hand, the one-dimensional results at $T = 0.5$ are presented in Figs. 8 and 9. As expected, we can accurately recover the exact solution using the proposed method for $N = 40$ and considering different interface values. Additionally, we provide the absolute error for both cases, noting that the maximum error is found in close proximity to the point x_α. This behavior is also expected since the local truncation error for irregular points is less accurate than for regular points. Finally, the convergence analysis using $N = 10$, 20, 40, 80 and 160 confirms the fourth-order global convergence of the proposed method.

More details of the grid refinement analysis at $T = 0.5$ is presented in Table 4 for two different interface point locations. As expected, a fourth-order method is obtained for both interfaces. We remark that the variation of the errors using $x_\alpha = 0.63$ is because the different h_R/h values resulting from the discretization.

From Table 5, we can find the results when the time step is chosen as $\Delta t = h$. As expected, the proposed IFD-IIM is stable and has two-order convergence either we take a second- or fourth-order approximation in space. We remark that this is a property of the selected time integration method. For instance, similar findings as Table 5 can be obtained for smooth solutions.

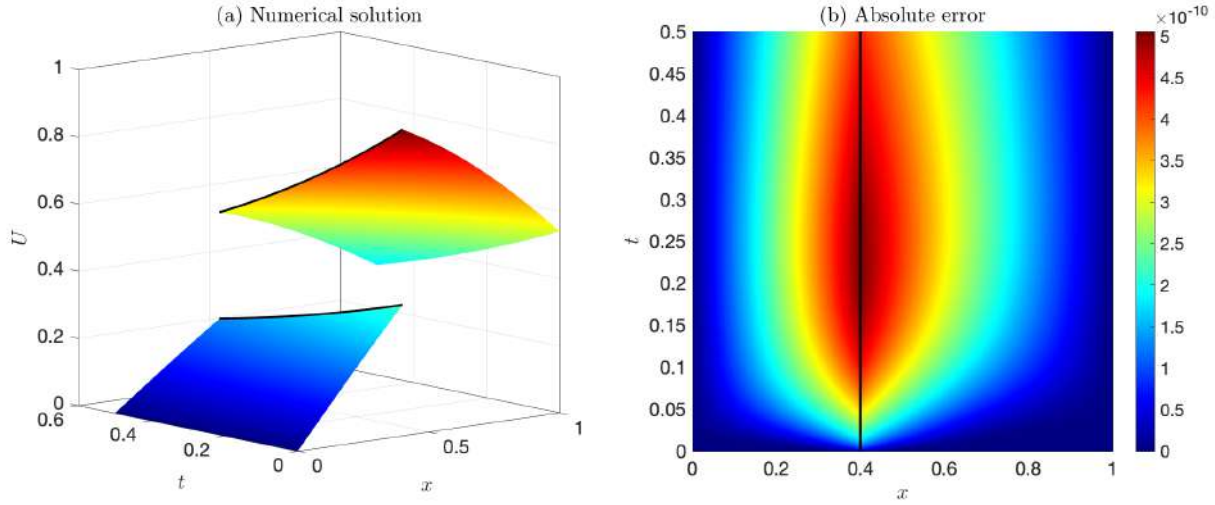


Figure 7: Numerical solution and absolute error of Example 5 using $N = 40$, $x_\alpha = 0.4$ for $t \in [0, 0.5]$.

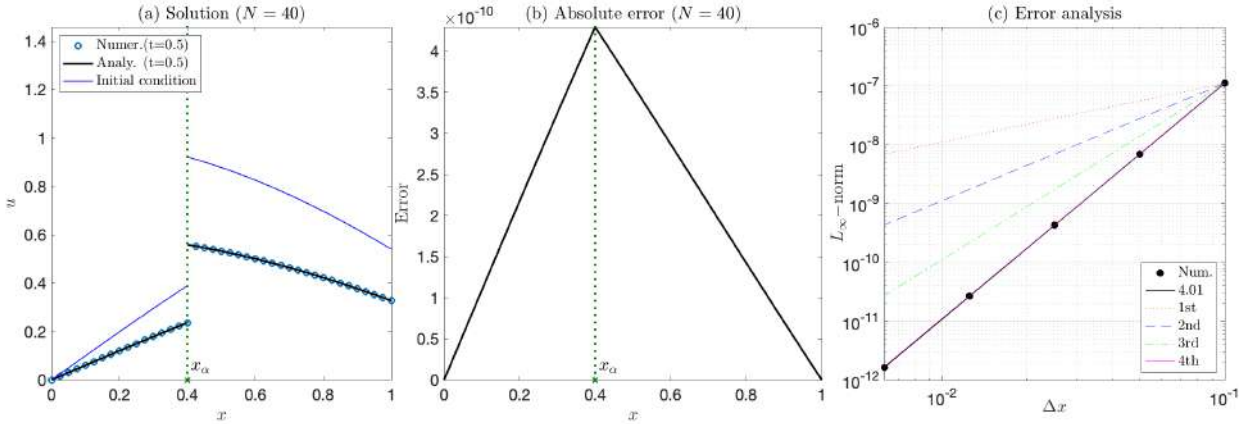


Figure 8: Numerical results at final time simulation $T = 0.5$ and $x_\alpha = 0.4$ of Example 5.

6.3 Example 6. Heat equation with a general discontinuous solution

Finally, Example 6 investigates the capability to solve a heat interface problem with a general discontinuous solution. For this problem, we slightly modified the previous example such that the exact solution of the heat conduction problem is given by

$$u(x) = \begin{cases} \sin(2\pi x)e^{-t}, & x < x_\alpha, \\ \cos(2\pi x)e^{-t}, & x > x_\alpha. \end{cases} \tag{49}$$

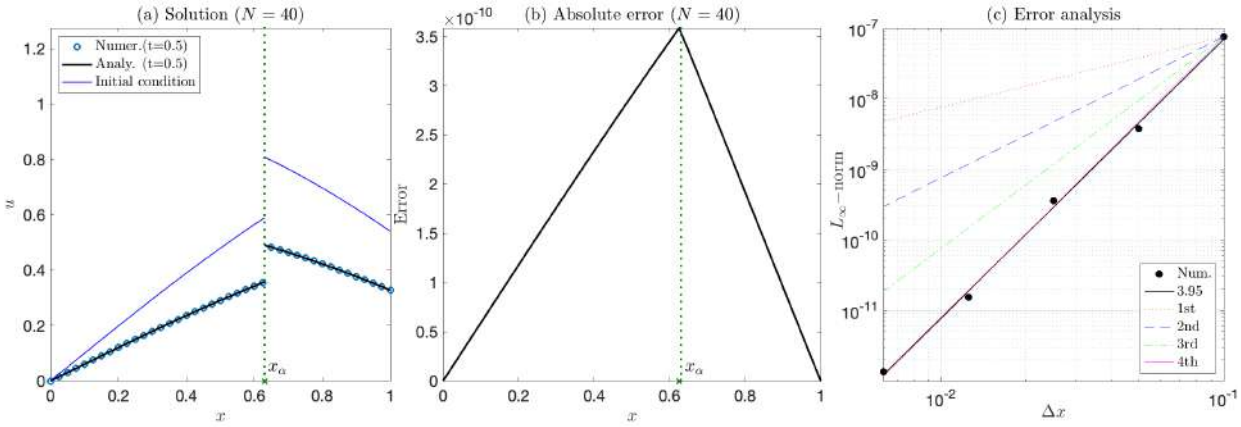


Figure 9: Numerical results at final time simulation $T = 0.5$ and $x_\alpha = 0.63$ of Example 5.

Table 4: Convergence analysis of Example 5 using the IFD-IIM for the heat equation with a discontinuous solution.

	IFD-IIM ($x_\alpha = 0.4$)				IFD-IIM ($x_\alpha = 0.63$)			
N	L_∞-norm	Order	L_2-norm	Order	L_∞-norm	Order	L_2-norm	Order
10	1.11e-07	—	6.52e-08	—	7.57e-08	—	4.49e-08	—
20	6.90e-09	4.01	4.01e-09	4.02	3.75e-09	4.34	2.25e-09	4.32
40	4.29e-10	4.00	2.49e-10	4.01	3.58e-10	3.39	2.09e-10	4.43
80	2.68e-11	4.00	1.55e-11	4.00	1.53e-11	4.55	8.93e-12	4.55
160	1.63e-12	4.03	9.39e-13	4.04	1.35e-12	3.50	7.82e-13	3.51
LSM		**4.01**		**4.02**		**3.95**		**3.96**

Table 5: Convergence analysis of Example 5 testing the heat equation with $\Delta t = h$.

	IIM ($b = 0$)				IFD-IIM ($b = 1/12$)			
N	L_∞-norm	Order	L_2-norm	Order	L_∞-norm	Order	L_2-norm	Order
10	2.94e-04	—	1.87e-04	—	3.78e-05	—	2.67e-05	—
20	8.53e-05	1.78	4.85e-05	1.95	1.03e-05	1.87	7.22e-06	1.89
40	2.11e-05	2.01	1.26e-05	1.94	2.74e-06	1.91	1.92e-06	1.91
80	5.31e-06	1.99	3.15e-06	2.00	6.91e-07	1.99	4.83e-07	1.99
160	1.34e-06	1.99	7.83e-07	2.00	1.72e-07	2.00	1.21e-07	2.00
LSM		**1.96**		**1.97**		**1.95**		**1.95**

Here, the source term f is a general function that varies both in time and space. It is directly derived from equation (27) as well as the exact solution (49). Furthermore, the exact solution is utilized to specify the boundary condition, initial condition, and all jumps contributions.

In Fig. 10, we depict the temporal evolution of the numerical solution and absolute errors using parameters $N = 40$ and $x_\alpha = 0.4$. It is noteworthy that the behavior of the solution differs from the previous example. More in-depth analysis of the results at $T = 0.5$ are illustrated in Figs. 11 and 12, corresponding to values of $x_\alpha = 0.4$ and $x_\alpha = 0.63$, respectively. First, the numerical solution is

accurately approximated using the proposed implicit method for $N = 40$, regardless of the interface locations. It is worth noting that the maximum error for $x_\alpha = 0.63$ is concentrated near to this point; however, this is not observed for $x_\alpha = 0.4$. This difference in behavior can be attributed to the interplay between the interface location ($h_R/h = 1$ and $x_\alpha = 0.4$) and the non-zero right-hand side values, which contribute to the local truncation errors. Grid refinement analyses are also provided in these figures using $N = 10$, 20, 40, 80 and 160. As anticipated, the IFD-IIM method demonstrates fourth-order accuracy, a validation that is further detailed in Table 6.

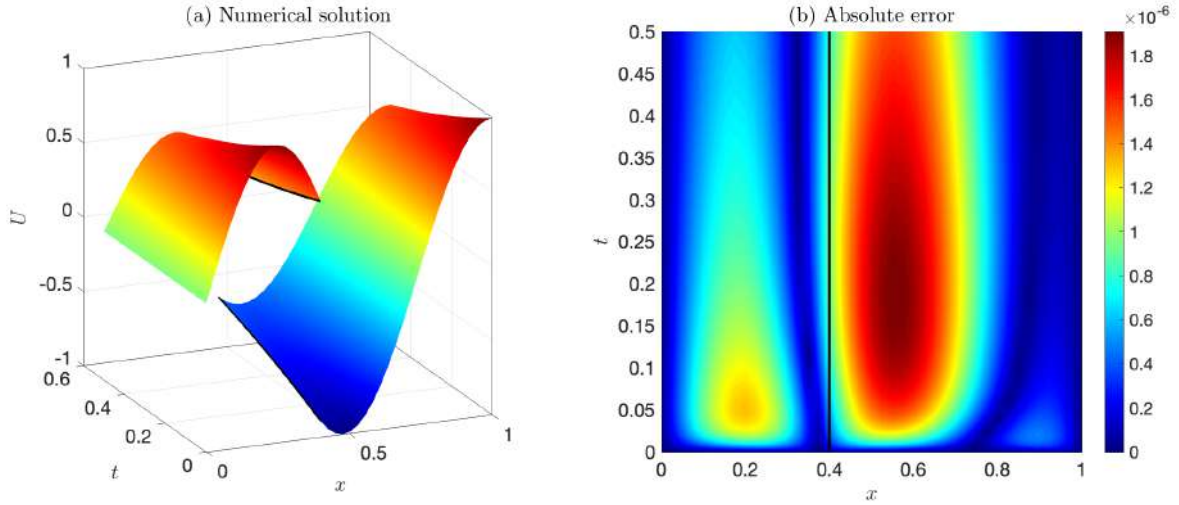


Figure 10: Numerical solution and absolute error of Example 6 using $N = 40$, $x_\alpha = 0.4$ for $t \in [0, 0.5]$.

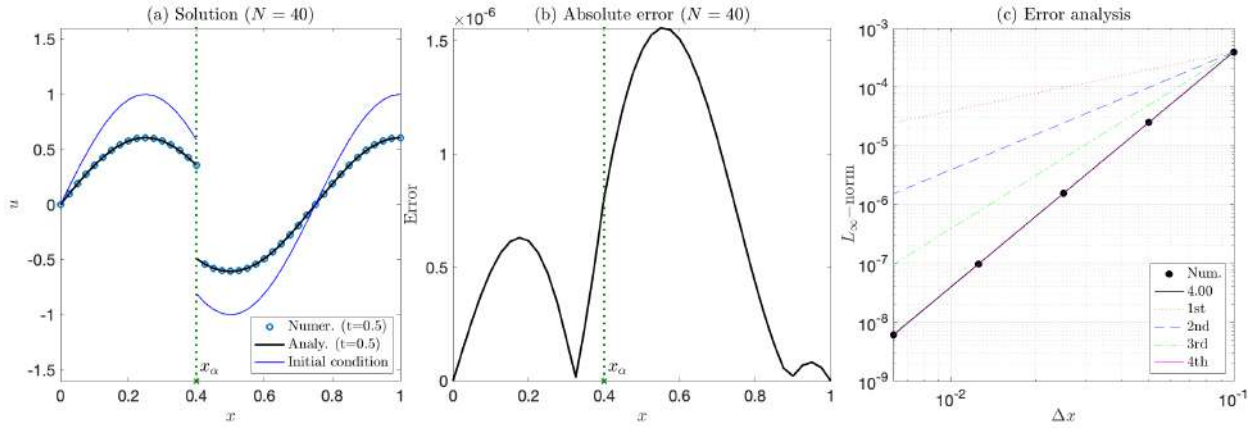


Figure 11: Numerical results at final time simulation $T = 0.5$ and $x_\alpha = 0.4$ of Example 6.

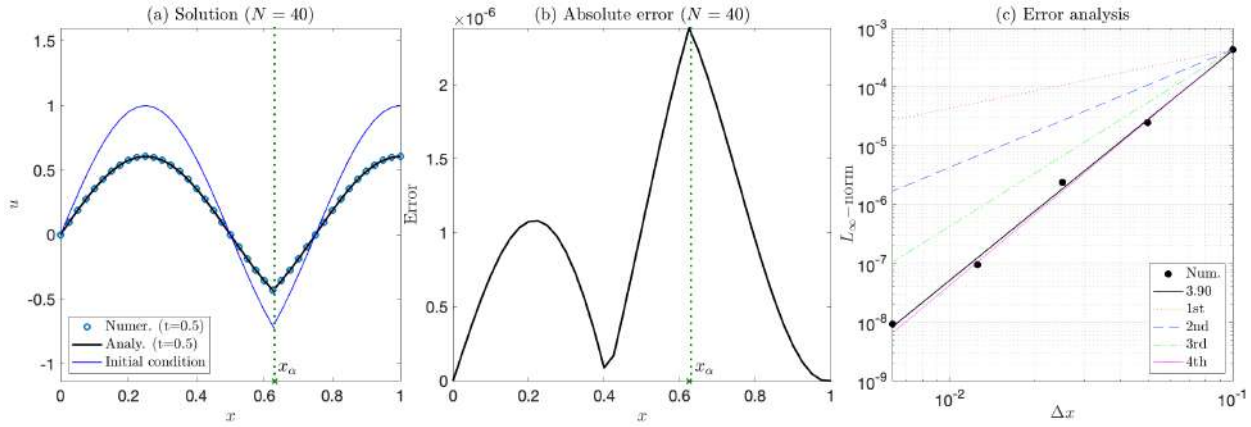


Figure 12: Numerical results at final time simulation $T = 0.5$ and $x_\alpha = 0.63$ of Example 6.

Table 6: Convergence analysis of Example 6 using the IFD-IIM for the heat equation with a general discontinuous solution.

	IFD-IIM ($x_\alpha = 0.40$)				IFD-IIM ($x_\alpha = 0.63$)			
N	L_∞-norm	Order	L_2-norm	Order	L_∞-norm	Order	L_2-norm	Order
10	3.91e-04	—	2.18e-04	—	4.32e-04	—	2.43e-04	—
20	2.50e-05	3.97	1.34e-05	4.02	2.48e-05	4.12	1.45e-05	4.07
40	1.56e-06	4.00	8.37e-07	4.01	2.38e-06	3.38	1.12e-06	3.69
80	9.72e-08	4.00	5.22e-08	4.00	9.48e-08	4.65	5.56e-08	4.34
160	6.08e-09	4.03	3.26e-09	4.00	9.29e-09	3.35	4.37e-09	3.69
LSM		**4.00**		**4.01**		**3.90**		**3.95**

7 Conclusions

This work serves as the general foundation for addressing high-order IIMs to solve interface problems, facilitating comprehensive investigations into this theoretical framework. Here, we present a fourth-order finite-difference scheme for approximating the second-order derivative of real-valued continuous and discontinuous functions. This method combines an implicit formulation with the IIM. Our proposed scheme employs a three-point stencil, achieving different accuracy at regular and irregular points. To illustrate the effectiveness of the proposed IFD-IIM approach, we applied it to solve the one-dimensional Poisson equation and the heat conduction equation. The global accuracy of the fourth order was demonstrated using several numerical examples for both equations. Hence, this study establishes a general strategy for high-order IIMs, enabling their application to elliptic and time-evolving problems in several dimensions. Additionally, the implicit procedure lends itself to developing of higher-order numerical schemes based on the IIM, including sixth-order methods.

Acknowledgments

This work was partially supported by CONAHCYT under the program *Investigadoras e Investigadores por México*. We also thanks the facilities provided by the *Facultad de Matemáticas (UADY)* for the development of this work.

References

[1] P. A. Berthelsen. A decomposed immersed interface method for variable coefficient elliptic equations with non-smooth and discontinuous solutions. *Journal of Computational Physics*, 197(1):364–386, 2004.

[2] H. Cho, H. Han, B. Lee, Y. Ha, and M. Kang. A second-order boundary condition capturing method for solving the elliptic interface problems on irregular domains. *Journal of Scientific Computing*, 81:217–251, 2019.

[3] J. F. Claerbout. *The craft of wave-field extrapolation, in Imaging the earth's interior*, volume 1. Blackwell scientific publications Oxford, 1985.

[4] M. Colnago, W. Casaca, and L. F. de Souza. A high-order immersed interface method free of derivative jump conditions for poisson equations on irregular domains. *Journal of Computational Physics*, 423:109791, 2020.

[5] H.-J. Diersch. Fletcher, caj, computational techniques for fluid dynamics. vol. i: Fundamental and general techniques. vol. ii: Specific techniques for different flow categories. berlin etc., springer-verlag 1988. xiv, 409 pp., 183 figs./xi, 484 pp., 183 figs., dm 198, 00 as a set. isbn 3-540-18151-2/3-540-18759-6 (springer series in computational physics), 1990.

[6] X. Feng and Z. Li. Simplified immersed interface methods for elliptic interface problems with straight interfaces. *Numerical Methods for Partial Differential Equations*, 28(1):188–203, 2012.

[7] X. Feng, Z. Li, and Z. Qiao. High order compact finite difference schemes for the helmholtz equation with discontinuous coefficients. *Journal of Computational Mathematics*, pages 324–340, 2011.

[8] F. Gibou and R. Fedkiw. A fourth order accurate discretization for the laplace and heat equations on arbitrary domains, with applications to the stefan problem. *Journal of Computational Physics*, 202(2):577–601, 2005.

[9] K. Ito, Z. Li, and Y. Kyei. Higher-order, cartesian grid based finite difference schemes for elliptic equations on irregular domains. *SIAM Journal on Scientific Computing*, 27(1):346–367, 2005.

[10] R. Itzá Balam, F. Hernandez-Lopez, J. Trejo-Sánchez, and M. Uh Zapata. An immersed boundary neural network for solving elliptic equations with singular forces on arbitrary domains. *Math. Biosci. Eng*, 18(1):22–56, 2021.

[11] R. Itzá Balam and M. Uh Zapata. A new eighth-order implicit finite difference method to solve the three-dimensional helmholtz equation. *Computers & Mathematics with Applications*, 80(5):1176–1200, 2020.

[12] R. Itza Balam and M. Uh Zapata. A fourth-order compact implicit immersed interface method for 2d poisson interface problems. *Computers & Mathematics with Applications*, 119:257–277, 2022.

[13] E. Javierre, C. Vuik, F. Vermolen, and S. Van der Zwaag. A comparison of numerical models for one-dimensional stefan problems. *Journal of Computational and Applied Mathematics*, 192(2):445–459, 2006.

[14] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis*, 31(4):1019–1044, 1994.

[15] M. Li, T. Tang, and B. Fornberg. A compact fourth-order finite difference scheme for the steady incompressible navier-stokes equations. *International Journal for Numerical Methods in Fluids*, 20(10):1137–1151, 1995.

[16] Z. Li and K. Ito. *The immersed interface method: numerical solutions of PDEs involving interfaces and irregular domains*. SIAM, 2006.

[17] M. N. Linnick and H. F. Fasel. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics*, 204(1):157–192, 2005.

[18] X.-D. Liu and T. Sideris. Convergence of the ghost fluid method for elliptic equations with interfaces. *Mathematics of computation*, 72(244):1731–1746, 2003.

[19] Y. Liu and M. K. Sen. A practical implicit finite-difference method: examples from seismic modelling. *Journal of Geophysics and Engineering*, 6(3):231–249, 2009.

[20] L. Mu, J. Wang, X. Ye, and S. Zhao. A new weak galerkin finite element method for elliptic interface problems. *Journal of Computational Physics*, 325:157–173, 2016.

[21] M. Nabavi, M. K. Siddiqui, and J. Dargahi. A new 9-point sixth-order accurate compact finite-difference method for the helmholtz equation. *Journal of Sound and Vibration*, 307(3-5):972–982, 2007.

[22] K. Pan, D. He, and Z. Li. A high order compact fd framework for elliptic bvps involving singular sources, interfaces, and irregular domains. *Journal of Scientific Computing*, 88(3):67, 2021.

[23] K. Pan, Y. Tan, and H. Hu. An interpolation matched interface and boundary method for elliptic interface problems. *Journal of computational and applied mathematics*, 234(1):73–94, 2010.

[24] C. S. Peskin. The immersed boundary method. *Acta numerica*, 11:479–517, 2002.

[25] J. H. Seo and R. Mittal. A high-order immersed boundary method for acoustic wave scattering and low-mach number flow-induced sound in complex geometries. *Journal of computational physics*, 230(4):1000–1019, 2011.

[26] J. A. Sethian et al. *Level set methods and fast marching methods*, volume 98. Cambridge Cambridge UP, 1999.

[27] Y.-e. Shi, R. K. Ray, and K. D. Nguyen. A projection method-based model with the exact c-property for shallow-water flows over dry and irregular bottom using unstructured finite-volume technique. *Computers & Fluids*, 76:178–195, 2013.

[28] M. Uh and S. Xu. The immersed interface method for simulating two-fluid flows. *Numerical Mathematics: Theory, Methods and Applications*, 7(4):447–472, 2014.

[29] M. Uh Zapata and R. Itzá Balam. High-order implicit finite difference schemes for the two-dimensional poisson equation. *Applied Mathematics and Computation*, 309:222–244, 2017.

[30] Y. Wang and J. Zhang. Sixth order compact scheme combined with multigrid method and extrapolation technique for 2d poisson equation. *Journal of Computational Physics*, 228(1):137–146, 2009.

[31] A. Wiegmann and K. P. Bube. The explicit-jump immersed interface method: finite difference methods for pdes with piecewise smooth solutions. *SIAM Journal on Numerical Analysis*, 37(3):827–862, 2000.

[32] S. Xu and Z. J. Wang. Systematic derivation of jump conditions for the immersed interface method in three-dimensional flow simulation. *SIAM Journal on Scientific Computing*, 27(6):1948–1980, 2006.

[33] M. U. Zapata, R. I. Balam, and J. Montalvo-Urquizo. A compact sixth-order implicit immersed interface method to solve 2d poisson equations with discontinuities. *Mathematics and Computers in Simulation*, 210:384–407, 2023.

[34] S. Zhai, X. Feng, and Y. He. A new method to deduce high-order compact difference schemes for two-dimensional poisson equation. *Applied Mathematics and Computation*, 230:9–26, 2014.

[35] J. Zhang. Multigrid method and fourth-order compact scheme for 2d poisson equation with unequal mesh-size discretization. *Journal of Computational Physics*, 179(1):170–179, 2002.

[36] X. Zhong. A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity. *Journal of Computational Physics*, 225(1):1066–1099, 2007.

[37] Y. Zhou, S. Zhao, M. Feig, and G.-W. Wei. High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources. *Journal of Computational Physics*, 213(1):1–30, 2006.

El algoritmo KMeans y el problema de los centroides iniciales

Gael Antonio Torres Aguirre [*1], Adianez Arhely Gamboa Rivas[1], José Luis Fraga Almanza[1], Julio Saucedo Zul[1] y Jesús Alberto López Valdez[2]

[1]Facultad de Ciencias Físico Matemáticas, UAdeC
[2]Facultad de Ciencias Químicas, UAdeC

Resumen

El gran volumen de datos en el que se vive en la actualidad, hace evidente la necesidad de clasificar los datos para obtener relaciones, asociaciones y correlaciones de ellos. Para llevar a cabo esta clasificación es necesario emplear algoritmos de agrupamiento del tipo no supervisado y particionales. Un candidato de este tipo es el algoritmo KMeans, ampliamente utilizado para resolver el problema de agrupamiento. Sin embargo, este algoritmo necesita de argumentos iniciales como lo son el número de grupos y un conjunto de datos llamados centroides que son los representantes de cada uno de ellos. Esto puede ser una fortaleza pero a la vez puede representar limitaciones del algoritmo. Es por ello que en este trabajo se inicia con la caracterización general del algoritmo KMeans en base a la selección de los centroides iniciales para después estudiar la técnica de análisis de componentes principales y con esta proporcionar centroides iniciales óptimos, pero esa etapa aún está en desarrollo y por tanto solo se presenta la idea inicial. La herramienta computacional es clave para el trabajo con alta densidad de datos es por eso que en este trabajo también se tiene como objetivo implantar un marco de trabajo llamado Apache Spark.

Palabras clave: KMeans; Voronoi; Convex Hull; Centroides Iniciales; Análisis de Componentes Principales

1 Introducción

El big data está inmerso hoy en día en las áreas de investigación de las distintas ramas de las matemáticas aplicadas, es un tema relevante de investigación actual. Sin embargo, abordar o estudiar conjuntos de datos inmensos con técnicas computacionales en forma secuencial e incluso con hardware especializado, ya no es un camino viable. Es por eso que en las distintas áreas de la

[*]gael.torres@uadec.edu.mx

matemática aplicada surge el problema de cómo obtener información relevante en las distintas bases de datos que se tengan del estudio de algún fenómeno o problema a estudiar. Uno de ellos es el problema de agrupamiento de datos cuya solución resuelve e incentiva a la mejora en la toma de decisiones dando una metodología de clasificación de ellos.

El problema puede ser tratado utilizando computadoras personales con recursos computacionales rápidos y eficientes que obtienen buenos resultados. Sin embargo, este problema demanda alto desempeño al recurso físico (hardware), por medio de lógica computacional escrita en diversos lenguajes de programación de alto nivel, generando así altas demandas de procesamiento y por tanto es necesario disponer de una capacidad de cómputo mayor. Este poder de cómputo lo puede ofrecer un arreglo de PCs interconectadas bajo una red de área local. A este tipo de interconexión de computadoras se le conoce como Cluster de PCs (computadoras personales interconectadas para obtener un comportamiento único). Un tipo de estos clusters de PCs son los llamados cluster beowulf diseñados para cálculo numérico intensivo.

El agrupamiento de estos datos (clustering) es una metodología ampliamente utilizada, que resuelve los problemas antes mencionados. Entre los algoritmos de clustering más simples y fáciles de implementar computacionalmente está el llamado KMeans. Es uno de los algoritmos de Machine Learning no supervisados más utilizados por la Ciencia de Datos y que además es muy fácil de usar como de interpretar. Su objetivo es agrupar observaciones similares para descubrir patrones que a simple vista se desconocen. Para conseguirlo, el algoritmo busca un número fijo (k) de clusters en una base de datos. Sin embargo, las aplicaciones del mundo real producen grandes volúmenes de datos, por lo tanto, el cómo tratar con estas grandes cantidades de datos el problema de agrupamiento es un problema significativo y desafiante

Una metodología para trabajar con altas prestaciones de datos, escalabilidad y portabilidad es Apache Spark, ideal para interconectar PCs y que unidas trabajen en un mismo problema. Por lo tanto, con la unión de Apache Spark y el algoritmo KMeans, se puede abordar el problema de agrupamiento de datos, que es el tema inicial en este trabajo que aún se encuentra en desarrollo y una primera etapa en él es comenzar por estudiar el algoritmo KMeans.

2 Preliminares

En esta sección se presenta el algoritmo KMeans. Se describe la idea general del algoritmo por medio de una descripción textual y puntualizando los conceptos de centroide y métrica utilizada. Esto hace evidente la idea geométrica detrás del algoritmo KMeans y que se describe por medio de los diagramas de Voronoi y la envoltura convexa.

2.1. El algoritmo

El algoritmo KMeans es uno de los más simples y conocidos algoritmos de agrupamiento. Utiliza una forma fácil y sencilla para dividir un conjunto de datos en k grupos (k fijado de antemano)[5]. El método se basa en la idea de que los elementos de un cluster están agrupados alrededor de un centro o centroide geométrico. La tarea entonces es encontrar dichos centroides. Para ello se definen

k centroides iniciales, luego se toma cada punto del conjunto de datos y se sitúa en la clase del centroide más cercano. El paso siguiente consiste en recalcular el centroide de cada grupo y volver a distribuir todos los objetos según el centroide más cercano [4, 5]. El proceso se repite una y otra vez hasta que ya no haya cambio en los grupos de un paso al siguiente.

Conviene puntualizar lo que se entiende por un centroide y que el criterio de cercanía está basado en la distancia euclidiana, a menos que se señale otra cosa [5, 12]. La idea más importante es que a cada grupo se le puede asignar un centroide, que corresponde al promedio de los datos de cada grupo. A continuación se presenta la definición de centroide.

Definición 1 (Centroide) *El centro o centroide de una clase π_i se define como:*

$$c_i = \frac{1}{n_i} \sum_{v \in \pi_i} a_v$$

donde n_i es el número de elementos de la clase $\pi_i = \{v | a_v$ pertenece al grupo $i\}$.

Otro concepto a tomar en cuenta es la métrica, con ella se determina si un punto del conjunto de datos está más próximo a uno de los centroides, en tal caso ese punto pertenecerá al grupo de dicho centroide, la métrica utilizada para este trabajo es la euclidiana que se define de la manera siguiente[4].

Definición 2 (Distancia Euclidiana entre dos puntos) *En el espacio euclidiano la distancia entre dos puntos $X = (x_1, x_2, \ldots, x_n)$ y $Y = (y_1, y_2, \ldots, y_n)$ se define así:*

$$d_E(X, Y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

Un ejemplo sencillo de aplicación del algoritmo es el siguiente. Se toma un conjunto de datos en $\mathbb{R}^2$, en donde a sus coordenadas (x, y) se les conoce como atributos de los datos (ver Figura 1). Para encontrar los centroides, seguimos los pasos siguientes[5]:

1. Se escoge k centroides iniciales de manera arbitraria ($k = 4$)
2. Se forman cuatro grupos de la forma siguiente: se toman las distancias de un dato a los k centroides y ese dato pertenecerá al grupo del centroide más cercano, así se construyen los grupos.
3. Se toma el promedio de cada grupo y estos serán los centroides siguientes.
4. Se repiten los pasos 2 y 3 hasta que los centroides ya no cambian.

Con el ejemplo anterior se ha descrito de una manera simple el funcionamiento del algoritmo KMeans. Ahora se está en condiciones de mostrar el algoritmo, que de acuerdo a la literatura consultada [5], está definido de la siguiente manera.

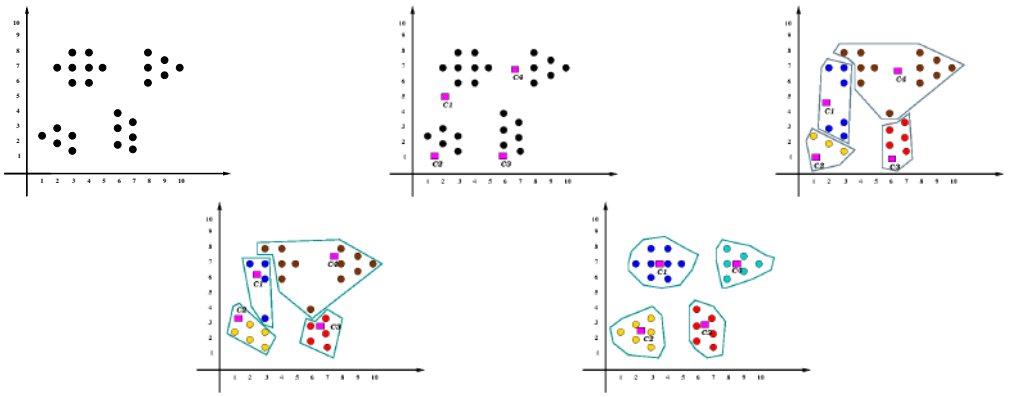


Figura 1: Ejemplo de conjuntos de puntos en el plano y su proceso de agrupamiento con KMeans. Fuente: Elaboración propia.

Paso 1. Inicialización: Se define un conjunto de datos a particionar, el número de grupos y un centroide por cada grupo de forma arbitraria o aleatoria.

Paso 2. Clasificación: Para cada dato del conjunto, se calcula su distancia a cada centroide, se determina el centroide más cercano, y el conjunto es incorporado al grupo relacionado con ese centroide.

Paso 3. Cálculo de centroides: Para cada grupo generado en el paso anterior se vuelve a calcular su centroide.

Paso 4. Condición de convergencia: El algoritmo converge cuando no existe un intercambio de objetos entre los grupos, si la condición de convergencia no se satisface, se repiten los pasos 2 y 3 del algoritmo.

2.2. Diagramas de Voronoi

Definición 3 (Diagramas de Voronoi) *El diagrama de Voronoi de* $P = \{p_1, p_2, \ldots, p_n\}$*, está formado por todos los puntos* x *que equidistan de los dos elementos de* P *más cercanos a* x*.*

En la Figura 2 se muestran ejemplos de diagramas de Voronoi en el plano: para dos, tres y cuatro puntos, siendo las lineas rojas los diagramas de Voronoi.

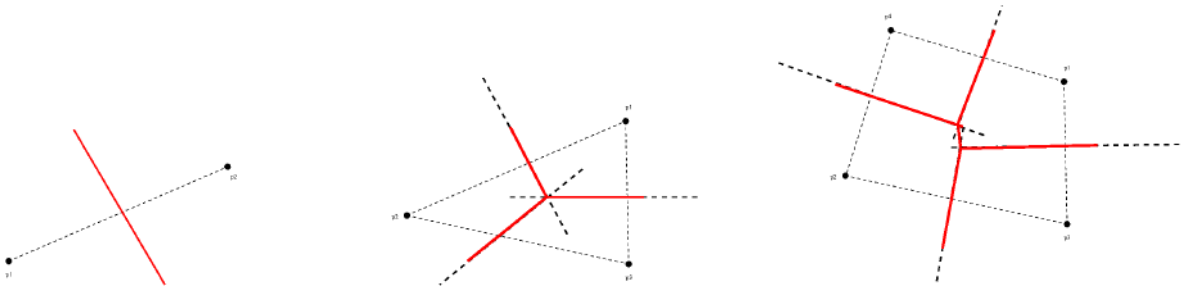


Figura 2: Ejemplos de los diagramas de Voronoi en el plano. Fuente: Elaboración propia.

Cada punto $p_i \in P$ determina una región en el plano, a saber los puntos del plano que están más cerca de p_i que del resto de los puntos de P; a esto se le conoce como *Celdas de Voronoi* y se definen de la manera siguiente.

Definición 4 (Celdas de Voronoi) *Sea* $p_i \in P$

$$V(p_i) = \{p|\ d(p, p_i) \leq d(p, p_j),\ p_j \in P\}$$

2.3. Envoltura Convexa

Definición 5 (Envoltura convexa) *Se llama envoltura convexa de un conjunto dado* X *al menor (por inclusión) conjunto convexo que contiene a* X *y se define como:*

$$conv(X) = \{tx + (1-t)y | x, y \in X, t \in [0, 1]$$

si $X \subseteq P$, y P es un conjunto convexo, entonces $X \subseteq conv(X) \subseteq P$.

Las Figura 3 representa un ejemplo de envolturas convexas de varios conjuntos de puntos en el plano.

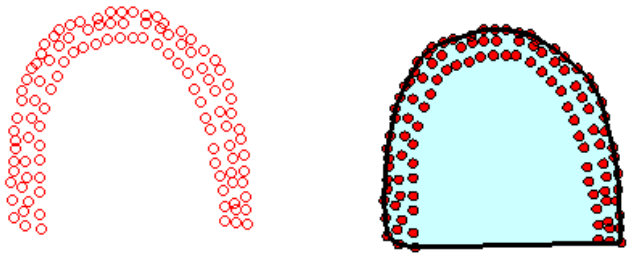


Figura 3: Ejemplos de la cascara convexa de puntos en el plano. Fuente: Elaboración propia.

3 Objetivos

3.1. Objetivo General

- Estudiar el problema de agrupamiento de datos utilizando el algoritmo KMeans

3.2. Objetivos Específicos

- Revisar literatura especializada sobre el algoritmo KMeans.
- Caracterizar el algoritmo KMeans.
- Estudiar la idea geométrica del algoritmo.
- Presentar el marco de trabajo llamado ApacheSpark.

4 Metodología

En esta sección se presenta la metodología a utilizar para llevar a cabo este trabajo. Se describe el proceso de la búsqueda de bibliografía y la idea del algoritmo. El algoritmo tiene la característica de ser simple, compacto y fácil de implementar computacionalmente. Sin embargo no está exento de limitaciones, es por eso que se hace la siguiente pregunta ¿Cuando funciona el algoritmo?, y se da una respuesta en base a la idea geométrica del mismo. Así mismo se presenta el marco de trabajo computacional en el cual se basará el trabajo, el llamado Apache Spark.

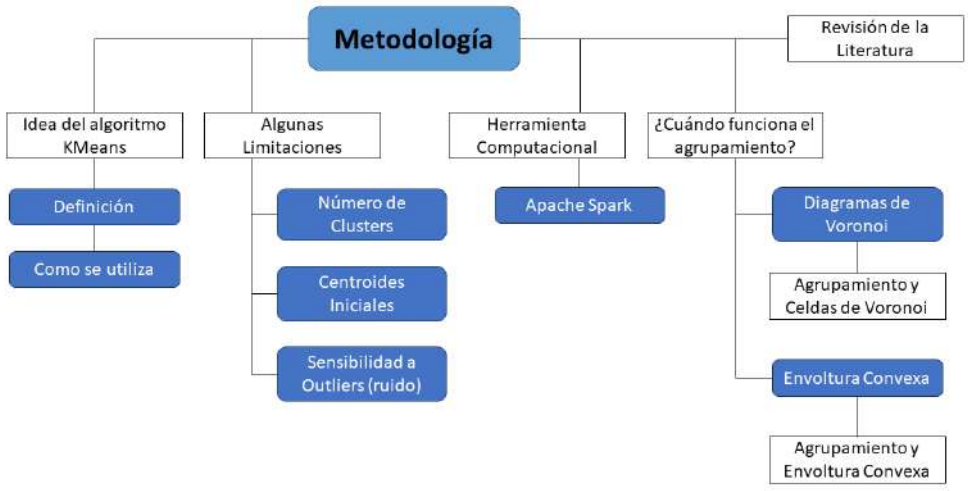


Figura 4: Metodología de trabajo para el desarrollo de esta investigación. Fuente: Elaboración propia.

4.1. Revisión de la literatura

La revisión de la literatura para esta investigación se realiza utilizando los motores de búsqueda especializados como Google Académico, Journal of citation of reports (JCR) y otros como Latinindex. La busqueda se realiza mediante los criterios siguientes:

- Frases claves en inglés como *Unsupervised algorithms*, *kmeans aplications*. *paralel kmeans*.
- Análisis de la información centrado en las algoritmos no supervisados de tipo particionales, enfocándose en aspectos tales como: (i) geometrías, (ii) centroides iniciales y (iii) posibles aplicaciones.
- Se indagó sobre el problema de agrupamiento poco estudiadas como lo son los algoritmos en paralelo y algoritmos robustos.

4.2. La idea del KMeans

El problema es encontrar grupos en un conjunto de datos. Una pregunta que surge de manera natural es la siguiente: ¿Cómo encontrar grupos en un conjunto de datos óptimos de manera automatizada? Veámoslo por medio de un ejemplo.

Tenemos un conjunto de datos (Tabla 1), cada dato se representa como un punto en el plano cartesiano con coordenadas (x, y), y lo que se quiere realizar con dicho conjunto de datos, es encontrar grupos de manera automatizada.

Tabla 1: Representación de los datos en coordenadas (x, y).

x	1	2	2	3	3	3	2	3	3	3	4	4	4	5	6	6	6	7	7	8	8	8	9	9	10
y	2.5	2.3	1.5	2.5	3.5	7	6	7	8	6	6	8	7	2	3	4	2.5	3.5	6	7	8	6.5	7.5	7	7

La representación de los puntos de la Tabla 1 en el plano cartesiano es la siguiente:

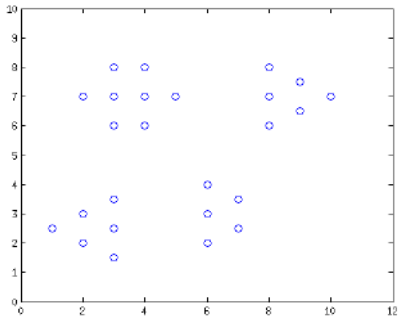


Figura 5: Representación gráfica del conjunto de datos en el plano de la forma (x, y) de la Tabla 1.

En la Figura 5 es fácil localizar visualmente cuatro grupos. Sin embargo, aún en dos dimensiones no siempre está claro cuántos cluster hay y cómo se agrupan los datos. Por ejemplo, del siguiente conjunto de datos, se requiere encontrar los grupos asociados a ellos.

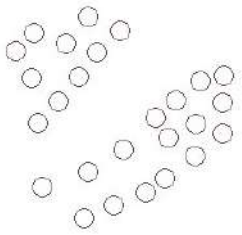


Figura 6: Conjunto de datos a agrupar.

De la Figura 6, se pueden citar aquí cuatro opciones distintas de agrupar los datos que se observan a continuación:

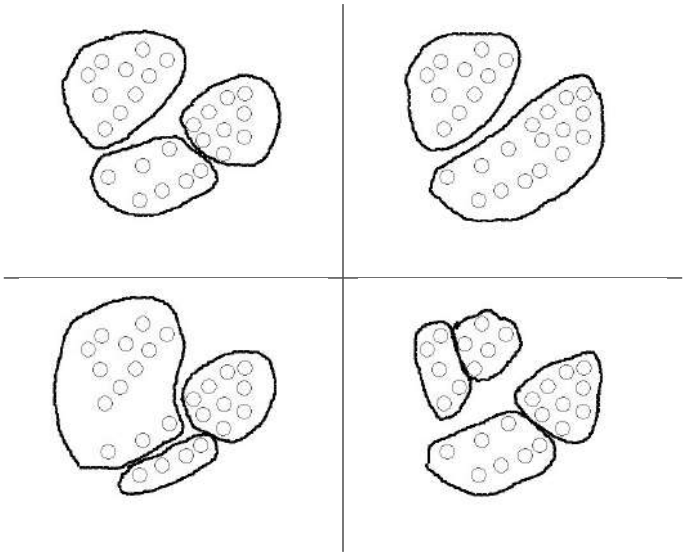


Figura 7: Distintos agrupamientos.

Para problemas de mayor dimensión, encontrar grupos se vuelve mucho más complicado, porque no podemos apoyarnos de la parte visual, como en los casos de dos dimensiones $\mathbb{R}^2$ y en los de tres dimensiones $\mathbb{R}^3$.

4.3. Ejemplo

En el artículo [12] se plantea un ejemplo numérico del algoritmo, veamos este ejemplo. Se supone en él que se tienen cuatro tipos de medicamentos con dos atributos cada uno a saber (Peso e Índice PH) como se muestra en la tabla siguiente:

Medicina	Peso	Índice PH
A	1	1
B	2	1
C	4	3
D	5	4

Se quieren encontrar dos grupos basados en los dos atributos de los datos. Cada medicamento puede ser representado como un punto en el plano cartesiano de la manera siguiente:

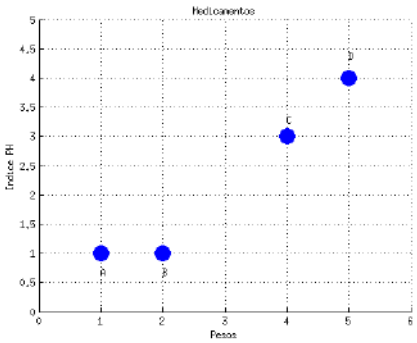


Figura 8: Representación gráfica de los atributos de los medicamentos en $\mathbb{R}^2$. Fuente Elaboración propia.

Iteración 1

Se elige el número de grupos que se desea encontrar, es decir el valor del número k, en este caso $k = 2$. Se seleccionan dos centroides iniciales, que en este caso serán los primeros dos medicamentos, es decir:

$$c_1 = (1, 1) \qquad c_2(2, 1)$$

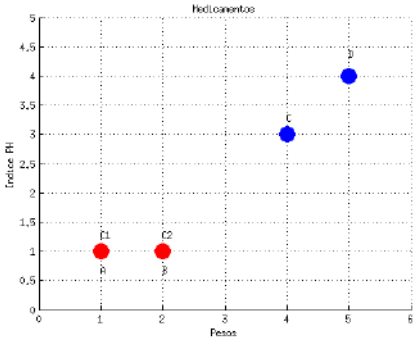


Figura 9: Elección de los centroides iniciales, en este caso son los primeros dos medicamentos que están coloreados de color rojo. Fuente: Elaboración propia.

Para encontrar los grupos se calcula las distancias de los datos a los centroides c_1 y c_2, tomando la distancia mínima para asignarlos al centroide más cercano.

Los nuevos grupos son $G_1 = \{A\}$ y $G_2 = \{B,\ C,\ D\}$

Iteración 2

Tabla 2

	c_1	c_2
A	0	1
B	1	0
C	3.61	2.83
D	5	4.24

En el grupo 1 solo se tiene a un miembro, el centroide de ese grupo será nuevamente

$$c_1 = (1, 1)$$

Mientras en el grupo 2, se tienen tres miembros que son $B(2, 1)$, $C(4, 3)$, $D(5, 4)$
Para calcular c_2 lo hacemos de la manera siguiente

$$c_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3}\right) = \left(\frac{11}{3}, \frac{8}{3}\right)$$

Por lo tanto los nuevos centroides son: $c_1 = (1, 1)$ y $c_2 = (\frac{11}{3}, \frac{8}{3})$

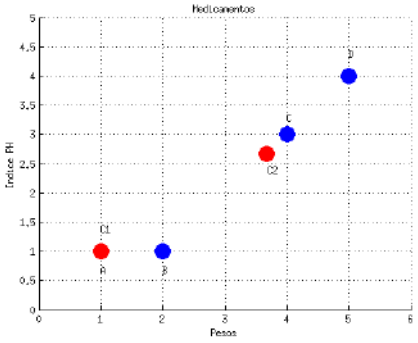


Figura 10: Calcular los nuevos centroides c_1 y c_2 calculando la distancia de ellos a todos los datos y en base a la distancia mínima. Fuente: Elaboración propia.

Se vuelven a encontrar nuevos grupos calculando las distancias de los datos a los centroides c_1 y c_2

	c_1	c_2
A	0	3.14
B	1	2.36
C	3.61	0.47
D	5	1.89

Los nuevos grupos son $G_1 = \{A,\ B\}$ y $G_2 = \{C,\ D\}$

Iteración 3

El grupo 1 tiene como miembros $A = (1, 1)$, $B = (2, 1)$
Para calcular c_1 lo hacemos de la manera siguiente

$$c_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = \left(\frac{3}{2}, 1\right)$$

El grupo 2 tiene dos miembros que son $C = (4, 3)$, $D = (5, 4)$
Para calcular c_2 lo hacemos de la manera siguiente

$$c_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = \left(\frac{9}{2}, \frac{7}{2}\right)$$

Por lo tanto, los nuevos centroides son: $c_1 = \left(\frac{3}{2}, 1\right)$ y $c_2 = \left(\frac{9}{2}, \frac{7}{2}\right)$

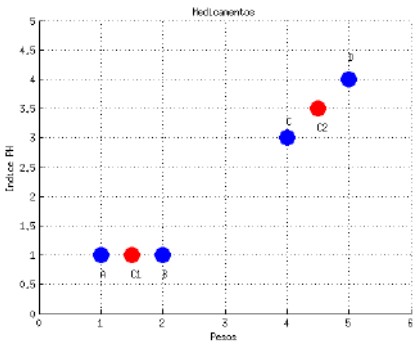


Figura 11: Elección de los nuevos centroides en base al procedimiento basado en la Iteración 2. Fuente: Elaboración propia.

Se vuelve a encontrar nuevos grupos calculando las distancias de los datos a los centroides c_1 y c_2

	c_1	c_2
A	0.5	4.30
B	0.5	3.54
C	3.20	0.71
D	4.61	0.71

Nuevos grupos $G_1 = \{A,\ B\}$ y $G_2 = \{C,\ D\}$

Es fácil ver que en el paso siguiente los centroides no cambian, por lo tanto tampoco los grupos, de tal forma que el proceso termina.

Medicina	Peso	Índice PH	Grupo
A	1	1	1
B	2	1	1
C	4	3	2
D	5	4	2

Los medicamentos A y B difieren en una unidad en la propiedad de peso, mientras que en la propiedad del índice PH son iguales, por esa razón son muy parecidos y el algoritmo KMeans los clasifica en un mismo grupo, mientras que los medicamentos C y D difieren en una unidad en ambas propiedades, quedando así en el segundo grupo. Como puede observarse los medicamentos con características similares pertenecen al mismo grupo.

Con el ejemplo se ha mostrado el funcionamiento del algoritmo KMeans, en donde cada medicamento se representa como un punto en el plano $\mathbb{R}^2$. Se pudo graficar cada dato debido a la naturaleza de los mismos. En el caso en donde los datos son de mayor dimensión, esa representación visual no puede auxiliarnos, esto no es un impedimento ya que el algoritmo puede realizar agrupamientos en cualquier dimensión.

4.4. ¿Cuándo funciona?

Se presentó la idea del algoritmo KMeans, su definición y se mostró un ejemplo de como se utiliza, para encontrar grupos de conjuntos de datos. KMeans encuentra buenos agrupamientos y malos agrupamientos también, la cuestión es ¿Cuándo hace buenos agrupamientos?

A continuación se presenta un ejemplo donde el algoritmo hace un buen agrupamiento y otro ejemplo donde hace un mal agrupamiento.

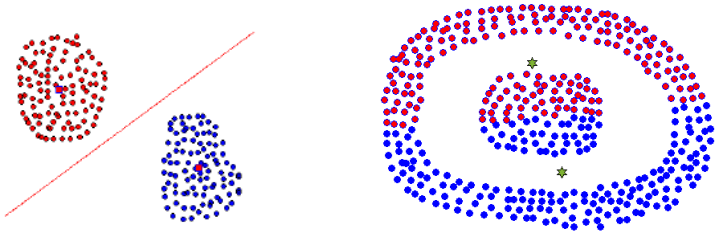


Figura 12: Ejemplos en $\mathbb{R}^2$ donde el algoritmo KMeans agrupa bien y otro ejemplo donde agrupa mal. Fuente: elaboración propia.

En el caso del buen agrupamiento ver Figura 12 es posible trazar una línea recta que divide al plano por regiones, cada una de estas regiones contendrá a uno de los grupos. Por lo contrario, no

será posible trazar una línea recta con las características mencionadas. Esta es una característica relevante para dar respuesta a esta cuestión.

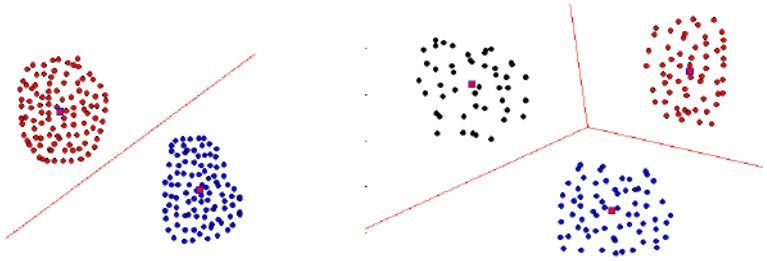


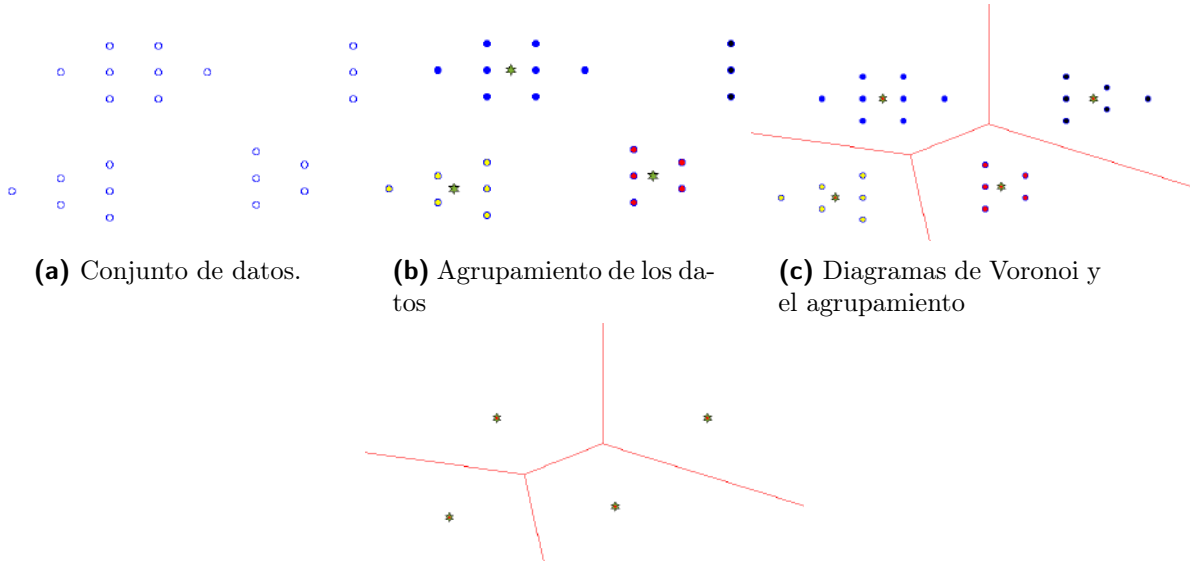
Figura 13: Ejemplos de buenos agrupamientos porque los datos están linealmente separados. Fuente: Elaboración propia

Estos conjuntos se consideran linealmente separables en el plano, dada la existencia de una recta tal que cada grupo incluido en el semiplano diferente, como se observa en la Figura13 a). En el conjunto $\mathbb{R}^n$ se dice que dos conjuntos son linealmente separables si existe un hiperplano tal que cada grupo cae en uno de los subconjuntos determinados dicho hiperplano.

Una colección de estos grupos es linealmente separable, si cada par de ellos es linealmente separable, ver Figura13 b). Esta idea se puede observar más fácilmente con diagramas de Voronoi.

4.5. Agrupamiento y Celdas de Voronoi

Si se considera el diagrama de Voronoi de los centroides de un agrupamiento, a este diagrama lo llamamos *diagrama de Voronoi del agrupamiento*. Se considera como ejemplo el grupo de datos siguiente, el agrupamiento de datos, el agrupamiento junto con el diagrama de Voronoi y por ultimo el diagrama de Voronoi de este agrupamiento, con respecto a sus centroides, Figura 14.



(a) Conjunto de datos.

(b) Agrupamiento de los datos

(c) Diagramas de Voronoi y el agrupamiento

(d) Diagrama de Voronoi del agrupamiento

Figura 14: Buenos agrupamientos y su diagrama de Voronoi. Fuente: Elaboración propia.

Cuando el algoritmo KMeans proporciona un mal agrupamiento, es fácil observarlo mediante los diagramas de Voronoi. Se puede observar un ejemplo de mal agrupamiento en la Figura 15 (b), que corresponde al conjunto de datos a agrupar con KMeans de la Figura 14 (a).

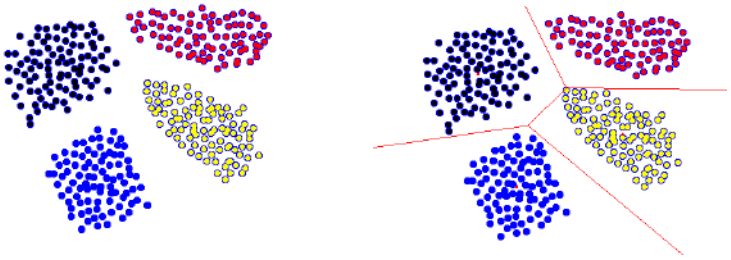
(a) Conjunto de datos a agrupar.

(b) Agrupamiento de los datos con KMeans.

(c) Agrupamiento de datos y diagramas de Voronoi.

Figura 15: Malos agrupamientos y su diagrama de Voronoi. Fuente: Elaboración propia.

En la Figura 16 (a) se muestra el mismo conjunto de datos, pero con un buen agrupamiento con KMeans. Ver su diagrama de Voronoi en la Figura 16 (b).



(a) Conjunto de datos bien agrupados con KMeans

(b) Buen agrupamiento y su diagrama de Voronoi

Figura 16

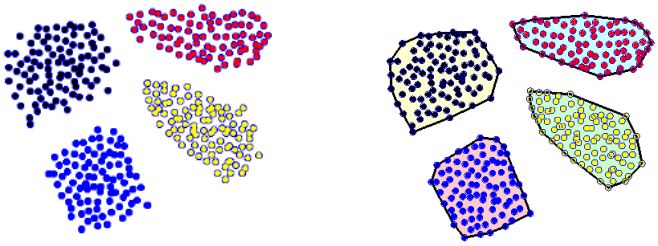
Los grupos que produce KMeans siempre están en distintas celdas de Voronoi generadas por los centroides del agrupamiento [13]. Por lo tanto, KMeans agrupa bien cuando los grupos están en diferentes celdas de Voronoi, o bien cuando están linealmente separados dos a dos.

Otra forma de ver la separación lineal es mediante la envoltura convexa (*convex hull*) de los conjuntos que conforman el agrupamiento.

4.6. Agrupamiento y envoltura convexa

Es evidente observar que dos conjuntos de puntos son linealmente separables si su cascaras convexas no se intersectan. Es decir, dado un agrupamiento, se considera linealmente separable si estas envolturas de sus grupos no se intersectan dos a dos.

Se considera los siguientes 4 grupos y sus envolturas correspondientes. Estos grupos están linealmente separados y sus envolturas convexas no se intersectan dos a dos. Ver Figura 17



(a) Buen agrupamiento con KMeans

(b) Buen agrupamiento y su envoltura convexa

Figura 17: Ejemplo de conjunto de puntos en el plano que son linealmente separables. Fuente: Elaboración propia.

5 Algunas limitaciones

A pesar de ser utilizado ampliamente en una gama de aplicaciones, el algoritmo KMeans no está exento de limitaciones. Algunos de estos inconvenientes han sido ampliamente descritos en la literatura [9], y son los siguientes:

Número de clusters: El número de clusters tiene que ser conocido por el algoritmo en su inicialización. Por lo tanto, k tiene que ser un parámetro de entrada del algoritmo.

Centroides iniciales: Sensibilidad a la inicialización de los centroides.

Sensibilidad a outliers (ruido): La existencia de un conjunto pequeño de datos que distan mucho de las clases afecta el desempeño del algoritmo.

6 El Marco de trabajo llamado Apache Spark

El rápido aumento de usuarios de internet en los últimos años, combinado con la gran cantidad de datos que se generan, ha creado la necesidad de servicios, herramientas para proporcionar y procesar de manera confiable grandes conjuntos de datos. La computación en la nube cumple este propósito como un conjunto compartido de recursos informáticos (redes, servidores, almacenamiento, aplicaciones, servicios, etc.) que se puede alimentar e interactuar inmediatamente utilizando el concepto de computación en la nube, que permite el acceso a internet bajo demanda a la red [10].

Uno de los marcos más utilizados para aplicaciones de análisis de datos basados en la nube es Apache Spark [3, 8]. Es una potente plataforma informática Hadoop de código abierto escrita en el lenguaje de programación Scala. Como Apache Spark es una infraestructura de propósito general para clústeres de computadoras, se utiliza principalmente para diversas aplicaciones. Además, el desarrollo de infraestructuras de computación en la nube ha llevado a la creación de nuevas técnicas y algoritmos para el procesamiento y análisis de datos, así como bibliotecas de aprendizaje automático destinadas a explotar su potencial [11].

Desde el punto de vista del aprendizaje automático, Apache Spark proporciona a los usuarios técnicas tradicionales, como clasificación, agrupamiento y regresión, que constituyen la llamada biblioteca MLlib [7]. Este marco de trabajo se caracteriza por altas velocidades de ejecución, paralelización de tareas y utilización de la memoria caché del sistema al implementar diferentes algoritmos.

Por lo descrito anteriormente, es de gran interés en esta investigación implantar y conocer este tipo de herramientas computacionales dada la necesidad constante de obtener información oculta en los datos y tener mayor conocimiento de algún problema en particular haciendo que los mismos datos de problema, muestren indicios o correlaciones importantes para encontrar los grupos de datos de interés.

7 Consideraciones finales

KMeans es un algoritmo heurístico que ha demostrado que tiene relación con otros métodos de agrupamiento como factorizaciones matriciales no negativas y análisis de componentes principales, por ejemplo. Este último será de especial interés en este trabajo en desarrollo.

El análisis de componentes principales (PCA) es una técnica estadística ampliamente utilizada para la reducción de dimensiones no supervisada. La agrupación de KMeans es una agrupación de datos de uso común para tareas de aprendizaje no supervisadas. Uno de los objetivos de este trabajo es mostrar que los componentes principales pueden ser una solución óptima a la inicialización de los centroides para la agrupación de KMeans, tal como lo indican diversos autores [1, 6] y que es un punto inicial del cual se partirá para seguir con este trabajo..

En esta investigación se estudiará la conexión entre estos dos métodos ampliamente utilizados. Se estudiará si la reducción de dimensiones no supervisada está estrechamente relacionada con el aprendizaje no supervisado.

Por otro lado, los datos de alta dimensión a menudo se transforman en datos de menor dimensión mediante el análisis de componentes principales (PCA) (o descomposición de valores singulares), donde se pueden detectar patrones coherentes con mayor claridad. Esta reducción de dimensiones no supervisada se utiliza en áreas muy amplias como la meteorología, el procesamiento de imágenes, el análisis genómico y la recuperación de información. También es común que se utilice PCA para proyectar datos a un subespacio de dimensiones inferiores y luego se aplique KMeans en el subespacio, pero esto es un trabajo futuro en este trabajo inicial y sustentado en diversas publicaciones base para su desarrollo [1, 2].

Por último se implementará una metodología basada en el cómputo en la nube. Implantando un marco de trabajo llamado Apache Spark para el trabajo con grandes cantidades de datos implantándolo en un cluster de computadoras personales y que será del tipo Beowulf para el alto desempeño de cálculo numérico. Es por eso que en este trabajo se implantará un cluster de computadoras instalando este marco de trabajo para trabajar con alta densidad de datos y poder estudiar el problema de los centróides iniciales usando la técnica de PCA dado que se espera que esta técnica proporcione centroides iniciales óptimos para realizar buenos agrupamientos con KMeans.

Referencias

[1] J. Chen, J. Zhu, H. Jiang, H. Yang, and F. Nie. Sparsity fuzzy c-means clustering with principal component analysis embedding. *IEEE Transactions on Fuzzy Systems*, 2022.

[2] V. Divya and K. N. Devi. An efficient approach to determine number of clusters using principal component analysis. In *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, pages 1–6. IEEE, 2018.

[3] E. Dritsas, I. E. Livieris, K. Giotopoulos, and L. Theodorakopoulos. An apache spark implementation for graph-based hashtag sentiment classification on twitter. In *Proceedings of the 22nd Pan-Hellenic Conference on Informatics*, pages 255–260, 2018.

[4] L. Elden. *Matrix Methods in Data Mining and Pattern Recognition*. SIAM-Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition, 2019.

[5] G. Gan, C. Ma, and J. Wu. *Data clustering: theory, algorithms, and applications*, volume 20. Society for Industrial and Applied Mathematics, 2007.

[6] S. Lu, H. Yu, X. Wang, Q. Zhang, F. Li, Z. Liu, and F. Ning. Clustering method of raw meal composition based on pca and kmeans. In *2018 37th Chinese control conference (CCC)*, pages 9007–9010. IEEE, 2018.

[7] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, et al. Mllib: Machine learning in apache spark. *The journal of machine learning research*, 17(1):1235–1241, 2016.

[8] N. Ntaliakouras, G. Vonitsanos, A. Kanavos, and E. Dritsas. An apache spark methodology for forecasting tourism demand in greece. In *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pages 1–5. IEEE, 2019.

[9] J. Pérez, M. Henriques, R. Pazos, L. Cruz, G. Reyes, J. Salinas, and A. Mexicano. Mejora al algoritmo de agrupamiento k-means mediante un nuevo criterio de convergencia y su aplicación a bases de datos poblacionales de cáncer. *II Taller Latino Iberoamericano de Investigación de Operaciones*, 2007.

[10] A. Rashid and A. Chaturvedi. Cloud computing characteristics and services: a brief review. *International Journal of Computer Sciences and Engineering*, 7(2):421–426, 2019.

[11] S. Salloum, R. Dautov, X. Chen, P. X. Peng, and J. Z. Huang. Big data analytics on apache spark. *International Journal of Data Science and Analytics*, 1:145–164, 2016.

[12] K. Teknomo. K-means clustering tutorial. *Medicine*, 100(4):3, 2006.

[13] Wikipedia. Kmeans. http://es.wikipedia.org/wiki/K-means.

Simulación numérica sobre el efecto del plasmón localizado de nanopartículas de plata (Ag), bajo el marco de la teoría de Mie

José Luis Fraga Almanza*[1], Carlos Eduardo Rodríguez García[1], Elmer Cruz Mendoza[2], Adianez Arhely Gamboa Rivas[1] y Gael Antonio Torres Aguirre[1]

[1]Facultad de Ciencias Físico Matemáticas, UAdeC
[2]Facultad de Ciencias, UABC

Resumen

Las nanopartículas (Nps) de metales nobles como plata (Ag) y oro (Au) en las últimas décadas son objeto de extensa investigación en el contexto de las nanociencias y la nanotecnología, principalmente esto se debe a sus propiedades ópticas únicas. En las Nps, el gas de electrones libres en ellas presenta propiedades de oscilación resonante, fenómeno conocido como el plasmón superficial localizado (PSL). Al irradiarse las Nps en el ultravioleta y visible del espectro, la absorción óptica, depende fundamentalmente del material constitutivo, la geometría de la Nps y su entorno. La dispersión de la radiación electromagnética de una esfera, es definida por la Teoría de Mie. Los cálculos al usar esta teoría pueden llegar a ser largos e incluso imposibles para aquellos con recursos hardware limitados. Por lo tanto en esta investigación que actualmente, se encuentra en desarrolló, se diseño e implanto una simulación numérica escrita en Python 3 para la obtención de los coeficientes de Mie en nanopartículas de plata (Ag) y contar con una herramienta computacional que generé escenarios permitiendo el estudio y obtención de los coeficientes de extinción de NPs de Ag, siendo esta metodología fácilmente adaptable para otros metales donde se pueden calcular y/o simular una variedad de propiedades ópticas.

Palabras clave: Nanoestructuras; Nanopartículas; Plasmones; Resonancia-Superficial; Teoría de Mie

1 Introducción

El fácil acceso a los dispositivos tecnológicos debido a su bajo costo, a la necesidad de estar comunicados y por el cambio radical en la dinámica poblacional ha iniciado una carrera por la miniaturización en ellos. Permientiendo un desarrollo en áreas como la electrónica y la nanoelectrónica [9], donde su configuración física es cada vez más pequeña (debido a nanoestructuras y nanopartículas). Es por ello, que en trabajos teóricos y experimentales, la reducción en el tamaño en su estructura da lugar a cambios a sus propiedades, dando

*josefraga@uadec.edu.mx

lugar a nuevas metodologías en la investigación científica en áreas como la Nanociencia y Nanotecnología [4].

Las nanoestructuras como partículas con dimensiones nanométricas, dependen de su tamaño debido a que sus propieades en esta escala cambian de forma radical, tal como lo indican diversos autores [6]. Es decir, las nanoestructuras son nanopartículas con geometrías diversas tales como nanoesferas, nanocubos, nanobarras, llegando a tener otras geometrías más complejas y notables como nanoestrellas, nanoflores, nanodonas, por mencionar algunas [20]. Estas nanopartículas tienen potenciales aplicaciones en la tecnología actual. Como ejemplo, se puede mencionar a las nanoesferas de Ag (plata) que poseen propiedades antimicrobianas y son utilizadas en vendajes médicos [7]. Las nanodonas de NiO (óxido de Níquel) son otro ejemplo, donde se utilizan como componentes de un sensor electroquímico para medir la glucosa teniendo bueno resultados [3], comparados con un sensor comercial. Es decir, el estudio de las nanoestructuras es una área fértil en las nanociencias que da la posibilidad de ser utilizadas en la nanotecnología en beneficio de la humanidad.

En la Figura 1 se muestra nanoestructuras con distintas geometrías. El estudio de éstas indican que en investigaciones recientes se han encontrado geometrías exóticas como las nanodonas y nanocintas de Moebius que son nanoestructuras singulares que existen experimentalmente [14, 3]. Este tipo de nanoestructuras son objeto de estudio y de exploración profunda debido a la presencia de sus propiedades atípicas con el objetivo de tener un mayor entendimiento de éstas para posibles aplicaciones tecnológicas.

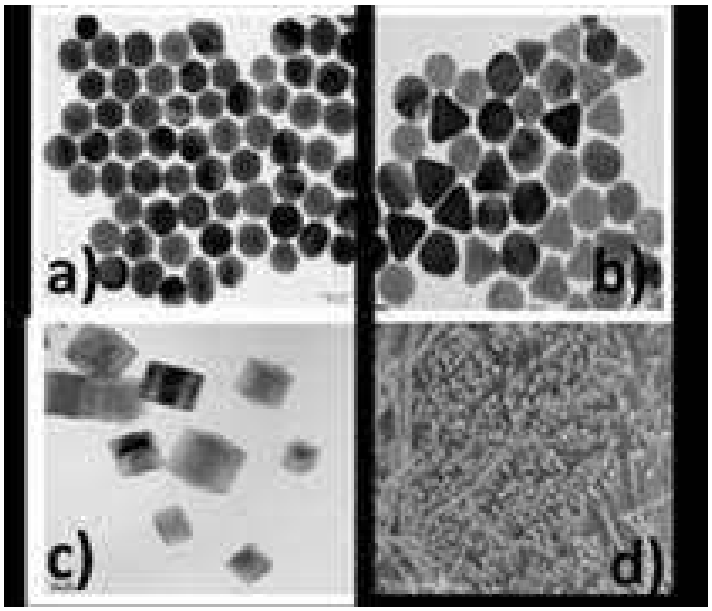


Figura 1: Micrografías de nanopartículas: a) nanoesferas, b) nanotriángulos con nanoesferas, c) nanocubos d) nanotubos. Fuente: [27]

Un efecto que ha tenido gran relevancia en las últimas décadas es la Resonancia de Plasmones Localizados de Superficie (RPLS), éstas son exitaciones colectivas de la densidad de carga presentes en estructuras con alta densidad electrónica, tales como nanopartículas y nanoestructuras metálicas. Se pueden mencionar

investigaciones de nanoestructuras metálicas sobre metales nobles como la plata (Ag) y el oro (Au) [23]. El estudio y entendimiento de dicho efecto (RPLS) sobre nanoestructuras metálicas en sus propiedades ha aperturado un gran número de interesantes aplicaciones tecnológicas, tal como lo indica (Loiseau y otros, 2019) en la fabricación de sensores ultra sensibles, en (Swami Muddineti y otros, 2015) muestran fabricación de fuentes de radiación térmica localizada para la eliminación de células cancerígenes en tejido vivo, celdas solares con mayor eficiencia y el mejoramiento de espectroscopías electrónicas [16].

Dada la relevancia de las propiedades de las nanopartículas metálicas en distintas áreas de la ciencia, en este trabajo se presenta un avance en el desarrollo de una aplicación móvil para el cálculo de absorbancias de nanopartículas de plata (Ag) haciendo evidente la respuesta óptica de la resonacia del plasmón superficial para una sola nanopartícula de forma esférica, en el marco de la teoría de Mie.

2 Antecedentes

Uno de los eventos más importantes en la historia de la ciencia se dió cuando, en siglo XIX, el físico escocés James Clerk Maxwell proporcionó las bases teóricas del electromagnetismo, en donde se establece la íntima relación entre los fenómenos eléctricos con los magnéticos [8]. Esta conexión se resume a través del conjunto de ecuaciones conocidas como las ecuaciones de Maxwell definidas como:

$$\nabla \cdot \vec{D} = \rho_{ext}, \tag{1}$$

$$\nabla \cdot \vec{B} = 0, \tag{2}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \tag{3}$$

$$\nabla \times \vec{H} = \vec{J}_{ext} + \frac{\partial \vec{D}}{\partial t} \tag{4}$$

las cuales, junto con la ecuación de conservación de la carga interna

$$\nabla \cdot \vec{J} = \frac{\partial \rho_{int}}{\partial t} \tag{5}$$

caracterizan completamente el electromagnetismo en medios materiales, ya que relacionan los cuatro campos macroscópicos $\vec{D}$ (el vector de desplazamiento eléctrico), $\vec{E}$ (el campo eléctrico), $\vec{B}$ (la inducción magnética o densidad de flujo magnético) y $\vec{H}$ (el campo magnético).

La ecuación (1) es conocida como la ley de Gauss para el vector de desplazamiento y establece que la forma e intensidad de las líneas de dicho campo son definidas por la distribución de cargas externas caracterizada por ρ_{ext}. Esta ecuación contiene de forma implícita la ley de Coulomb, la cual establece la relación de la fuerza entre partículas cargadas. Por otro lado, la ecuación (2) es la ley de Gauss para el campo $\vec{B}$ y establece que las líneas de dicho campo son cerradas. Por lo tanto, afirma la inexistencia del monopolo magnético. La ecuación (3) es conocida como la ley de Maxwell-Faraday y establece que la existencia de campos magnéticos variables en el tiempo dan lugar a campos eléctricos transversales, para los cuales $\nabla \times \vec{E} \neq 0$. Finalmente, la ecuación (4) predice que la presencia de densidades de corriente externas, $\vec{J}_{ext}$,

y de variaciones temporales en la densidad del flujo eléctrico producen campos magnéticos transversales, tal que $\nabla \times \vec{H} \neq 0$. Como ejemplo de esto, considere la corriente eléctrica que pasa por un alambre largo y recto, la cual da lugar a líneas de campo magnético que son círculos concéntricos al eje del alambre, y donde el vector de campo magnético es tangencial a dichas líneas de campo.

Uno de los éxitos más importantes de la teoría de Maxwell es la predicción de la existencia de ondas electromagnéticas generadas por la coexistencia de campos eléctricos y magnéticos que oscilan en el tiempo con frecuencia y los cuales son transversales a la dirección de propagación, tal como se muestra en la Figura 2. Dichas ondas se propagan exactamente a la velocidad de la luz. Cabe señalar que, haber condensado a las leyes electromagnéticas a través de expresiones matemáticas abstractas abrió la posibilidad de abordar muchos problemas desde el punto de vista teórico, lo cual complementa a los estudios experimentales.

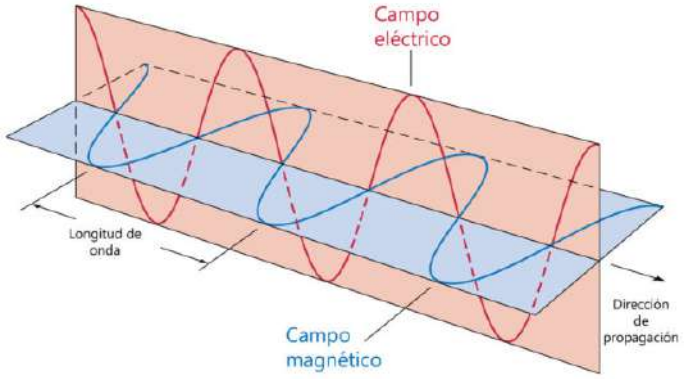


Figura 2: Esquema de la propagación de la luz como una perturbación electromagnética. Fuente: [19]

Estudios teóricos sobre la interacción de campos electromagnéticos con la materia han hecho posible la caracterización de efectos y fenómenos interesantes derivados de la interacción luz materia, en particular, cuando esta interacción involucra sistemas que poseen una alta densidad de electrones libres (tales como los metales oro, plata, cobre, aluminio). Debido a procesos de absorción de energía y mediante la interacción electrón-electrón de largo alcance se generan modos colectivos de la densidad de carga, también conocidos como *plasmones*. Al área encargada de realizar los estudios de generación y manipulación del plasmón se le conoce con el nombre de Plasmónica, la cual explora la posibilidad de confinar campos electromagnéticos en sistemas cuyas dimensiones son más pequeñas que la longitud de onda de la luz utilizada [10].

Un plasmón es una cuasipartícula que describe el campo electromagnético producido por las oscilaciones colectivas de los electrones de conducción presentes en un material [25]. Cabe mencionar que existen distintos tipos de plasmones, entre los que se distinguen por ejemplo: i) los generados dentro de un medio metálico infinito en bulto, a los cuales se le conoce como plasmones de volumen (o de bulto), ii) el plasmón propagante que se excita en una interfase metal-dieléctrico conocido como plasmón polaritón de superficie (para superficies planas consideradas infinitas) y iii) las oscilaciones colectivas de electrones restringidos a pequeños volúmenes metálicos, también conocidos como plasmones de superficie localizados [21]. Estos últimos de particular interés debido a sus potenciales aplicaciones en áreas tales como la nanofotónica,

biomedicina, espectroscopías, etc.

Para entender en qué consiste un plasmón de superficie localizado considere la siguiente situación física. Suponga un partícula metálica con forma esférica (como la mostrada en la Figura 3, sobre la cual, debido a una razón por el momento irrelevante, se induce una polarización en el sistema (flechas delgadas), separando la carga positiva (+) de la negativa (-). Sin embargo, debido a fuerzas de restitución, la carga + y la - tenderá a recombinarse produciendo fluctuaciones en la polarización caracterizadas por una frecuencia de resonancia, también conocida como frecuencia de plasma. Cabe señalar que la frecuencia del plasmón depende fuertemente de la forma y tamaño de la partícula.

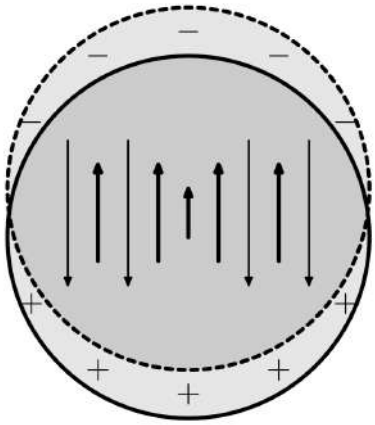


Figura 3: Representación de una esfera metálica (círculo sólido) con electrones que se desplazan a una misma distancia (flecha corta), ocupando una región desplazada (círculo punteado), produciendo cargas superficiales negativas (arriba) y positivas (abajo). El interior (oscuro) permanece neutral. La polarización (flechas estrechas) y campo eléctrico (flechas anchas) dentro de la esfera se muestran esquemáticamente. Fuente: [21]

El estudio de las propiedades ópticas de una partícula de forma esférica y estructuras esféricas más complejas comienza a principios del siglo pasado. En 1908, Gustav Mie en su artículo titulado *Consideraciones sobre la óptica de los medios turbios, especialmente soluciones coloidales* [17, 15], calculó los efectos de la dispersión de la luz por pequeñas partículas de oro partiendo de la teoría electromagnética clásica [29]. Resolvió las ecuaciones de Maxwell para el problema de una partícula metálica envuelta en un medio dieléctrico, explicó el cambio de color de los coloides de oro con el diámetro de las partículas, y que posteriormente se interpretó en función de resonancia de plasmones superficiales [26, 15].

La teoría de Mie se basa en las ecuaciones de Maxwell y da una solución completa a estas ecuaciones para calcular la dispersión de las ondas electromagnéticas en las partículas esféricas. Es necesario utilizar métodos numéricos para obtener soluciones aproximadas de estas ecuaciones cuando las geometrías en las nanoestructuras no son esféricas.

Desde el punto de vista teórico, el cambio en las propiedades de la materia con respecto al tamaño ha traído un gran impacto y esto implica nuevos retos en el campo de las nanociencias [5]. Actualmente las

nanopartículas son un área de intensa investigación científica y desde la década de los 1990 ha sido de gran interés estudiar el fenómeno de Resonancia de Plasmones Localizados de Superficie (RPLS), el cual se presenta en nanopartículas y nanoestructuras metálicas, específicamente se han estudiado los metales nobles como la Ag (plata) y el Au (oro) [24], debido a una amplia variedad de aplicaciones potenciales en campos tales como la biomedicina, óptica, electrónica, nanoquímica, agricultura [12], etc.

3 Propiedades de Ag forma nanoestructurada.

Las propiedades básicas de un material se ven alteradas por la reducción en su tamaño en la escala nanométrica. Propiedades como la forma, la morfología, el tamáño, la topología y la relación entre superficie y volumen, toman gran relevancia [22]. En los materiales nanoestructurados ocurren diversos fenómenos en la superficie y se relacionan directamente con otros, tales como, la adsorción, relajación y reconstrucción atómica del sistema [2, 22].

Las nanopartículas métalicas de oro y plata son ampliamente estudiadas debido a sus singulares propiedades eléctricas, ópticas y antimicrobianas que permiten visualizar su aplicación en dispositivos electrónicos, sensores y productos bactericidas entre muchos más.

La caracterización de nanoparticulas metálicas da lugar a incorporar diferentes tamaños y morfologías, pero es importante tener métodos sencillos que permitan analizar una muestra e identificar estos parámetros. Estas nanopartículas exhiben un espectro UV-Vis caracterizado por una banda en el visible conocida como plasmón. Propiedades como intensidad, forma y posición de este pico varía ya que es dependiente del tamaño y forma de la nanopartícula. Por lo tanto la identificación y cuantificación permite prever las propiedades morfológicas de las nanopartículas de forma indirecta.

4 Objetivo General

Estudiar teóricamente el efecto del plasmón localizado de superficie sobre las propiedades ópticas de nanoestructuras metálicas de plata con geometrías esféricas, esto a través de simulación numérica para calcular los espectros de absorbancia basada en la teoría de Mie.

4.1. Objetivos Específicos

1. Revisar la literatura para comprender el método de diferencias finitas en el estudio de plasmones en nanoestructuras metálicas (NEM).
2. Desarrollar un programa computacional para el cálculo de absorbancias de nanopartículas de Ag.
3. Obtener espectros de absorbancias de nanopartículas de plata (Ag) en forma teórica
4. Aplicar la teoría de Mie para el cálculo de la función dieléctrica de nanopartículas de plata.

5 Metodología

Para llevar a cabo los objetivos específicos, a continuación se presenta la metodología que se llevará a cabo durante el desarrollo de la investigación.

- La revisión de la literatura se hará utilizando los buscadores, SciELO, DialNet, WorldWideScience, Springer Link, CERN Document Server, Google Scholar, etc.
- En el desarrollo computacional se usarán dos lenguajes, FORTRAN y Python, por ser orientado a objetos y contar con los ecosistemas de cómputo científico necesarios, siendo uno de los más relevantes la API para el cómputo paralelo.
- Se utilizará como herramienta computacional un clúster de computadoras tipo Beowulf, para implementar la teoría de Mie y calcular los coeficientes de la misma.

El flujo de trabajo en la metodología se puede observar en la Figura 4.

6 Aspectos básicos a considerar en la teoría de Mie

Gustav Mie desarrolló por primera vez la teoría de la absorción y dispersión de la luz por una sola partícula esférica. Esta se basa en la interacción de las partículas esféricas con la luz electromagnética plana que incide sobre ellas. La teoría fue explicada con más detalle sobre una base matemática por Bohren y Huffman [1]. Aunque la teoría de Mie se basa en el modelo de una sola partícula, es posible utilizarla en la modelación de espectros de absorbancia de partículas coloidales en solución.

Cuando una onda plana monocromática incide sobre una esfera, dispersa y absorbe luz dependiendo de las propiedades de la luz y la esfera. Si la esfera está en el vacío, entonces el índice complejo de refracción de la esfera es:

$$m_{vacio} = m_{real} - im_{img} \tag{6}$$

donde $m_{img} = k$ es el índice de absorción o atenuación complejo. El parámetro de tamaño de una esfera de radio r es:

$$x_{vacio} = k \cdot r = \frac{2\pi}{\lambda} r \tag{7}$$

donde λ es la logitud de onda en el vacío. El parámetro de tamaño adimensional de una esfera incluye la longitud de onda. Esta debería ser la longitud de onda de la onda plana en el entorno, por lo tanto.

$$x_{vacio} = \frac{2\pi r}{\lambda_{vacio}/n_{entorno}}; \quad n_{entorno} = \textit{índice real del entorno} \tag{8}$$

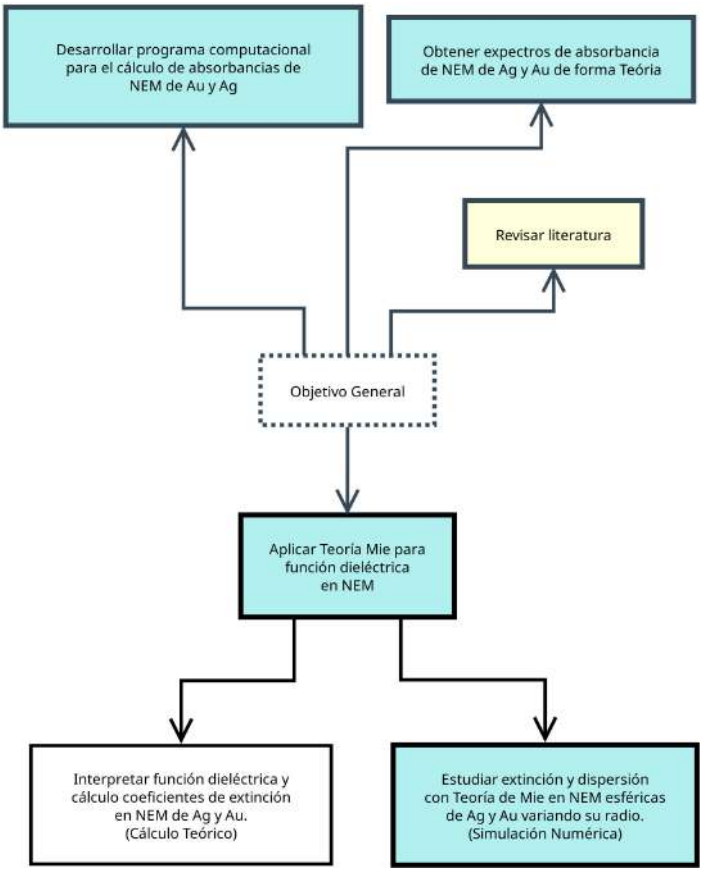


Figura 4: Diagrama de flujo de la estructura metodológica para el estudio de Nanoestructuras Metálicas (NEM). Fuente: Elaboración propia

6.1. Índices de refracción del Ag

Como se indicó en la sección (6), m es el índice de refracción relativo de la partícula en la longitud de onda (λ). La cantidad m_{img} es el índice de refracción complejo de la partícula y $n_{entorno}$ es el índice de refracción real del medio en la longitud de onda λ. Para obtener los espectros de absorbancia de nanopartículas en solución, la absorbancia debe calcularse para un rango de longitudes de onda. Es necesario conocer los valores del índice de refracción complejo de las partículas y el índice de refracción real del medio para las longitudes de onda correspondientes.

Por lo tanto, la constante dieléctrica de la plata en bulto publicada por [18] se utilizó en la simulación numérica que se presenta en la sección 7. A continuación se muestra el índice de refracción complejo del metal noble como la plata (Ag).

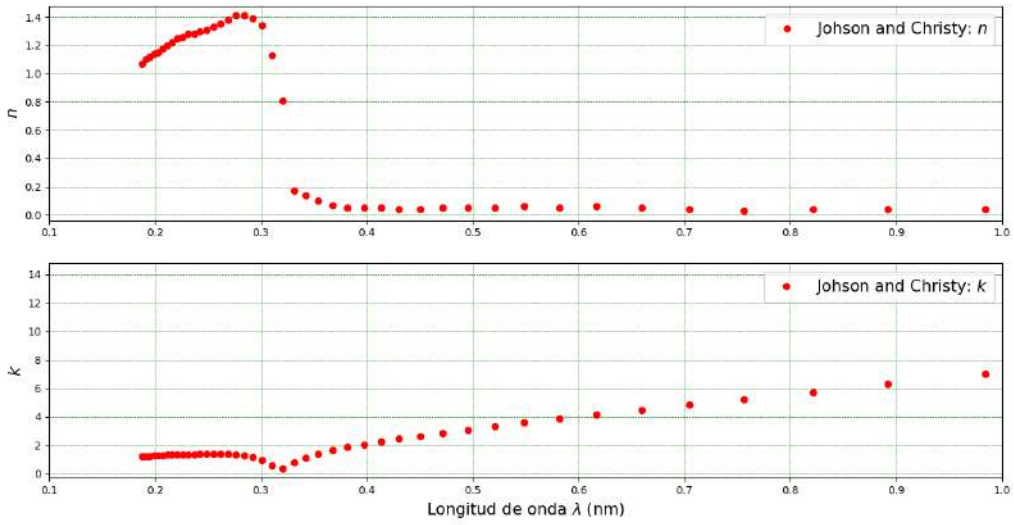


Figura 5: Índices de refracción complejo ($n + ik$) para la plata (Ag). Fuente: Elaboración propia

Por lo tanto los datos obtenidos de [18] son utilizados como parte importante para determinar el tamaño x (ver ecuación 7) de las nanopartículas de Ag en un rango establecido.

6.2. Absorbancia

La absorbancia de una solución que contiene partículas idénticas a una longitud de onda (λ) específica, está dada por:

$$A = -log_{10}\left(\frac{I}{I_0}\right)^{\frac{1}{2}} \tag{9}$$

donde I_0 e I son la intensidad de la luz incidente y transmitida que pasa a través de la solución. La relación se puede expresar aún más en términos de eficiencia de extinción Q_{ext}, camino de longitud l (típicamente con valor de 1), densidad numérica N y radio r [1]. Como se puede observar en la siguiente expresión.

$$A = \frac{\pi r^2 Q_{ext} l N}{2{,}303} \tag{10}$$

donde $Q_{ext} = \frac{C_{ext}}{\pi r^2}$, C_{ext} es el coeficiente de extinción transversal geométrico de la partícula. Es fundamental conocer el valor de la eficiencia de extinción para obtener los espectros de absorbancia de las nanopartículas en solución y se puede calcular mediante la ecuación obtenida de [28]:

$$Q_{ext} = \frac{2}{x^2} \sum_{n=1}^{\infty} (2n+1) Re(a_n + b_n) \tag{11}$$

donde

$$a_n = \frac{m\psi_n(mx)\psi_n^{'}(x) - \psi_n(x)\psi_n^{'}(mx)}{m\psi_n(mx)\xi_n^{'} - \xi_n(x)\psi_n^{'}(mx)} \tag{12}$$

$$b_n = \frac{\psi_n(mx)\psi_n^{'}(x) - m\psi_n(x)\psi_n^{'}(mx)}{\psi_n(mx)\xi_n^{'} - m\xi_n(x)\psi_n^{'}(mx)} \tag{13}$$

Aquí ψ_n y ξ_n son las funciones de Ricatti–Bessel, $x = \frac{2\pi n_m R}{\lambda}$ es el parámetro de tamaño adimensional y $m = \frac{(n+ik)}{n_{entorno}}$ es el índice de refracción relativo. La cantidad $n + ik$ es el índice de refracción complejo de la partícula en λ, R el radio y $n_{entorno}$ es el índice de refracción real del medio en λ.

Para calcular un espectro de absorbancia debe calcularse varias longitudes de onda de luz incidente. Para ello, se necesita el índice de refracción complejo de las partículas y el índice de refracción real del medio en cada una de estas longitudes de onda. En la sección 6.1 están los datos de índice de refracción complejos para la plata a granel. Esto se ajustó con una interpolación tipo spline y se obtuvieron índices de refracción interpolados generando 1000 puntos entre 200 y 900 nm.

7 Simulación Numérica

En la actualidad los pilares de la ciencia se han incrementado a tres, el primero de ellos es la teoría, el segundo la experimentación y el tecero que en las últimas decadas ha tomado gran relevancia en el ámbito científico, es la simulación numérica [13]. Las simulaciones computacionales son ahora una herramienta de gran ayuda, debido a la flexibilidad que tiene de formar un vínculo entre la teoría y la corroboración empírica. Formando un pilar intermedio donde la interpretación es posible, ya que con ellas se puede generar una gran cantidad de datos que se puede convertir en información relevante a trevés del modelado y utilizando lenguajes de programación. Esto mitiga el problema de la imposibilidad de recurrir a la experimentación directa, por factores de recursos o viabilidad. Las simulaciones numéricas constituyen la recreación de un micromundo (en el caso de estudio de la nanotecnología) donde es posible emular escenarios de estudio de

forma segura y eficiente, permitiendo incluso la posibilidad de proponer condiciones que experimentalmente podrían resultar difíciles o riesgosas de alcanzar. En este sentido la metodología de la experimentación se beneficia del uso de recursos computacionales y metodologías específicas como las simulaciones numéricas.

7.1. Espectros de absorbancia de NP Ag

Dado que este trabajo es teórico y no experimental, con ayuda del departamento de División de Ciencias e Ingeniería de la Universidad de Guanajuato Campus León se caracterizaron nanopartículas de plata (Ag) de forma experimental de aproximadamente $20\ nm$ para hacer la comparación teórica utilizando la teoría de Mie y observar que los resutados teóricos se aproximen a los experimentales. El espectrofotómetro de la Figura 6 fue el utilizado para llevar a cabo la caracterización de los espectros de absorbancia.

Figura 6: Espectrofotómetro marca UNICO 4802 UV-Vis de doble haz, con un rango 200-900 nm. Fuente: FCFM, UAdeC

Con este espectrofotómetro se caracterizaron las nanopartículas de plata diluidas en agua. En la Figura (7) se muestra este proceso.

Figura 7: Disolución de nanoparticulas de plata 4 veces en agua. Fuente: Elaboración propia

La gráfica del espectro de absorbancia de la cuarta disolución ($39{,}0625\ ppm$) se muestra en la Figura 8, para nanopartículas de aproximadamente 20nm.

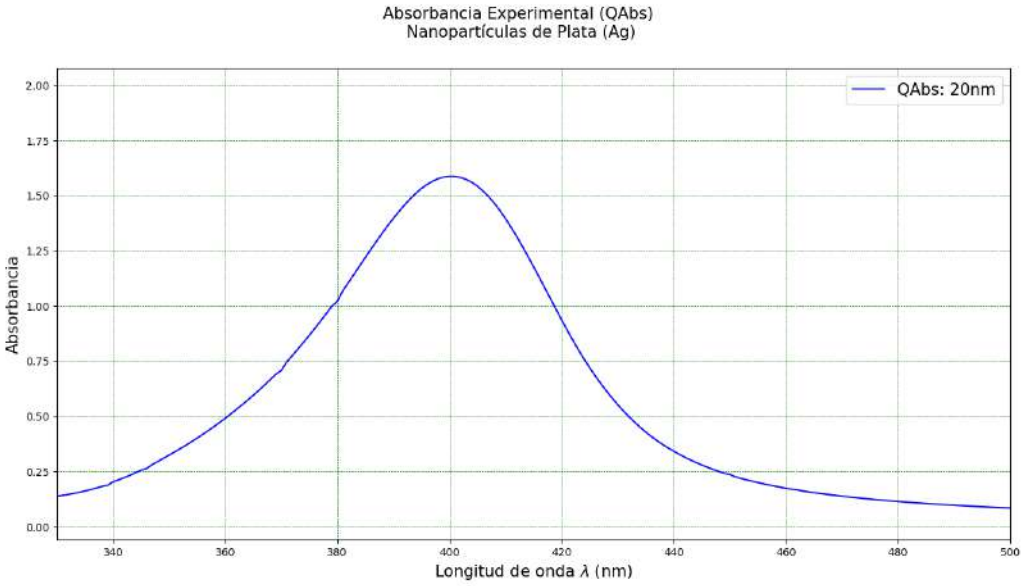


Figura 8: Espectros de absorbancia de nanopartículas de Ag de aproximadamente 20 nm. Fuente: Elaboración propia.

La longitud de onda (λnm) seleccionada es de $300nm$ a $550nm$ donde se puede observar la respuesta óptica (absorbancia) de las nanopartículas de plata (Ag), ver Figura 8. El pico del plasmón superficial localizado se encuentra en $400nm$, tal como se reporta en la literatura para rangos de tamaños de nanopartículas plata de entre $10nm$ y $20nm$ [11].

8 Teoría de Mie en Python

Para el cómputo de las eficiencias obtenidas por la teoría de Mie. Es necesario implementar algoritmos desarrollados para obtener la caracterización del efecto del plasmón superficial por medio de esta teoría. En la historia de los algoritmos desarrollados para este fin, existe una gran diversidad de ellos [26].

Los cálculos de la dispersión de la luz de las partículas son necesarios en la más amplia variedad de actividades de investigación. El cálculo prototípico de este tipo supone que la partícula es una esfera homogénea y la luz incidente una onda plana monocromática. En 1908 Mie publicó soluciones completamente generales. Por un extraño giro del destino, el nombre de Mie se asoció exclusivamente con este problema. La historia computacional de la dispersión de Mie es bastante diferente y mucho más reciente que la historia teórica [29].

8.1. Descripción de las funciones a utilizar

La librería MiePython se utilizó para el desarrollo de las simulaciones numéricas de esta investigación. Con el propósito de generar escenarios y estudiar las propiedades ópticas de las nanopartículas de plata, se llevó a cabo la escritura de guiones (scripts) de prueba para ir conociendo esta librería y algunas modificaciones específicas para adatparla al problema en partícular.

Tabla 1: Módulos para el cálculo de las eficiencias de Mie. Fuente: Elaboración propia.

Nombre de funciones	Descripción
ez_mie(m, d, lambda0[, n_env])	Calcula las eficiencias de Mie de una esfera.
mie(m, x)	Calcula las eficiencias de Mie para una esfera donde m o x pueden ser matrices.
mie_S1_S2(m, x, mu[, norma])	Calcula las funciones de amplitud de dispersión para esferas.

Los cálculos de dispersión de Mie para esferas, se escriben en código fuente Python en módulos (o funciones). La eficiencia de extinción, la eficiencia de dispersión, la retrodispersión y la asimetría de dispersión para una esfera con índice de refracción m complejo, diámetro d y longitud de onda (λ) se pueden encontrar mediante:

```
qext, qsca = miepython.ez_mie(m, d, lambda0)
```

donde los argumentos de entrada son calculados previamente, como se mencionó en la sección 6 y la salida son los coeficientes de extinción y dispersión respectivamente. Sin embargo, dado que parte de la luz incidente puede ser absorbida (cuando m_{img} no es cero), también hay un área de la onda incidente que es absorbida $qabs$. Para calcular el coeficiente de absorbancia se utiliza la expresión $qext = qsca + qabs$, despejando $qabs$ (coeficiente de absorbancia), es decir: $qabs = qext - qsca$.

Las secciones transversales de dispersión y absorción σ tienen unidades de área y se pueden obtener a partir de las eficiencias $qabs = qext - qsca$ multiplicando por la sección transversal geométrica πr^2 de la esfera.

$$\sigma_{sca} = \pi r^2 qsca \tag{14}$$
$$\sigma_{ext} = \pi r^2 qext \tag{15}$$
$$\sigma_{abs} = \pi r^2 qabs \tag{16}$$
$$\tag{17}$$

Por ejemplo, la sección transversal de dispersión σ_{sca} es el área efectiva de una onda plana incidente que interactúa y produce luz dispersa.

En este trabajo se hizo una modificación al algoritmo estándar presentado en [28] de las eficiencias de Mie. La expresión utilizada en el algritmo actual (modificado) es la presentada en la ecuación (10) de la sección 6.2 para calcular la absorbancia teórica de nanopartículas de Ag.

8.2. APP Teoría de Mie

Para cumplir con el objetivo específico de calcular las respuestas ópticas de nanopartículas de plata es necesario crear una aplicación que utilice la simulación numérica con la librería llamada MiePython, para generar distintos escenarios y obtener los espectros de absorbancia de manera teórica.

En este sentido, se utiliza un marco de trabajo para el desarrollo de interfaces gráficas llamado Kivy y que es otro de los proyectos desarrollados en Python, para solventar la problemática de la compatibilidad entre los sistemas operativos más utilizados en el mundo (Unix BSD,GNU/Linux, Mac IOs, MS Windows, Android, etc). A la aplicación desarrollada se le dió el nombre de *AppMie*.

La aplicación se divide en dos etapas principales. La etapa de procesamiento, que es propiamente la implementación de los algoritmos [28] y la etapa de diseño o visualización de la aplicación. La relevancia de escribir este tipo de aplicaciones utilizando Kivy, es precisamente la separación del desarrollo en la parte lógica (simulación numérica) y la visual.

El procesamiento implica, la lectura de los datos, el cálculo de las eficiencias de Mie y la definición de los objetos para la interacción entre ellos (flujo de datos entre métodos e instancias de los objetos). La graficación es el resultado final de todo el cómputo numérico obtenido con las eficiencias de Mie y donde se visualiza la respuesta óptica de los nanopartículas de plata, así como la presentación gráfica de los índices de refracción complejos de los metales en bulto y que son discretos, como los índices de refracción interpolados.

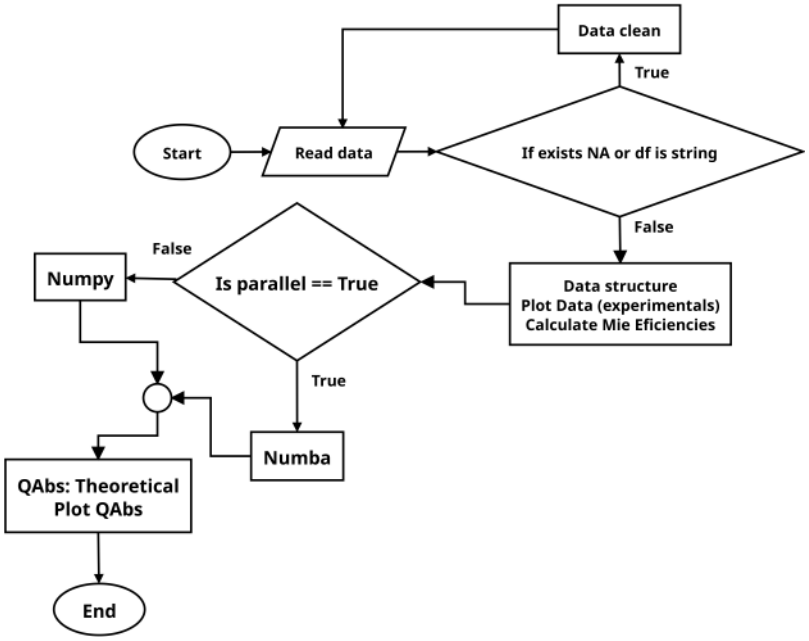


Figura 9: Etapa de procesamiento de Eficiencias de Mie. Fuente: Elaboración propia

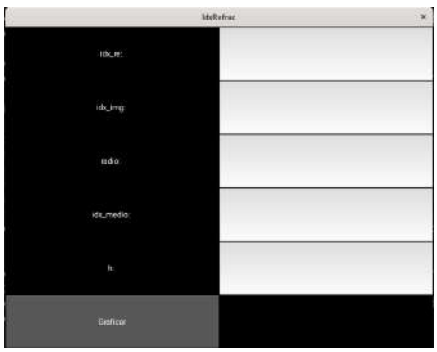
La Figura 9 es un diagrama de flujo, donde se observan los distintos procesos que realiza el conjunto de scripts escritos en Python. La primer condición if $\exists$ NA || df is string se utiliza después de dar lectura a los

archivos a utilizar y es la encargada de detectar la existencia de valores perdidos o no capturados (NA: Not a Number) o de encontrar si en el contenido del archivo leído existen caracteres o cadena de caracteres que no son números. Este es el procesamiento de limpieza de los datos.

Las siguientes etapas son las responsables de hacer la estructuración de los datos, así como su graficación (datos experimentales obtenidos) y que no demanda de procesamiento intenso, solo lectura de datos.

La condición $if\ \ paralelo == True$ es para seleccionar en la AppMie si se utilizará el clúster de computadoras para procesamiento en paralelo o se hará procesamiento secuencial utilizando solo una de ella. Esta etapa está en fase de prueba y se encuentra funcional, sin embargo, se debe seguir precisando detalles en la programación que por el momento no son relevantes para el estudio de las propiedades ópticas de las nanopartículas de plata.

En la Figura 10 se muestra un conjunto de capturas de pantalla donde se observan las primeras ventanas iniciales programadas para la AppMie. Esta es la etapa de diseño, donde se utiliza programación orientada a objetos para abstraer el conjunto de objetos a utilizar en código fuente utilizando Kivy y Python.



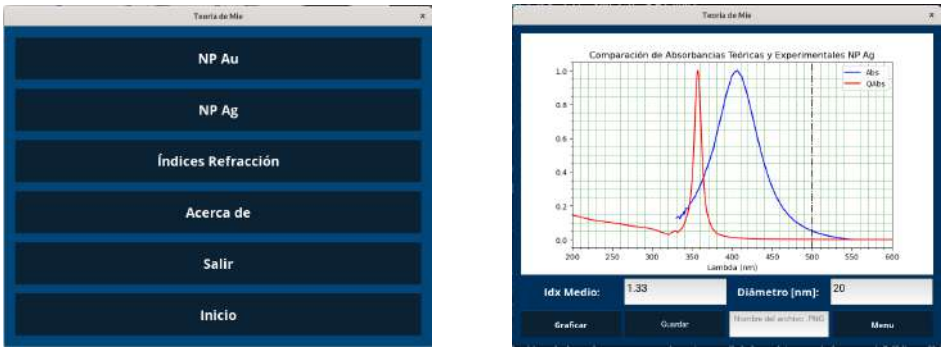
(a) Obtener datos para eficiencias de Mie versión 0.2

(b) Menú principal versión 0.1

(c) Menú de procesamiento de la aplicación.

Figura 10: Diseño inicial de AppMie. Fuente: Elaboracion propia

La etapa actual del diseño de la interfaz gráfica se muestra en la Figura 11.



(a) Menú de opciones.

(b) Ejemplo de obtención de absorbancias de NP Ag (experimental y teórica).

Figura 11: Desarrollo funcional de AppMie, para generar escenarios de la respuesta óptica de nanopartículas de plata. Fuente: Elaboración propia.

La aplicación es completamente funcional y es la utilizada para generar los distintos escenarios para estudiar el efecto de la resonancia del plasmón superficial localizado tanto para nanopartículas de plata (Ag) como de nanopartículas de oro (Au) con diferentes tamaños e índices de refracción del medio. En la Figura 11 (b) se observan dos curvas, una de color azul que se refiere a la gráfica de las absorbancias experimentales obtenidas y la curva roja que se refiere a la gráfica de la absorbancia teórica calculada, donde se puede visualizar la respuesta óptica aproximada a la respuesta experimental que en este caso es la resonacia del plasmón superficial localizado de la plata (Ag).

9 Conclusiones

El estudio de las nanopartículas metálicas es un campo fértil y desafiante tanto en lo teórico como en lo experimental. Propiedades como el tamaño y la geometría a dimensiones nanométricas implican el estudio de fenómenos como el de la RPLS a través de métodos computacionales y teorías que expliquen su comportamiento.

Los resultados parciales de este trabajo sobre las nanoesctructuras metálicas de plata (Ag) utilizando la teoría de Mie son favorables. Las comparaciones de los cálculos teóricos obtenidos con respecto a lo reportado en trabajos experimentales son similares. Sin embargo, es necesario construir un modelo para obtener la absorbancia de una distribución de nanopartículas de diferentes tamaños. Existe en la literatura investigaciones en este sentido, pero no es clara la manera en que lo realizan, omiten detalles teóricos que no son simples de reproducir. En este sentido, se está trabajando actualmente para obtener un modelo que obtenga dicha absorbancia.

Uno de los elementos principales para obtener las eficiencias de Mie son los índices de refracción complejos. Estos son utilizados como datos de entrada en algoritmos de Mie y actualmente en este trabajo fueron utilizados los reportados por [18]. Sin embargo, es importante utilizar otro conjunto de datos y seguir estudiando las propiedades ópticas este tipo de nanopartículas.

El desarrollo computacional es una herramienta importante. Los cálculos de las eficiencias de Mie demandan en algún momento un alto costo computacional en el cálulo de los armónicos esféricos en las eficiencias de Mie para sistemas coloidales de nanopartículas, donde se desconoce la cantidad y tamaño de ellas y esto implica tener algoritmos implementados de manera secuencial como paralelos. Actualmente se tiene un avance en este sentido, se cuenta con un desarrollo preliminar que utiliza de forma vectorial y paralela, los algoritmos reportados en la literatura y que son una base para el cálculo de las eficiencias de Mie. La pieza clave en este trabajo es la librería Numba, diseñada para la implementación de este tipo de algoritmos y que en la aplicación desarrollada en este trabajo está en su fase preliminar.

Referencias

[1] D. Acharya, B. Mohanta, S. Deb, and A. K. Sen. Theoretical prediction of absorbance spectra considering the particle size distribution using mie theory and their comparison with the experimental uv–vis spectra of synthesized nanoparticles. *Spectroscopy Letters*, 51(3):139–143, 2018.

[2] M. Aguilar. *Síntesis verde de nanopartículas de Ag, Au, Cu2O y preparación convencional de nanoestructuras de Cu, Cu2O y CuO en diferentes morfologías para la evaluación de sus propiedades catalíticas y ópticas*. PhD thesis, Tesis de Doctorado. Instituto de Investigación en Metalurgía y Materiales, 2019.

[3] R. Ahmad, M. Khan, M. R. Khan, N. Tripathy, M. I. R. Khan, P. Mishra, M. A. Syed, and A. Khosla. Nano-donuts shaped nickel oxide nanostructures for sensitive non-enzymatic electrochemical detection of glucose. *Microsystem Technologies*, pages 1–6, 2020.

[4] S. Bayda, M. Adeel, T. Tuccinardi, M. Cordani, and F. Rizzolio. The history of nanoscience and nanotechnology: from chemical–physical applications to nanomedicine. *Molecules*, 25(1):112, 2019.

[5] Á. S. C. Beltrán, M. Z. Herrera, et al. Plasmónica en el régimen subnanométrico. *Revista Digital Universitaria*, 2015.

[6] M. Boholm and R. Arvidsson. A definition framework for the terms nanomaterial and nanoparticle. *NanoEthics*, 10:25–40, 2016.

[7] P. C. Cardoso. Nanopartículas de plata: obtención, utilización como antimicrobiano e impacto en el área de la salud. *Rev. Hosp. Niños (B. Aires)*, 58(260):19–28, 2016.

[8] F. M. S. Carreño. Análisis de campos débiles en la predicción de ondas gravitacionales. *Revista Bases de la Ciencia*, 7(2):33–47, 2022.

[9] J.-H. Choi, H. Wang, S. Oh, T. Paik, P. Sung, J. Sung, X. Ye, T. Zhao, B. Diroll, C. Murray, and C. Kagan. Exploiting the colloidal nanocrystal library to construct electronic devices. *Science*, 352:205–208, 04 2016.

[10] V. Coello. Nanofotónica. los grandes avances y retos de un mundo pequeño. *Mundo nano. Revista interdisciplinaria en nanociencias y nanotecnología*, 13(24), 2020.

[11] V. E. Escobar Falconí. Síntesis y caracterización de nanopartículas de plata por espectroscopia de infrarrojos (ft-ir), uv-vis, absorción atómica de llama (faas) y microscopía de barrido electrónico (sem). B.S. thesis, PUCE, 2015.

[12] N. Feng, Y. Zhang, X. Tian, J. Zhu, W. T. Joines, and G. P. Wang. System-combined adi-fdtd method and its electromagnetic applications in microwave circuits and antennas. *IEEE Transactions on Microwave Theory and Techniques*, 67(8):3260–3270, 2019.

[13] O. Garate, L. Veiga, P. Lloret, and G. Ybarra. Introducción a los fenómenos ópticos de nanopartículas metálicas a través simulaciones computacionales en línea. *Educación química*, 30(1):31–41, 2019.

[14] Z. Geng, B. Xiong, L. Wang, K. Wang, M. Ren, L. Zhang, J. Zhu, and Z. Yang. Moebius strips of chiral block copolymers. *Nature Communications*, 10(1):4090, 2019.

[15] M. Gustav. Beitrage zur optik truber medien, speziell kolloidaler metallosungen. *Ann. Phys. Band*, 25:377–445, 1908.

[16] K. Haume, S. Rosa, S. Grellet, M. A. Śmiałek, K. T. Butterworth, A. V. Solov'yov, K. M. Prise, J. Golding, and N. J. Mason. Gold nanoparticles for cancer radiotherapy: a review. *Cancer nanotechnology*, 7:1–20, 2016.

[17] H. Horvath. Gustav mie and the scattering and absorption of light by particles: Historic developments and basics. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 110(11):787–799, 2009.

[18] P. B. Johnson and R.-W. Christy. Optical constants of the noble metals. *Physical review B*, 6(12):4370, 1972.

[19] Y. H. Ko and R. Magnusson. Wideband dielectric metamaterial reflectors: Mie scattering or leaky bloch mode resonance? *Optica*, 5(3):289–294, 2018.

[20] A. Loiseau, V. Asila, G. Boitel-Aullen, M. Lam, M. Salmain, and S. Boujday. Silver-based plasmonic nanoparticles for and their use in biosensing. *Biosensors*, 9(2):78, 2019.

[21] W. L. Mochan. Plasmons. *Reference Module in Materials Science and Materials Engineering; Elsevier: Amsterdam, The Netherlands*, 2016.

[22] M. F. Ramírez Ayala. *Síntesis de nanopartículas de Au, Ag y AuAg con forma y tamaño controlado, mediante el uso de polielectrolitos ácidos.* PhD thesis, Universidad Autónoma del Estado de Hidalgo, 2020.

[23] J. M. J. Santillán. *Estudio de las propiedades ópticas de materiales nanoestructurados y aplicaciones*. PhD thesis, Universidad Nacional de La Plata, 2013.

[24] T. V. Shahbazyan and M. I. Stockman. *Plasmonics: theory and applications*, volume 15. Springer, 2013.

[25] M. I. Stockman. Nanoplasmonics: The physics behind the applications. *Physics Today*, 64(2):39–44, 2011.

[26] B. Stout and N. Bonod. Gustav mie: the man, the theory. *Photoniques*, I(101):22–26, 2020.

[27] J. N. Tiwari, R. N. Tiwari, and K. S. Kim. Zero-dimensional, one-dimensional, two-dimensional and three-dimensional nanostructured materials for advanced electrochemical energy devices. *Progress in Materials Science*, 57(4):724–803, 2012.

[28] W. J. Wiscombe. Mie scattering calculations: Advances in technique and fast, vector-speed computer codes. In *NCAR Technical Note*, 1979.

[29] T. Wriedt. Mie theory: a review. *The Mie theory: Basics and applications*, pages 53–71, 2012.

A Mathematical Model to Estimate the COVID-19 Pandemic in Red Lights of México

Alejandro Peregrino*[1], Domingo González[1] y Jorge López[1]

[1]División Académica de Ciencias Básicas, Universidad Juárez Autónoma de Tabasco

Abstract

The goal of this work is a dynamic and numerical study of a compartmental mathematical model in order to know the evolution of state variables of the pandemic caused by COVID-19 on red spotlights in México. The model include 7 compartments and the results were compared with the classic SIR model, resulting that our predictions adjusted better the official data. A parameter sensitivity analysis is included.

keywords: COVID-19 pandemic; local stability; Equilibrium point; reproductive basic number.

1 Introduction

On December 31 2019, Wuhan Municipal Health Commission in Hubei Province, China reported a cluster of 27 pneumonia cases with unknown etiology. The outbreak is associated with common exposures in one wholesale seafood and living animals market in Wuhan City including seven serious cases [14]. The starting point for the first symptomatic case was notified on December 2019. By the 7th of January 2020, the Chinese authorities identified as the causative agent of this outbreak a new type of coronavirus from the family coronaviridae that was called later SARS-CoV-2 which genetic sequence was shared by the Chinese authorities on January the 12th [2]. The via of transmission between human beings, it is similar to the one described by other coronaviruses, by secretions of infected people, mainly by having direct contact with breathing droplets up to 5 microns (allowing them to be transmitted over distances closer than 2 meters) and the contaminated hands or fomites with such secretions, followed by the direct contact of mucosa emissions from the mouth, nose or eyes [9]. The virus SARS-CoV-2 has been founded into the nasopharynx excretions too, including the saliva [15]. On March 11 2020 OMS declared COVID-19 as a world pandemic, since then it is considered the main health problem not only by the mortality rate but also by the secondary

*alejandro.peregrino@ujat.mx

effects presented on people such as: Psychological illness, economic loss and the negative impact on their daily activities [18]. From March 20th to May 31st The Secretariat of Health Mexico implemented the social distancing, best known as Healthy Distance (quarantine) because of the COVID-19 pandemic, then to avoid a catastrophic impact on the economy's country, on June 1st, a "The New Normality" started and the goverment reported weekly health alert per region symilar to the traffic light sytem, so with colors there was indicated the type of authorized activities in all economic, labor, educational and social spheres. In this context, the mathematical models of the dynamic transmission on infected diseases have taken an unprecedented relevance. They were applied to help the government's actions against COVID-19. These models play an important role so as to quantify some possible strategies for mitigation and control of infected diseases [6, 13]. There are several models for the infected diseases, since the Clasical SIR Model to more complex proposals [4]. The applied mathematical models, such as Gompertz and Logistic [8, 17], have been used with some success to predict the number of infected people with COVID-19. Unfortunately these ones predict on right acurately in short time periods and a huge quantity of data is requested for a good sickness estimation, as it is shown on a recent work done by Torrealba Rodriguez et al. [16], for a COVID-19 estimation in México. On his contributions, Denaiuru et al. [11], makes the proposal for a compartmental mathematical model to the spread of COVID-19 disease, by giving special emphasis on the super-spreaders individuals transmissibility. The problem by working with extensive compartmental models, it is the lack of information for most of the parameters. In a recent work, Neves Armando G.M et. al [12] propose a slightly generalized version of the A-SIR model and propose a scheme for fitting the parameter for the model to real data using the time series only of the deceased individuals. The scheme is applied to the concrete cases of Lombardy, Italy and São Paulo state, Brazil, showing different aspects of the epidemic. In both cases they see strong evidence that the adoption of social distancing measures contributed to a lower increase in the number of deceased individuals when compared to the baseline of no reduction in the infection rate.

On this work, we consider a adaptation of the model in [11] by discarding the super-infectious class but considering the asymtomatic class as an infectious vector. Also we have adjusted the parameters of the resulting model to the Mexican country situations. So we present a consistent compartmental model involving to susceptible class S, exposed class E, infectious with symptoms class I, asymptomatic infectious class A, Hospitalized class H, recovery class R, and death class F. Then a qualitative analysis of the model is done. We show the basic reproduction number and the local stability of the free disease equilibrium. The model is solved numerically to predict the dynamic's pandemic in two federal entities which are considered "red spotlights" for the COVID-19 such as: the CDMX and Tabasco state where both have implemented the National Health Distance and "New Normality". Finally we present the estimations with the Classic SIR Model in order to be compared with the proposed model predictions.

2 Model formulation

We propose a compartment model that takes into account the hospitalized individuals and died individuals due to the virus infection. Considering that the size of the population is constant, N,

we subdivide this population into seven classes: Suceptibles class S, Exposed class E, infectious and symptomatic class I, infectious and asymptomatic class A, sick hospitalized class H, recovery class R and fatality class F, We consider that the rate at which susceptible individuals are infected by having contact with the symptomatic class is β_s and the rate at which succeptible individuals are infected by having contact with the asymptomatic class is β_a. On the other hand, the incubation period of the virus is considered to be $\frac{1}{k}$ and that a fraction σ of exposed individuals do not develop symptoms, while a fraction ϕ_h of individuals who developed symptoms, require hospitalization. The average recovery period of symptomatic individuals, individuals asymptomatic and hospitalized individuals is represented by $\frac{1}{\gamma_s}$, $\frac{1}{\mu}$ and $\frac{1}{\gamma_h}$, respectively. Furthermore, we assume that symptomatic individuals and hospitalized individuals die at a rate δ_s and δ_h respectively. These relationships between the different classes are represented in the diagram of the Figure 1.

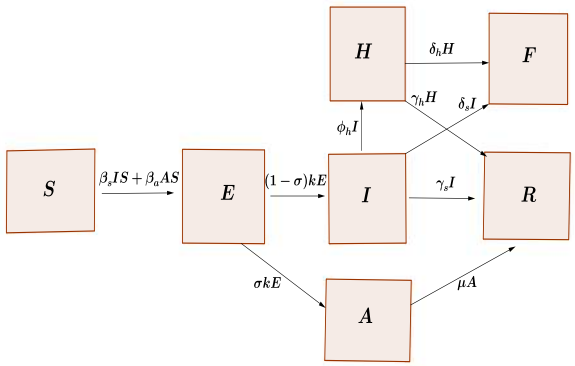


Figure 1: Transference diagram for the proposed COVID-19 model.

According to the diagram of in Figure 1, the dynamics of the epidemic can be represented using the following model of ordinary differential equations

$$\begin{aligned}
\frac{dS}{dt} &= -\frac{\beta_s IS}{N} - \frac{\beta_a AS}{N} - \frac{\beta_h HS}{N} \\
\frac{dE}{dt} &= \frac{\beta_s IS}{N} + \frac{\beta_a AS}{N} + \frac{\beta_h HS}{N} - kE \\
\frac{dI}{dt} &= k(1-\sigma)E - (\phi_h + \gamma_s + \delta_s)I \\
\frac{dA}{dt} &= k\sigma E - \mu A \\
\frac{dH}{dt} &= \phi_h I - (\gamma_h + \delta_h)H \\
\frac{dR}{dt} &= \gamma_s I + \mu A + \gamma_h H \\
\frac{dF}{dt} &= \delta_s I + \delta_h H.
\end{aligned} \tag{1}$$

The region of interest for the analysis of the model is

$$\Gamma = \{(S, E, I, A, H, R, F) \in \mathbb{R}^7 : S \geq 0,\ E \geq 0,\ I \geq 0,\ A \geq 0,\ H \geq 0,\ R \geq 0,\ F \geq 0\}.$$

3 Stability analysis of disease-free equilibrium.

The model (1) has the disease-free equilibrium point $E_0 = (N, 0, 0, 0, 0, 0, 0)$. The next generation operator method of Van den Driessche is used to calculate R_0 [5]. According to the method, we consider the group of disease variables to $x = (E, I, A, H)$ and the disease-free variable group to $y = (S, R, F)$. Thus,

$$x' = \Psi(x, y) - \Theta(x, y),$$

where

$$\Psi = \begin{pmatrix} \frac{\beta_s IS}{N} + \frac{\beta_a AS}{N} + \frac{\beta_h HS}{N} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

and

$$\Theta = \begin{pmatrix} kE \\ -k(1-\sigma)E + (\phi_h + \gamma_s + \delta_s)I \\ -k\sigma E + \mu A \\ -\phi_h I + (\gamma_h + \delta_h)H \end{pmatrix}$$

represent new infections and transitions of individuals between groups, respectively. Therefore

$$F = \left[\frac{\partial \Psi_i}{\partial x_j}(E_0)\right] = \begin{bmatrix} 0 & \beta_s & \beta_a & \beta_h \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$V = \left[\frac{\partial \Theta_i}{\partial x_j}(E_0)\right] = \begin{bmatrix} k & 0 & 0 & 0 \\ -k(1-\sigma) & h_1 & 0 & 0 \\ -k\sigma & 0 & \mu & 0 \\ 0 & -\phi_h & 0 & h_2 \end{bmatrix}$$

and

$$V^{-1} = \begin{bmatrix} \frac{1}{k} & 0 & 0 & 0 \\ \frac{1-\sigma}{h_1} & \frac{1}{h_1} & 0 & 0 \\ \frac{\sigma}{\mu} & 0 & \frac{1}{\mu} & 0 \\ \frac{\phi_h(1-\sigma)}{h_1 h_2} & \frac{\phi_h}{h_1 h_2} & 0 & \frac{1}{h_2} \end{bmatrix}$$

where $h_1 = \phi_h + \gamma_s + \delta_s$ and $h_2 = \gamma_h + \delta_h$.

The basic reproduction number, which measures the secondary cases of COVID-19 caused by an infected individual in a susceptible population during your infectious period, is given by $R_0 = \rho(FV^{-1})$, where ρ denotes the spectral radius. That is

$$R_0 = \frac{\beta_s(1-\sigma)}{h_1} + \frac{\beta_a \sigma}{\mu} + \frac{\beta_h \phi_h (1-\sigma)}{h_1 h_2}.$$

Note that R_0 is the sum of three reproductive numbers :

$$R_I = \frac{\beta_s(1-\sigma)}{h_1}, \quad R_A = \frac{\beta_a \sigma}{\mu}, \quad R_H = \frac{\beta_h \phi_h (1-\sigma)}{h_1 h_2}.$$

R_I is given by the product of: the infection rate of susceptible individuals by interaction with symptomatic individuals β_s, the proportion of individuals who develop symptoms $(1-\sigma)$ and the average duration of an individual in the infectious class $\frac{1}{h_1}$. R_A is given by the product of: the infection rate of susceptible individuals by interaction with asymptomatic individuals β_a, the proportion of individuals who do not develop symptoms after the incubation, σ, and the average duration of an individual in the asymptomatic class $\frac{1}{\mu}$. Similarly, R_H is given by the product of: the infection rate of susceptible individuals by interaction with hospitalized individuals β_h, the proportion of individuals who develop symptoms and require hospitalization, $\phi_h(1-\sigma)$, and the average duration of an individual to travel from the first contact with the virus to leave the class of hospitalized $\frac{1}{h_1 h_2}$.

Theorem 1 *If $R_0 < 1$ the disease-free equilibrium E_0 is locally asymptotically stable.*

Proof 1 *Noting that the two last equations of system (1) are uncoupled to the remaining equations of the system, we can easily obtain, by direct integration, the following analytical results:*

$$\begin{aligned} R(t) &= \gamma_s \int_0^t I(s)ds + \mu \int_0^t A(s)ds + \gamma_h \int_0^t H(s)ds, \\ F(t) &= \delta_s \int_0^t I(s)ds + \delta_h \int_0^t H(s)ds. \end{aligned} \tag{2}$$

In the same way as the total population N is constant, then

$S(t) = N - [E(t) - I(t) - A(t) - H(t) - R(t) - F(t)]$.

Thus, the local stability of model (1) can be studied through the remaining coupled system of state variables, namely, the variables E, I, P , and H. The Jacobian matrix of (1) restricted to these variables and evaluated at the equilibrium point $E_0 = (N, 0, 0, 0, 0, 0, 0)$ is given by

$$J(E_0) = \begin{bmatrix} -k & \beta_s & \beta_a & \beta_h \\ k(1-\sigma) & -h_1 & 0 & 0 \\ k\sigma & 0 & -\mu & 0 \\ 0 & \phi_h & 0 & -h_2 \end{bmatrix}.$$

The characteristic polynomial of $J(E_0)$ is

$$P(\lambda) = \lambda^4 + a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4, \tag{3}$$

where

$$\begin{aligned} a_1 &= k + h_1 + h_2 + \mu, \\ a_2 &= (\delta_h + \mu + \gamma_h)h_1 + (k + \mu)h_2 + (\gamma_s + \delta_s + \mu)k \\ &\quad - (\beta_s(1-\sigma) + \beta_a\sigma - \phi_h)k, \\ a_3 &= (\gamma_h - \beta_a\sigma + \mu + \delta_h)kh_1 + (\mu - \beta_s(1-\sigma) - \beta_a\sigma)kh_2 \\ &\quad + \mu h_1 h_2 + (\beta_s\mu(1-\sigma) + \beta_h\phi_h(1-\sigma))k, \\ a_4 &= (\gamma_s\mu + \delta_s\mu - \beta_s\mu(1-\sigma) - \beta_a\gamma_s\sigma - \beta_a\delta_s\sigma + \mu\phi_h \\ &\quad - \beta_a\sigma\phi_h)kh_2 - k\beta_h\mu(1-\sigma)\phi_h. \end{aligned} \tag{4}$$

Using the Liénard-Chipard test [10], all the roots of $P(\lambda)$ (the eigenvalues of $J(E_0)$) are negative or have negative real part if, and only if, the following conditions are satisfied:

- $a_i > 0$, $i = 1, 2, 3, 4$.
- $a_1 a_2 > a_3$.

In order to check these conditions of the Liénard-Chipard test, we rewrite the coefficients a_1, a_2, a_3,

and a_4 of the characteristic polynomial in terms of the basic reproduction number R_0:

$$\begin{aligned}
a_1 &= k + h_1 + h_2 + \mu, \\
a_2 &= k(1-R_0)(h_1+h_2+\mu) + (\mu + k\beta_a\sigma)(h_1+h_2) + h_1h_2 \\
&\quad + k\beta_h\phi_a\left(\frac{1}{h_1} + \frac{1}{h_2} + \frac{\mu(1-\sigma)}{h_1h_2}\right) + \frac{k\beta_s(1-\sigma)(\mu+1)}{h_1}, \\
a_3 &= k(1-R_0)(h_1h_2 + h_1\mu + h_2\mu) + h_1h_2\mu + \frac{kh_1h_2\beta_a\sigma}{\mu} \\
&\quad + k\mu\beta_h\phi_h(1-\sigma)\left(\frac{1}{h_1} + \frac{1}{h_2}\right) + \frac{kh_2\mu\beta_s(1-\sigma)}{h_1}, \\
a_4 &= k(1-R_0)h_1h_2\mu.
\end{aligned} \tag{5}$$

In this way, the first Liénard-Chipard's condition is evident. On the other hand,

$$\begin{aligned}
a_1a_2 - a_3 = {} & (1-R_0)\,\Omega_1 + (1-\sigma)\Omega_2 \\
& + h_2\left(\mu\left(\mu+h_2\right) + k\sigma\left(1+k+\mu+h_2\right)\beta_a\right) \\
& + h_1^2\left(\mu + h_2 + k\sigma\beta_a\right) + \frac{k\phi\left(k+\mu+h_2\right)\beta_h}{h_1} \\
& + \frac{\Omega_3}{h_2},
\end{aligned}$$

where

$\Omega_1 = k\left(h_1^2 + \left(\mu+h_2\right)\left(k+\mu+h_2\right) + h_1\left(k+\mu+2h_2\right)\right)$.
$\Omega_2 = k\phi\beta_h + kh_2\beta_s + \frac{k^2\mu\phi\beta_h + k\mu^2\phi\beta_h + k\mu\phi h_2\beta_h}{h_1h_2}$.
$\Omega_3 = k\phi\left(k+\mu+2h_2\right)\beta_h$
$+ h_1\left(h_2^3 + (k+\mu)h_2\left(\mu+k\sigma\beta_a\right) + 2h_2^2\left(\mu+k\sigma\beta_a\right) + k\phi\beta_h\right)$.

This shows that Liénard-Chipard's second condition is fulfilled. Therefore, the proof is complete.

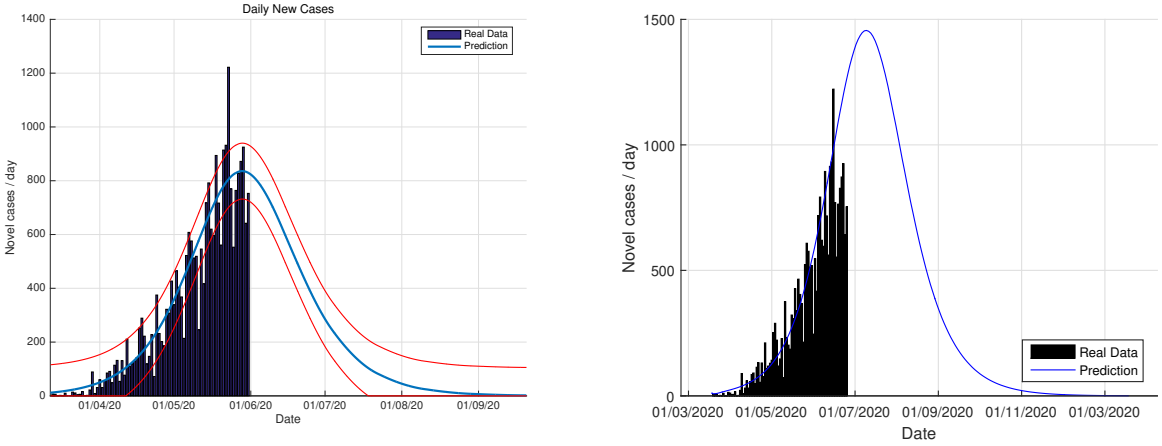
4 Numerical results (National Healthy Distance Period)

In this section, we show numerical results for the model (1), using Matlab and Mathematica. To solve SIR model we used the algorithms shown in [3]. The simulations were runed with the official data of the cities of México, CDMX and Tabasco, that was reported by the Healthy Secretariat during the National Healthy Distance Period, from March 20th to May 31st, 2020. The Table 1 shows the parameters used in the numerical simulations. Many of them were obtained from official data from the Mexican government. The infection rate of susceptibles by contact with symptomatic individuals, β_s, was estimated using least squares in the SIR model with reported data.

Table 1: Parameters for model (1).

Parameter	CDMX	Tabasco	Reference
β_s	0.42	0.302	Estimed
β_a	0.39	0.42	Fitting
β_h	0.08	0.08	Fitting
k	$\frac{1}{5}$	$\frac{1}{5}$	[1]
σ	0.68	0.68	[1]
ϕ_h	0.153	0.153	[7]
γ_s	$\frac{1}{7}$	$\frac{1}{7}$	[1]
γ_h	$\frac{1}{18}$	$\frac{1}{18}$	[1]
μ	$\frac{1}{5}$	$\frac{1}{5}$	[1]
δ_s	0.09	0.09	[1]
δ_h	0.035	0.035	[1]

The estimated basic reproductive number R_0 during the National Period of Healthy Distance, for the epidemic in the CDMX using the SIR model is $R_0 = 1.21$, while for the model (1) with the parameters of the table 1 is $R_0 = 1.79$. On the other hand, the estimated R_0 for the epidemic in Tabasco, that was obtained with the SIR model is $R_0 = 1.51$ and the estimation using the model (1) is $R_0 = 1.79$. This means that the model (1) predicts that the epidemic will be stronger with respect to the prediction of the SIR model.



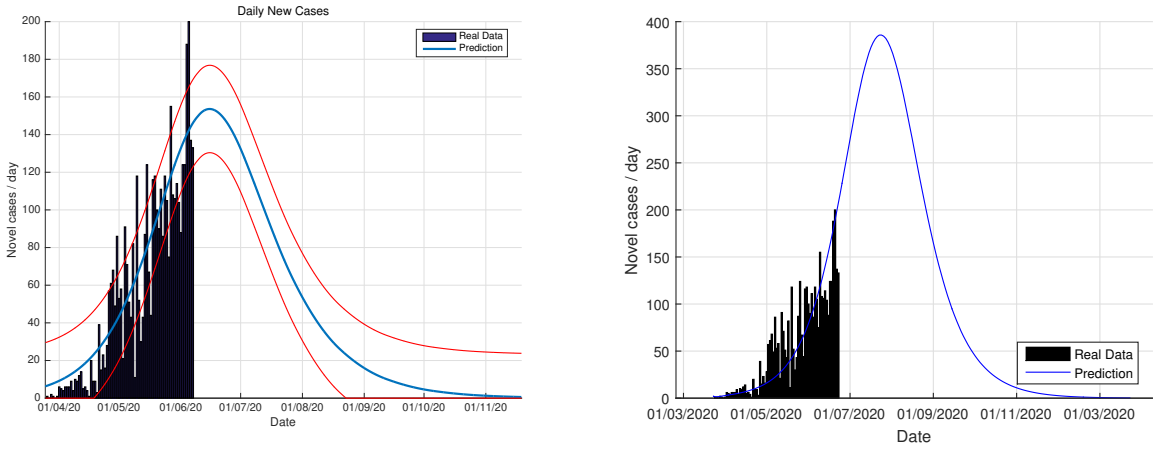
(a) Estimated daily cases for the CDMX, using the SIR model.

(b) Estimated daily cases for CDMX, using the model (1).

Figure 2: Daily infectious cases for CDMX.

Figures 2a and 2b show the estimations for infections per day using the SIR model and the model

(1) for the CDMX. It can be seen that both estimations adjust well the known data. However, with the model (1) the maximum daily infection rate is around 1450 infected, which is reached in mid-June, This is consistent with the reported data. This rate is aproximated better using our nodel instead the SIR model.



(a) Estimated daily new cases for Tabasco using the SIR model.

(b) Estimated daily cases for Tabasco, using the model (1).

Figure 3: Daily infectious cases for Tabasco.

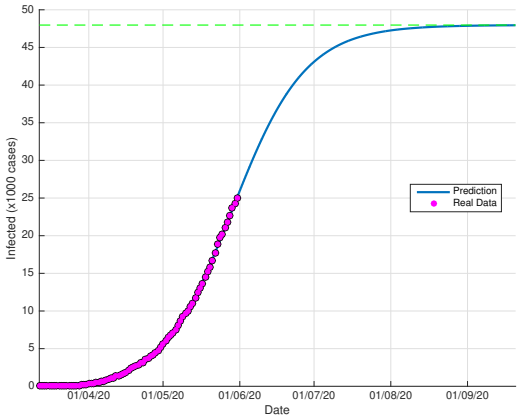
Figures 3a and 3b show the estimates of infections per day using the SIR model and model (1) for Tabasco. Like in the CDMX, the model (1) predicts a maximum of daily cases around 380, that is reached by the end of June 2020, when the confinement was implemented.

Figures 4a and 4b show the cumulative case estimates for the CDMX, using the SIR model and the model (1), respectively. We can note that, both models adjust appropriately the official data. However, they differ in the total number of people who were infected during the pandemic. The SIR model predicted around 48,000 total cases and the model (1) predicted 105,000 total cases. On the other hand, the Healthy Secretariat reported a total of 99564 infections at September 1st, 2020.

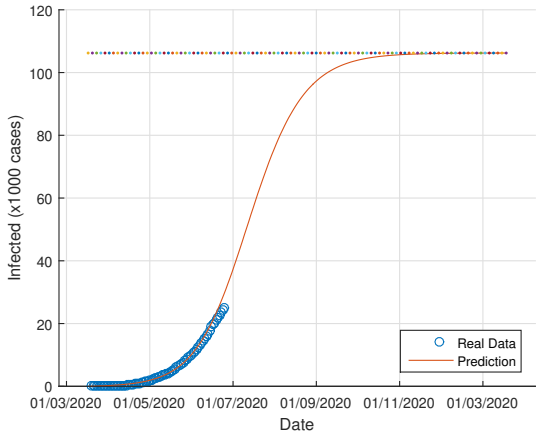
Figures 5a and 5b show the estimated cumulative case for Tabasco using the SIR model and the model (1), respectively. The SIR model estimates around 12000 total infected cases while the model (1) estimates around 29000 cumulative infected cases. At September 1st, 2020, the Healthy Secretariat reported 28471 confirmed infected cases.

5 Numerical results (New Normality Data)

On June 1st, the Mexican government activated the so-called new normality strategy in order to open up activities which are considered essential to reactivate the country's economy. This situation caused a minor increase in almost the entire national territory, furthermore, it caused two plateaus

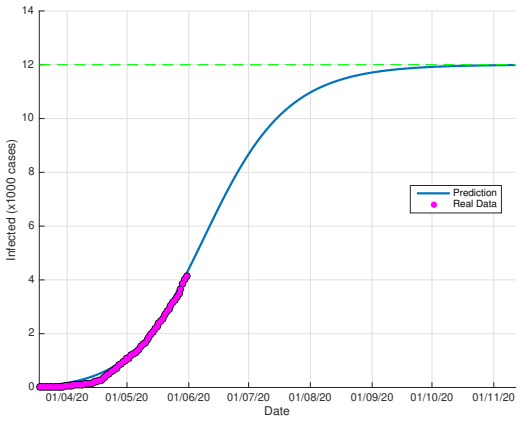


(a) Estimated cumulative infected cases in CDMX using the SIR model.

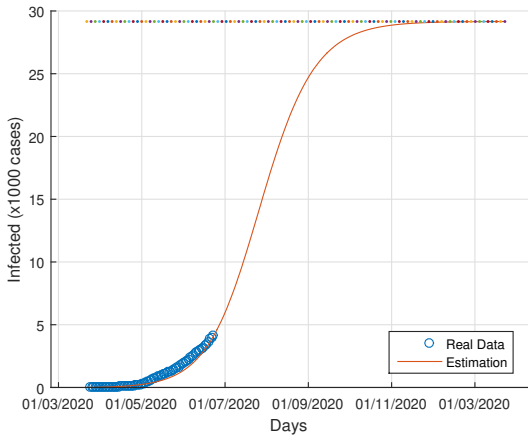


(b) Estimated cumulative infected cases in CDMX, using the model (1).

Figure 4: Estimated cumulative infected cases for CDMX.



(a) Estimation of cumulative infected cases in Tabasco, using the SIR model.



(b) Estimation of cumulative infected cases in Tabasco, using the model (1).

Figure 5: Estimated cumulative infected cases for Tabasco.

to be observed in the epidemiological curve. If we consider the data on the state of Tabasco after confinement until August 21st, 2020, the parameters were adjusted to this normality and we can see that the model predicts the maximum number of daily cases by mid-July as seen in figure 6. The official reports indicated that the maximum number of daily cases in Tabasco ocurrred in July 12th.

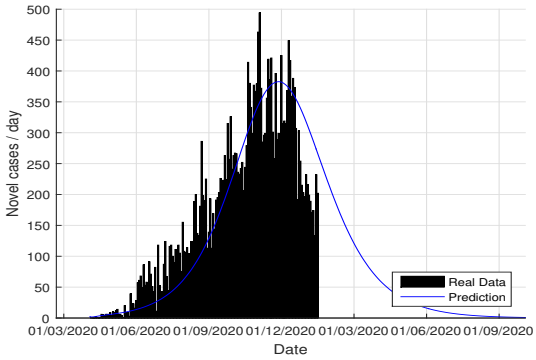


Figure 6: Estimation of daily cases using the model (1) for Tabasco considering the new normality.

For the case of Tabasco, in the new normality, the model (1) predicts a maximum of 640 hospitalized individuals at the beginning of August and around 4100 deaths are estimated at the end of the pandemic. In the same way, the model estimates the end of the pandemic by the end of November or the beginning of December as can be seen in Figures 7 and 8 (and 6).

Furthermore, also, we have obtained satisfactory respective estimations for the CDMX, which are not included here to obtain a document not so extensive.

The estimation of indicators such as number of infected, number of hospitalized and deaths are useful for the management of any pandemic, for example, in this case, in the planning of the hospital conversion.

6 Sensitivity analysis of the parameters

In this section we describe the sensitivity analysis of the parameters of model (1), this information is useful to determine the robustness of model predictions to parameter values. Furthermore, it is used to know the parameters that have a high impact on the threshold R_0 and these must be handled carefully . More specifically, sensitivity indices allows us to measure the relative change in a variable when a parameter changes. In order to do this analysis, we introduce the following definition.

Definition 1 *The normalized forward sensitivity index of R_0, which is differentiable with respect to a given parameter θ, is defined by*

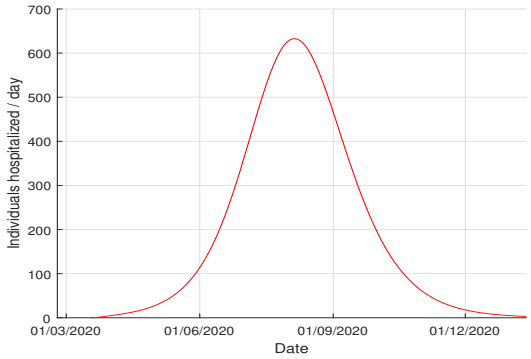


Figure 7: Estimation of hospitalized cases in Tabasco using the model (1).

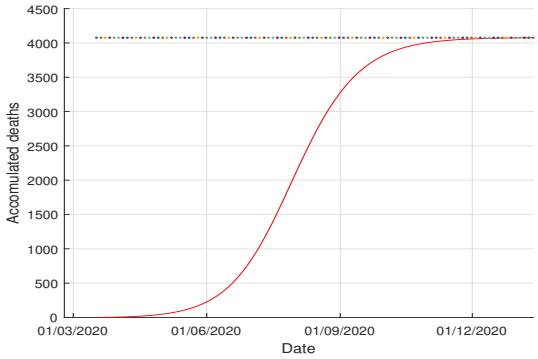


Figure 8: Estimated accumulated deaths in Tabasco using the model (1).

$$\Upsilon_{\theta}^{R_0} = \frac{\partial R_0}{\partial \theta} \frac{\theta}{R_0}.$$

The values of the sensitivity indices for the parameters values of CDMX and Tabasco, of Table 1, are showed in Table 2. These parameters have been taken from official data. We have seen the mathematical model (1) describes well the real data, giving rise to numerical results showed in the previous section.

Table 2: Sensitivity of R_0 in relation to the parameter values.

Parameter	CDMX	Tabasco
β_s	0.201	0.139
β_a	0.733	0.797
β_h	0.064	0.062
σ	0.192	0.367
ϕ_h	-0.04	0.017
γ_s	-0.09	0.074
γ_h	-0.039	0.038
μ	-0.733	-0.797
δ_s	-0.062	-0.047
δ_h	-0.025	0.024

Note that the sensitivity index may depend on several parameters of the system, but also can be constant, independent of any parameter. For example, $\Upsilon_\theta^{R_0} = 1$ means that increasing by a given percentage increases always R_0 by that same percentage. For the mathematical model (1), the estimation of a sensitive parameter should be carefully done, since a small perturbation in such parameter leads to relevant quantitative changes. On the other hand, the estimation of a parameter with a rather small value for the sensitivity index does not require as much attention to estimate, because a small perturbation in that parameter leads to small changes.

From Table 2, we see that the most sensitive parameters to the basic reproduction number R_0 of the COVID-19 model (1) are β_a and μ. However, these parameters were chosen from report of Healthy Secretary. In contrast, the lest sensitive parameter is δ_h.

7 Conclusion

We have introduced a compartmental mathematical model to understand the dynamic of the COVID 19 pandemic. This model considers a constant population splitted into susceptible class S, exposed class E, infectious with symptoms class I, asymptomatic infectious class I_a, hospitalized class H, recovery class R, and death class F. The used data correspond to two entities: Mexico city and the State of Tabasco, both considered during the pandemic as "red spotlights". First of all we showed the existence and stability of the free illnes equilibrium and then we considered two stages for the numerical study of model: The National Healthy Distance period and the "New normality" period. For comparison reasons, our simulations were done using our model an the classical SIR model. It must be mentioned that for all the numerical simulations, most of the parameters were taken from official government data such as the CONAHCYT and the updated reference literature linked to the presented models up to 2020.

As a first stage, the study is done during the lockdown (National Healthy Distance) strategy implemented from March 20th to May 31st. For both models, ours and SIR, the obtained numerical

estimations fit in an excellent way the reported data, as the accumulated number of infectious disease and the daily infectious cases as well as for the daily deaths (see Figures 4a-5b). However, it can be noticed that our model predicts better the infected accumulated cases, the accumulated deaths and the duration of the pandemic (see Figures 5b and 8). With respect to the reproductive number, for the model (1) and using data and parameters for the CDMX we obtain $R_0 = 1.79$, while for the SIR model the value is $R_0 = 1.21$. This means that our model is more realistic with respect to pandemic transmision speed, which can be corroborated with the monthly data until May 31, 2020.

In a second stage, simulations were done using the updated data until August 22th 2020, corresponding to the "New Normality" period when the Mexican government open the activities in sectors which they considered important for the economy, but when the peak's pandemic was not reached yet. The obtained numerical results with the model (1) fit in a satisfactory way and the predictions improve with respect to the obtained with the classic SIR Model (see Figure 6-8).

Finally it is important to mention that the inclusion of classes in the population, such as hospitalized and deaths cases, the models provide valuable information to the goverments during the management of pandemics.

References

[1] *COVID-19, México. https://coronavirus.gob.mx/datos/.* Conacyt, 2020.

[2] Wuhan seafood market pneumonia virus isolate wuhan-hu-1. *Complete genome*, pages 462–532, 2020.

[3] M. Batista. *Estimation of the final size of the COVID-19 epidemic.* medRxiv, 2020.

[4] F. Brauer, C. Castillo-Chavez, and Z. Feng. *Mathematical models in epidemiology.* New York: Springer-Verlag, 2019.

[5] P. V. den Driessche and J. Watmough. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Math Biosci*, 180:29–48, 2002.

[6] J. Djordjevic, C. J. Silva, and D. F. R. Torres. A stochastic sica epidemic model for hiv transmission. *Applied Mathematics Letters*, 84:168–175, 2018.

[7] N. M. Ferguson, D. Laydon, G. Nedjati-Gilani, N. Imai, K. Ainslie, M. Baguelin, S. Bhatia, A. Boonyasiri, Z. Cucunuba, and G. Cuomo-Dannenburg. Impact of non-pharmaceutical interventions (npis) to reduce covid-19 mortality and healthcare demand. *Imperial College COVID-19 Response Team*, 2020.

[8] F. Gompertz. On the nature of the function expressive of the law of h man mortality, and on a new mode of determining the value of life contingencies. in a letter to francis baily, esq. frs &c. *Philosophical transactions of the Royal Society of London*, 115:513–583, 1825.

[9] L. Hung. The sars epidemic in hong kong: what lessons have we learned? *J R Soc Med*, 96(8):374–378, 2003.

[10] A. Liénart and H. Chipart. Sur le signe de la partie réelle des racines d'une équation algébrique. *J Math Pures Appl (6 éme série)*, 10:291–346, 1914.

[11] F. Ndaïrou, I. Area, J. J. Nieto, and D. F. M. Torres. Mathematical modeling of covid-19 transmission dynamics with a case study of wuhan. *Chaos, Solitons and Fractals*, page 109846.

[12] A. G. M. Neves and G. Guerrero. Predicting the evolution of the covid-19 epidemic with the a-sir model: Lombardy, italy and são paulo state,brazi. *Nonlinear Phenomena Physica D*, 413:132693, 2020.

[13] A. Rachah and D. F. M. Torres. Dynamics and optimal control of ebola transmission. *Math Comput Sci*, 10:331–342, 2016.

[14] B. Tang, X. Wang, Q. Li, N. Bragazzi, S. Tang, Y. Xiao, and J. Wu. Estimation of the transmission risk of the 2019-ncov and its implication for public health interventions. *Journal of Clinical Medicine*, 9(2):462–532, 2020.

[15] K. K. To, O. T. Tsang, K. H. Chan, T. C. Wu, J. M. Chan, W. S. Leung, T. S. Chik, C. Y. Choi, and D. H. Kandamby. Consistent detection of 2019 novel coronavirus in saliva. *Clin Infect Dis Off Publ Infect Dis, Soc Am*, 71(15):841–843, 2020.

[16] O. Torrealba-Rodríguez, R. A. Conde-Gutiérrez, and A. L. Hernández-Javier. Modeling and prediction of covid-19 in mexico applying mathematical and computational models. *Chaos, Solitons and Fractals*, 138:109946, 2020.

[17] P. Verhulst. Notice sur la loi que la population poursuit dans son accroisse- ment. corresp. *Math Phys*, 10:113–121, 1838.

[18] E. Zowalaty, E. Mohamed, and J. D. Jarhult. From sars to covid-19: A previously unknown sars- related coronavirus (sars-cov-2) of pandemic potential infecting humans – call for a one health approach. *One Health*, 9, 2020.

Influencia de los factores socioeconómicos en la mortalidad por COVID-19 en México

Vanessa Itzel Soulé Flores [1] y Carlos Erwin Rodríguez Hernández-Vela[2]

[1]Facultad de Ciencias, Universidad Nacional Autónoma de México
[2]Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS, UNAM)

Resumen

La pandemia por el virus SARS-CoV-2 mostró la necesidad de realizar estudios multidisciplinarios que ayudaran a entender el riesgo de mortalidad por COVID-19. En este trabajo se buscó determinar la influencia de los factores socioeconómicos que existían dentro de la población mexicana mediante la aplicación del modelo de regresión lineal múltiple, para esto el análisis se hizo a nivel municipal en las 3 primeras olas de la pandemia, definiendo grupos poblacionales homogéneos para identificar cuáles son los factores socioeconómicos que tienen mayor influencia y nos ayudan a describir el riesgo de fallecer por COVID-19 en México.

Palabras clave: Multiple linear regression; socioeconomics; COVID-19 pandemic; Statistics

1 Introducción

Los primeros casos de la enfermedad COVID-19 en México, se detectaron a finales de febrero de 2020 desencadenando un brote de contagios que aumentó drásticamente a medida que la pandemia avanzaba, [9]. Es crucial tener en cuenta que nunca se pudo conocer el número real de personas infectadas, esto debido tanto a portadores asintomáticos, como a la falta de registro de enfermos que no asistieron a clínicas, centros de salud u hospitales[2] en busca de atención médica.

Es importante observar que el acceso a servicios de calidad en salud, educación, empleo y vivienda depende en gran medida del nivel socioeconómico de las personas. Estudios realizados en México han proporcionado evidencia concluyente de que la mayor concentración de patologías complejas y exceso de mortalidad se encuentra en las regiones o grupos donde prevalecen elevados niveles de marginación y exclusión social [3]. En todo el territorio mexicano, se observan notables contrastes entre distintas zonas geográficas, grupos étnicos y estratos socioeconómicos. Dada la relación entre estas disparidades y el alto índice de contagios y muertes causadas por las condiciones de salud predominantes, que a menudo facilitan la transmisión del virus en personas con comorbilidades, es

esencial analizar el impacto de la pandemia desde una perspectiva que englobe tanto los aspectos sociales como los económicos y los médicos.

El objetivo de este estudio es identificar factores socio-económicos que pudieran tener influencia en el riesgo de mortalidad por COVID-19. Lo anterior a un nivel de desagregación geográfica municipal. Para alcanzar esta meta, se utiliza la teoría de regresión múltiple, en donde se busca modelar el riesgo de mortalidad por COVID-19 como función lineal de muchas variables socioeconómicas. En este caso, la etapa clave sería la selección de variables.

El artículo se organiza de la siguiente manera, en la Sección 2 se da un breve resumen de la teoría de regresión múltiple. Las fuentes de información utilizadas, así como la manera en la que se construyó la variable dependiente para el modelo de regresión múltiple, se describen en la Sección 3. En la Sección 4, se identifica que el tamaño de la población en los municipios de México es muy heterogéneo y se presenta una solución para que esto no afecte el ajuste de los modelos de regresión. La Sección 5, presenta la forma en la que se construyeron las variables que nos permitirán incluir los factores socioeconómicos en el modelo y en la Sección 7 se presenta el ajuste de los modelos, así como las variables que contribuyen a explicar el riesgo de mortalidad. Finalmente, en la Sección 8 se presentan las conclusiones.

2 Modelo de regresión

La teoría de regresión puede consultarse en los libros clásicos [5] y [6]. Para una visión aplicada en donde se usa el paquete estadístico R, [7], ver el libro [11]. A continuación, se brinda una descripción muy breve de la teoría de regresión lineal múltiple enfocándonos en las herramientas que se usarán más adelante.

Un modelo de regresión es un modelo matemático que, de manera muy general, busca alcanzar dos objetivos 1) entender o identificar la relación que existe entre una variable dependiente $\mathbf{Y}_{n\times 1}$ y un conjunto de variables independientes $\mathbf{X} = (\mathbf{1}, X_1, X_2, \dots X_k)_{n\times(k+1)}$ y 2) predecir los valores de la variable dependiente usando las variables independientes. Estos dos objetivos generalmente se contraponen debido a que modelos más simples suelen ser más fáciles de interpretar, pero modelos con pocas variables suelen no capturar la complejidad de los datos (o predecir) como modelos con más variables. Adicionalmente, añadir demasiadas variables puede llevar a un sobreajuste, donde el modelo se ajusta demasiado bien a los datos de entrenamiento, pero no es capaz de describir datos nuevos. En muchos casos, se busca un equilibrio entre la simplicidad del modelo, su capacidad de predicción y la interpretación que nos brinda del fenómeno que buscamos entender; por lo que es indispensable tener claro lo que se busca en cada problema que se quiera analizar. En este trabajo lo que se desea es alcanzar el primer objetivo.

El modelo de Regresión Lineal Múltiple (RLM) puede ser escrito en forma matricial como:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{1}$$

en donde $\boldsymbol{\varepsilon}_{n\times 1}$ es un vector aleatorio, es decir, sus elementos ε_i son variables aleatorias. El vector $\boldsymbol{\beta}_{(k+1)\times 1}$ representa los coeficientes de regresión y son parámetros que se asumen como constantes

desconocidas. Cada elemento β_j de $\boldsymbol{\beta}$ es un coeficiente parcial de regresión que refleja el cambio en la variable dependiente $\mathbf{Y}$ por unidad en la variable independiente X_j. Cabe mencionar que en términos generales, la columna de unos dentro de nuestro conjunto de variables independientes, no se considera una variable independiente en el sentido tradicional, ya que no representa una característica específica del conjunto de datos original. Sin embargo, en el contexto del modelo de regresión lineal, se le trata como una variable adicional que permite al modelo ajustar la ordenada al origen. Por lo tanto, aunque no es una variable en el sentido típico, se incluye en la matriz de variables independientes para facilitar el ajuste del modelo.

Los supuestos básicos de la RLM se centran en el vector aleatorio $\boldsymbol{\varepsilon}$, conocido como el vector de errores del modelo. Los supuestos son: 1) la esperanza de los errores, ε_i, es cero, 2) su varianza es constante (σ^2) y 3) los errores son independientes entre sí. Adicionalmente, se tiene un supuesto distribucional que nos permitirá realizar pruebas de bondad de ajuste del modelo de RLM, i.e. pruebas estadísticas para determinar que tan bien describe el modelo a la variable dependiente. El supuesto distribucional 4) es el error ε_i que sigue una distribución normal con media cero y varianza σ^2. Considerando los supuestos básicos y que se conoce de antemano la matriz de variables independientes $\mathbf{X}$, es fácil observar que la relación media entre la variable dependiente e independientes está dada por $\mathbb{E}(\mathbf{Y}|\mathbf{X}) = \mathbf{X}\boldsymbol{\beta}$.

2.1. Estimación

El análisis de RLM nos da herramientas para, primero, estimar $\boldsymbol{\beta}$ y a continuación determinar que tan bien nuestro modelo ajusta los datos. Para estimar los coeficientes del modelo, lo más sencillo es utilizar el método de mínimos cuadrados, aquí se busca determinar el vector $\hat{\boldsymbol{\beta}}$ que minimiza la distancia

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \varepsilon_i^2 = \ (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}), \tag{2}$$

Derivando con respecto a $\boldsymbol{\beta}$, igualando a cero y simplificando se llega a

$$\mathbf{X}'\mathbf{X}\boldsymbol{\beta} = \mathbf{X}'\mathbf{Y}, \tag{3}$$

Multiplicando ambos lados de la ecuación (3), por la inversa de $\mathbf{X}'\mathbf{X}$, se obtiene el estimador para $\hat{\boldsymbol{\beta}}$ por mínimos cuadrados:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{Y}), \tag{4}$$

La matriz $(\mathbf{X}'\mathbf{X})^{-1}$ existe si las columnas de $\mathbf{X}$ (las variables independientes) son linealmente independientes. Por lo tanto, el modelo ajustado de regresión queda definido como $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}}$.

2.2. Comparación de modelos

Es sencillo definir una medida del error en la estimación, ésta es simplemente la suma del cuadrado de los errores en la estimación, i.e. $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$. También se puede demostrar que la varianza de

la variable dependiente, se divide en la cantidad de variabilidad en las observaciones explicada por el modelo de regresión, más el error en la estimación, i.e.

$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(\bar{y} - \hat{y}_i)^2 + \sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$$

A partir de la descomposición de la varianza, se obtiene el coeficiente de determinación

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2},$$

que indica la proporción de la varianza de los datos, efectivamente explicada por el modelo. La R^2 muy bien podría usarse para comparar modelos de RLM, sin embargo en el caso de modelos anidados (modelos donde las variables independientes de un modelo son un subconjunto de las variables independientes de otro modelo) es fácil verificar que modelos con un mayor número de variables siempre producen mayores valores de R^2, por lo que la R^2 no es una estadística recomendable para comparar modelos, obviamente lo que se busca es un modelo con el menor número de variables y que describa bien los datos. Estos modelos son más fáciles de interpretar, manejar y mantener.

Para resolver el problema con la R^2, una alternativa es comparar modelos usando el criterio de información de Akaike (AIC), [1]. El AIC se basa en el cálculo de la estadística

$$\text{AIC} = n \log\left(\sum_{i=1}^{n}(y_i - \hat{y}_i)^2/n\right) + 2k,$$

en donde la parte izquierda es una medida de falta de ajuste del modelo, n es el número de observaciones y $2k$ es una penalización por el número de variables independientes que se incluyen en el modelo. Al comparar modelos, se prefieren modelos con menor AIC.

La manera obvia para elegir el mejor modelo es hacer todas las regresiones posibles (todas las posibles combinaciones de variables independientes) y para cada regresión calcular el AIC. Sin embargo, el número de regresiones posibles con k variables independientes es 2^k. Si tenemos 20 variables independientes sería necesario ajustar $2^{20} = 1,048,576$ modelos, lo que quizá sea factible pero sería computacionalmente muy demandante. Sin embargo, si se tienen 30 variables, el número de modelos a considerar es de $2^{30} = 1,073,741,824$, lo que ya no es viable. Por lo anterior, las estrategias más utilizadas para seleccionar variables son los métodos de selección **paso a paso**. Estas estrategias consisten en agregar o eliminar una variable cada vez, comparando el modelo actual con el siguiente basándonos en el AIC (aunque podrían usarse otros criterios). Este tipo de métodos tiene 3 posibilidades:

Forward: Selecciona paso a paso hacia adelante. Empieza con el modelo más simple y va agregando una a una las variables hasta que el modelo deje de mejorar.

Backward: Selecciona paso a paso hacia atrás. Empieza con el modelo que incluye a todas las variables y elimina una a una, hasta que el modelo deje de mejorar.

Both: Selecciona paso a paso en ambas direcciones. Es una combinación de los dos métodos anteriores.

La librería `MASS`, [10], del paquete estadístico R, implementa estás técnicas vía la función `stepAIC`.

3 Los Datos

Fundamentalmente, se contó con dos fuentes de información:

- La base de datos de la Secretaría de Salud del Gobierno de México SINAVE/SISVER, [8]. Esta base de datos registra información de todas aquellas personas que acudieron a alguna clínica, centro de salud u hospital público o privado por sospecha de haberse contagiado de COVID-19. Esta información se tiene a nivel paciente, pero obviamente no tiene información de identificación.
- La base de datos del Censo de Población y Vivienda 2020 del INEGI, [4]. El Censo contiene información detallada sobre varias características de la población y de sus viviendas en todo el país. Puede descargarse en el sitio de INEGI y el nivel de desagregación puede ser manzana, AGEB, localidad, municipio, estado o nacional.

Inicialmente, se intentó unir la información de ambas bases de datos a nivel localidad (considerando sólo a la CDMX), pero fue imposible pues las claves de localidad en la base SINAVE/SISVER no corresponden con las mismas claves en las bases de datos del INEGI. En cambio, a nivel municipal existía suficiente homogeneidad entre ambas bases de datos para poder unirlas: ambas bases de datos cuentan con la clave de estado y clave de municipio de INEGI, además existe coincidencia de más del 98 % en estas variables de ambas bases de datos.

3.1. Variable dependiente: riesgo de mortalidad

La última versión de la base SINAVE/SISVER, que se descargó para este trabajo, fue la del 16 de marzo del 2022. En esta actualización, se contaba con información de $15,396,315$ casos totales, incluyendo positivos, sospechosos y negativos. Para obtener nuestra base de datos de trabajo, se filtraron los casos positivos a COVID-19 ($2,624,272$ observaciones). La evolución del número de casos positivos por mes se presenta en la Figura 1.

Es fácil apreciar que hasta marzo del 2022, México había sido impactado por cuatro olas de la pandemia por COVID-19. Estas sucedieron en los meses:

- Primera ola: junio, julio y agosto del 2020.
- Segunda ola: diciembre de 2020, enero y febrero del 2021.
- Tercera ola: julio, agosto y septiembre del 2021.
- Cuarta ola: diciembre de 2021, enero y febrero del 2022.

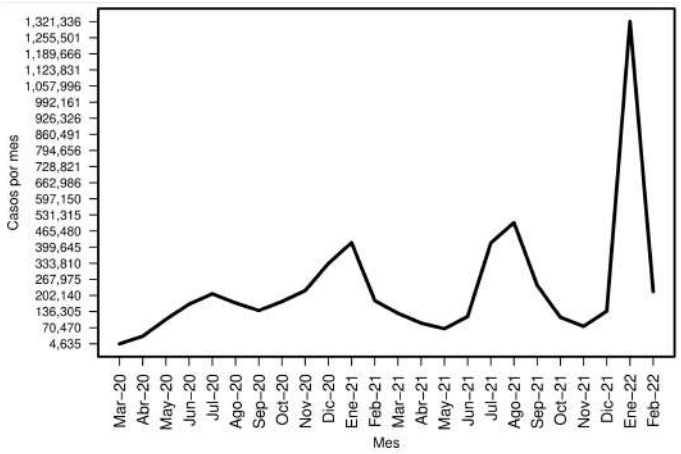


Figura 1: Evolución mensual del número de casos positivos a COVID-19.

Lo anterior, considerando el mes en donde se observó el pico de contagios, junto con el anterior y el posterior.

Como se mencionó en la introducción, el objetivo de este trabajo se centra en describir el riesgo de mortalidad por COVID-19, esto es

$$p = 100(d/c),$$

en donde d es el número de defunciones ocasionadas por el COVID-19 y c es el número de casos positivos a COVID-19. En la Figura 2, se muestra la evolución mensual del riesgo de mortalidad.

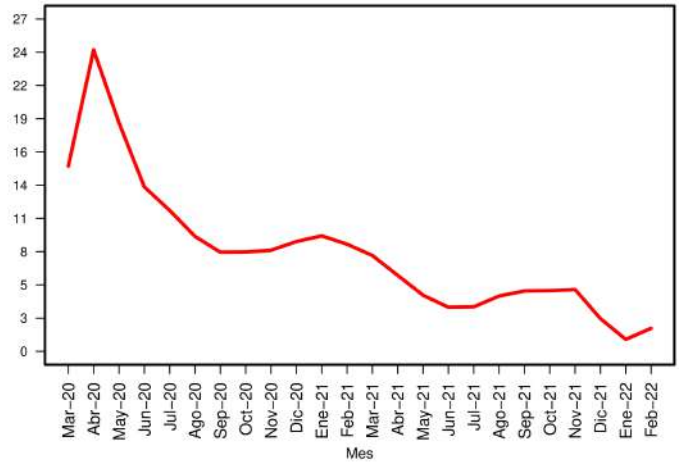


Figura 2: Evolución del riesgo de mortalidad por COVID-19.

Observando las Figuras 1 y 2 notamos que durante la primera ola de la pandemia había relativamente pocos casos, pero la mortalidad fue la más alta que se observó durante toda la pandemia.

En cambio, en la cuarta ola se dio el mayor pico en el número de casos y la menor mortalidad. Recordemos que al principio de la pandemia el conocimiento que se tenía de esta nueva enfermedad era mínimo, incluso algunos sectores de la población no le dieron relevancia. A medida que pasaron los meses, el conocimiento que existía de la enfermedad iba creciendo, e incluso a finales del año 2020 empezó la vacunación contra el COVID-19. Considerando con cuidado que estos factores iban evolucionando en el tiempo, se pensó que la mejor estrategia sería modelar el riesgo de mortalidad en cada una de las primeras tres olas de la pandemia, para finalmente intentar predecir lo que sucedería en la cuarta ola.

Filtrando la información para cada ola de la pandemia y obteniendo el porcentaje de fallecimientos por COVID-19, en cada municipio de la República Mexicana, se obtuvieron cuatro bases de datos. Cada una con aproximadamente $2,437$ registros. Dependiendo de la ola que se esté analizando, el número de municipios en donde se observaron casos positivos a COVID-19 cambia. En la primera ola, se observaron casos positivos en un total de $2,142$ municipios, en la segunda $2,219$ y en la tercera ola $2,306$. Cada base de datos, además del porcentaje de fallecimientos por COVID-19 en cada municipio, tiene la clave de estado y de municipio de cada registro. Estas variables nos dan la oportunidad de unir estas bases de datos con la información del INEGI.

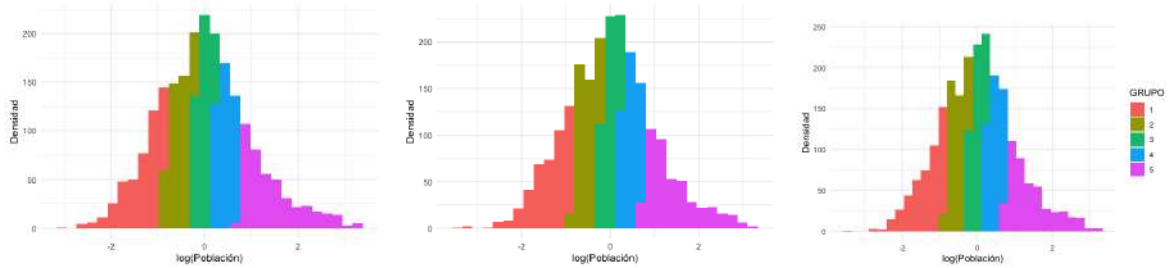
Es importante mencionar que la base SINAVE/SISVER cuenta con variables como la fecha de inicio de síntomas, fecha de fallecimiento, el sexo, edad, tipo de paciente (hospitalizado o ambulatorio), comorbilidades (9 variables dicotómicas que indican si el paciente padecía o no cierta comorbilidad) entre muchas variables más. Sin embargo, para alcanzar los objetivos delineados en este trabajo, las variables relevantes fueron fecha de inicio de síntomas (para determinar la ola de la pandemia a la que el paciente pertenece) y la fecha de fallecimiento (para crear una variable dicotómica con la información de los fallecimientos).

4 Número de habitantes en los municipios de México

La República Mexicana está integrada por $2,471$ municipios de los cuales, según la sección anterior, $2,437$ presentaron al menos algún caso positivo a COVID-19. Analizando el tamaño de la población de los municipios del país, se observó gran heterogeneidad. Por ejemplo, existe un municipio con sólo 81 habitantes y otro con 1,922,523. Debido a que se busca identificar la relevancia de diversos factores socioeconómicas para describir el riesgo de mortalidad, por COVID-19, en los municipios de México, lo anterior representa una dificultad ya que las conclusiones del análisis podrían estar dictadas mayormente por la disparidad entre el tamaño de la población de los municipios y no por el efecto de las variables socio-económicas.

Para resolver el problema identificado en el párrafo anterior, se dividió a los municipios en grupos de acuerdo al número de habitantes. Primero, se aplicó la función logaritmo al tamaño de la población, generando una distribución normal. Después, se usaron cuantiles para obtener 5 grupos, cada uno compuesto por municipios con aproximadamente el mismo tamaño poblacional. Al agregar esta información a la base de datos con los porcentajes de fallecimientos por COVID-19 (de las 3 olas de la pandemia), se observó que la distribución normal se mantiene. En la Figura 3 se muestra en colores los 5 grupos, notando que el grupo con mayor densidad, es decir, el grupo que presenta más

casos positivos son los municipios medianos (grupo 3 - municipios entre 9 mil y 19 mil habitantes).



(a) 1ra ola **(b)** 2da ola **(c)** 3ra ola

Figura 3: Distribución del tamaño de los municipios (con base logarítmica) en donde se observaron casos positivos en cada ola de la pandemia.

Esto nos ayudará a que las conclusiones del análisis, para los municipios del mismo tamaño, sean comparables entre las tres olas de la pandemia. Además, podremos identificar las variables significativas específicas en cada grupo poblacional.

5 Factores socioeconómicos

En la sección anterior, se describe el esfuerzo realizado para lograr que el tamaño de la población de los municipios no tenga impacto en las conclusiones que se pudieran obtener. Con el mismo objetivo, se construyen las variables socioeconómicas que se incluirían en nuestros modelos de RLM.

De manera general, para construir las variables independientes, primero se extrajeron los subtotales a nivel municipal del Censo, a continuación se calculó la proporción de cada una y se multiplicó por cien para tener porcentajes. Por ejemplo, para obtener el porcentaje de analfabetas por municipio, primero, se tomó la población total de 15 años o más analfabeta, segundo, se dividió entre la población total de 15 años o más y finalmente se multiplicó por cien. Se hizo algo similar para la población económicamente activa, la población sin seguridad social, etc. En otros casos, cuando era claro que no habría impacto del tamaño de la población de los municipios, simplemente se tomó el sub-total a nivel municipal directamente de la base de datos del Censo: grado promedio de escolaridad, promedio de hijos nacidos vivos, promedio de ocupantes en viviendas particulares habitadas, etc. Las variables socioeconómicas construidas a partir de variables del Censo, junto con sus nombres mnemotécnicos, consideradas en el análisis se describen en la Tabla 1 (18 variables).

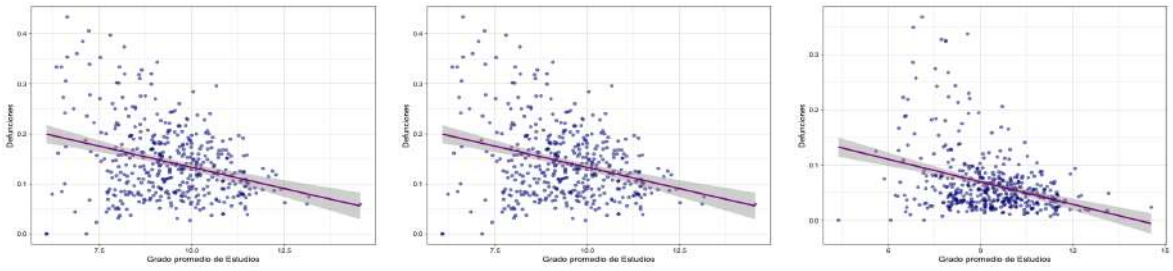
Las 18 variables se pegaron a las bases de datos con el porcentaje de fallecimientos por municipio y para cada una de las tres olas de la pandemia.

Tabla 1: Variables socioeconómicas a nivel municipal, junto con sus nombres mnemotécnicos.

1) Grado promedio de escolaridad de la población de 15 años o más (GPEST)	2) % Población económicamente activa de la población de 15 años o más (PEA)	3) % Población sin seguridad social (SSS)
4) Promedio de ocupantes por vivienda (POV)	5) Promedio de hijos nacidos vivos (PHNV)	6) % Población habla lengua indígena (PHLI)
7) % Población con analfabetismo (ANALF)	8) % Ocupantes en viviendas particulares sin drenaje ni excusado (SDE)	9) % Ocupantes en viviendas particulares sin energía eléctrica (SEE)
10) % Ocupantes en viviendas particulares sin agua entubada (SAE)	11) % Ocupantes en viviendas particulares con piso de tierra (VPT)	12) % Viviendas particulares con hacinamiento (VHAC)
13) % Población en localidades con menos de 5000 habitantes (P5000)	14) % Población masculina (POBMAS)	15) % Población de 60 años y más (P60YM)
16) % Población ocupada con ingresos de hasta 2 salarios mínimos (P2SM)	17) % Población con diabetes (PDIAB)	18) % Población con obesidad (POBES)

6 Análisis exploratorio

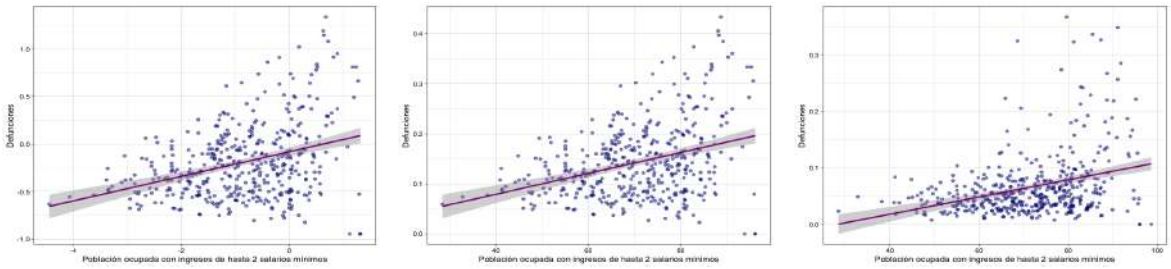
Una vez construidas nuestras bases de datos de trabajo, se realizó un análisis descriptivo exploratorio para detectar posibles asociaciones entre variables. A continuación se muestran un par de diagramas de dispersión en donde se presenta el porcentaje de fallecimientos por COVID-19 en los municipios del país contra el grado promedio de escolaridad y la población ocupada.



(a) 1ra ola **(b)** 2da ola **(c)** 3ra ola

Figura 4: Porcentaje de fallecimientos (y) Vs grado promedio de escolaridad (x) a nivel municipal.

En las tres olas, es posible identificar que el riesgo de mortalidad por COVID–19



(a) 1ra ola **(b)** 2da ola **(c)** 3ra ola

Figura 5: Porcentaje de fallecimientos (y) Vs población ocupada con ingresos de hasta 2 salarios mínimos. (x) a nivel municipal.

- Disminuye en municipios con mayor grado de escolaridad. (Ver Figura 4)
- Aumenta en municipios con población ocupada con ingresos de hasta 2 salarios mínimos. (Ver Figura 5)
- Aumenta en municipios con mayor porcentaje viviendas con hacinamiento (no se muestra la gráfica).
- Aumenta en municipios con mayor promedio de ocupantes por vivienda (no se muestra la gráfica).

7 Ajuste de modelos y selección de variables

Con el objetivo de describir el riesgo de fallecimiento por COVID-19, en cada municipio del país, se ajustaron 15 modelos de RLM. Estos 15 modelos consideran a los 5 grupos de municipios construidos a partir del número de habitantes en cada municipio, así como las 3 olas de la pandemia (ver Figura 3). Adicionalmente, para identificar a las variables socioeconómicas que pudieran tener mayor relevancia para describir el riesgo de fallecimiento, también se realizó una selección de variables. La estrategia fue la de selección hacia adelante y hacia atrás, buscando la combinación de variables que minimizará el criterio de información de Akaike (AIC). Los resultados se muestran en la Tabla 2 para la primera ola, Tabla 3 segunda ola y Tabla 4 para la tercera ola.

En todos los casos, se aprecia que los coeficientes de determinación, las R^2 de los modelos, son muy pequeñas. Es importante mencionar, que estamos trabajando datos reales y que el objetivo de este proyecto es identificar variables socioeconómicas que pudieran estar relacionadas con el riesgo de mortalidad en los municipios de México. Por lo tanto, nos concentramos sólo en los coeficientes de las variables seleccionadas por el AIC en cada caso.

Los coeficientes de regresión estimados son las pendientes del modelo de RLM, expresión 1, veamos como interpretarlos en un par de casos. Para la primera ola, si en algún municipio, de los grupos 2 ó 3, el grado promedio de escolaridad aumentara en 1 año. Entonces, las defunciones

Tabla 2: Ajuste y selección de modelos para la primera ola

	Grupos de acuerdo al tamaño de los municipios				
Variables	(1)	(2)	(3)	(4)	(5)
PDIAB	−0.692* (0.375)				
PEA	0.306* (0.180)			−0.179 (0.120)	
SSS				0.110* (0.065)	0.094** (0.042)
P60YM	0.899** (0.379)			0.588*** (0.207)	
POBES		0.303* (0.160)	0.677*** (0.183)		
PHNV		−7.612* (4.096)			
GPEST		−2.757** (1.302)	−3.849*** (1.340)		
PHLI					−0.104*** (0.034)
SDE	1.126*** (0.297)	0.695*** (0.224)			−0.531*** (0.132)
SEE	−1.575* (0.925)	−1.340*** (0.383)		−0.370* (0.191)	
VHAC					0.223*** (0.072)
ANALF			−0.605*** (0.221)		0.491*** (0.150)
POBMAS			−2.197** (1.061)		
SAE	0.293* (0.161)		0.133 (0.084)		
POV					−3.862*** (1.166)
P2SM		0.348** (0.137)	0.343*** (0.111)	0.204*** (0.064)	0.116*** (0.039)
Constant	−0.634 (10.633)	21.505 (26.381)	111.978** (55.307)	1.721 (9.905)	11.755*** (4.120)
Observations	429	428	428	428	429
R^2	0.060	0.089	0.084	0.054	0.214
Adjusted R^2	0.047	0.076	0.071	0.043	0.201
Residual Std. Error	26.479	21.779	17.019	12.608	6.613
F Statistic	4.506***	6.847***	6.435***	4.837***	16.365***

Note: *p<0.1; **p<0.05; ***p<0.01

Tabla 3: Ajuste y selección de modelos para la segunda ola

	Grupos de acuerdo al tamaño de los municipios				
Variables	(1)	(2)	(3)	(4)	(5)
POBES	0.373** (0.158)	0.247* (0.148)	0.345* (0.185)	0.330*** (0.114)	
PHNV	10.915** (4.571)				12.035*** (2.228)
VHAC		−0.245* (0.134)			−0.224*** (0.072)
GPEST	8.816*** (2.547)	−3.906*** (1.097)	−4.129*** (1.302)		
SSS	−0.228* (0.125)				0.340*** (0.049)
PEA			0.234* (0.130)		
ANALF	1.271*** (0.346)		−0.440* (0.232)		
PHLI				−0.093*** (0.031)	
POV	−7.590** (3.607)			7.875*** (2.616)	−2.501* (1.454)
SDE					−0.425*** (0.150)
SAE	−0.388** (0.169)				0.124* (0.069)
P2SM	0.526** (0.217)				
P5000				0.078*** (0.024)	0.053** (0.025)
P60YM			0.420* (0.254)	1.269*** (0.305)	
POBMAS			−1.839* (1.073)		
Constant	−111.784*** (42.096)	50.139*** (10.600)	117.604** (56.096)	−41.898*** (14.969)	−8.235 (5.528)
Observations	444	444	443	444	444
R^2	0.072	0.029	0.047	0.093	0.214
Adjusted R^2	0.055	0.023	0.034	0.083	0.201
Residual Std. Error	29.452	23.999	17.433	12.719	8.196
F Statistic	4.213***	4.408***	3.594***	9.005***	16.939***

Note: *p<0.1; **p<0.05; ***p<0.01

Tabla 4: Ajuste y selección de modelos para la tercera ola

	Grupos de acuerdo al tamaño de los municipios				
Variables	(1)	(2)	(3)	(4)	(5)
POBES	−0.400*** (0.141)			0.319*** (0.101)	
PDIAB	0.684* (0.398)				
ANALF	0.526*** (0.181)		−0.632*** (0.229)		
PHNV				3.295 (2.324)	7.694*** (0.896)
POBMAS	2.310*** (0.716)			−1.610*** (0.503)	
VPT	−0.276* (0.152)			−0.121* (0.072)	
GPEST		−2.631*** (0.711)	−5.201*** (1.106)	−3.481*** (0.594)	
PEA			0.170 (0.105)	0.206** (0.082)	
SSS		0.218*** (0.067)	0.299*** (0.071)	0.240*** (0.045)	0.184*** (0.027)
P60YM				0.369 (0.234)	0.139* (0.076)
POV		3.762* (1.971)		3.522* (1.899)	
SAE		−0.209*** (0.079)			0.090** (0.036)
SDE				−0.121 (0.078)	−0.371*** (0.073)
VHAC	0.317* (0.183)	−0.253** (0.114)		−0.240*** (0.078)	
PHLI			0.056 (0.038)		
P2SM		0.162* (0.093)	0.154* (0.093)		
P5000				0.079*** (0.020)	
Constant	−115.367*** (34.148)	6.903 (11.829)	29.754* (15.378)	65.262*** (25.185)	−16.417*** (2.180)
Observations	462	461	461	461	461
R^2	0.064	0.082	0.141	0.274	0.253
Adjusted R^2	0.052	0.070	0.129	0.254	0.245
Residual Std. Error	22.685	14.747	14.246	8.737	4.864
F Statistic	5.186***	6.789***	12.378***	14.086***	30.866***
Note:					*p<0.1; **p<0.05; ***p<0.01

disminuirían en 2.76 % y 3.85 % respectivamente. Observando ahora los municipios en el grupo 4 en las tres olas, si el porcentaje de la población de 60 años y más aumentará en 1 %, entonces las defunciones aumentarían 0.59 % en la primera ola, 1.27 % en la segunda ola y 0.37 % en la tercera ola. Finalmente, es interesante observar que durante la tercera ola y para los municipios más grandes (grupo 5), si aumentara el promedio de hijos nacidos vivos en uno, entonces el porcentaje de defunciones aumentaría en 7.7 %.

Analizando los 15 modelos ajustados, se identificó que uno de los factores socio-económicos más relevantes fue el grado promedio de estudios y se observa que municipios con mayor grado de escolaridad presentaron menor riesgo de mortalidad por COVID-19. Principalmente, en municipios de los grupos 2, 3 y 4 donde se tiene una población mayor a 3,419 habitantes y menor a 45,399. A continuación, se determinó que las variables relacionadas con el empleo también influyeron fuertemente pero en un modo negativo. Es decir, si el porcentaje población económicamente activa aumentara en los municipios medianos (al menos en la segunda y tercera ola), entonces aumentaría el riesgo de fallecer. En el caso de los municipios con el porcentaje de población ocupada con ingresos de hasta 2 salarios mínimos más grandes, se observó un alto riesgo de mortalidad en la primera ola de la pandemia. Durante este periodo, muchas personas quedaron sin ingresos, principalmente las que vivían de la economía informal, que por lo mismo no contaban con seguridad social para atenderse; y aunado a la ignorancia y desconocimiento del virus, causando un alto grado de mortalidad, [2].

Para tener una visión global, se construyó la Tabla 5 que muestra las variables socioeconómicas que tuvieron algún grado de asociación (positiva o negativa) con el porcentaje de fallecimientos por COVID-19 en al menos dos modelos de cada ola de la pandemia.

Un primer detalle, que es fácil apreciar, es que el número de variables socio-económicas signifi-

Tabla 5: Variables socioeconómicas que mostraron tener algún grado de asociación (positiva o negativa) con el porcentaje de fallecimientos a nivel municipal en cada una de las olas de la pandemia por COVID-19.

Variables – Fechas	**1ra ola (Junio 2020 - Agosto 2020)**	**2da ola (Diciembre 2020 - Febrero 2021)**	**3ra ola (Julio 2021 - Septiembre 2021)**
Grado promedio de escolaridad de la población de 15 años o más	✓	✓	✓
% Población económicamente activa de la población de 15 años o más	✓		✓
% Población sin seguridad social	✓	✓	✓
Promedio de ocupantes por vivienda		✓	✓
Promedio de hijos nacidos vivos		✓	✓
% Población habla lengua indígena			
% Población con analfabetismo	✓	✓	✓
% Ocupantes en viviendas particulares sin drenaje ni excusado	✓		✓
% Ocupantes en viviendas particulares sin energía eléctrica	✓		
% Ocupantes en viviendas particulares sin agua entubada	✓	✓	✓
% Ocupantes en viviendas particulares con piso de tierra			
% Viviendas particulares con hacinamiento		✓	✓
% Población en localidades con menos de 5000 habitantes		✓	
% Población masculina			✓
% Población de 60 años y más	✓	✓	✓
% Población ocupada con ingresos de hasta a 2 salarios mínimos	✓	✓	✓
% Población con diabetes			
% Población con obesidad	✓	✓	✓

cativas fue aumentando conforme iba evolucionando la pandemia, a pesar de que el conocimiento acerca de la enfermedad aumentaba así como la vacunación. Esto puede deberse al aumento de casos positivos de COVID–19 que hubo en la segunda y tercer ola con respecto a la primera, Figura 1.

Los factores socioeconómicos fueron sumamente importantes para determinar como sería la evolución del paciente llegando a fallecer en algunos casos si no se contaba con algún servicio de salud, por eso la variable del *porcentaje de población sin seguridad social* resultó muy significativa, sobre todo en los municipios con mucha población. De la misma forma, si el *promedio de hijos nacidos vivos* es alto o en localidades con menos de 5,000 habitantes (que por lo general son zonas marginadas), se corre un mayor riesgo de fallecer por COVID-19. Si consideramos las condiciones de vivienda, las personas que residen en *viviendas sin agua entubada* resultan más afectadas en esta pandemia, principalmente en municipios donde la población es muy grande. Esto obviamente, pues el virus se transmite al tener contacto con él (mientras más grande la población mayor es el riesgo) siendo indispensable tener agua para lavarse las manos y sin ella, aumenta el riesgo.

Por lo que podemos concluir que las variables socioeconómicas más significativas e influyentes en las defunciones por estar presente en las 3 olas de contagios fueron el *grado promedio de estudios*, la *población sin seguridad social*, la *población analfabeta*, el *porcentaje de ocupantes en viviendas sin agua entubada*, la *población de 60 años y más*, la *población ocupada con ingresos de hasta 2 salarios mínimos*, y la *población con obesidad*. Las cuales influyeron para un mayor o menor número de defunciones de acuerdo a cada ola de la pandemia y a cada grupo poblacional.

8 Conclusiones

Analizando la Pandemia por COVID–19, sin considerar las condiciones de salud de la población, los datos ajustados con 15 modelos de RLM muestran que los factores socioeconómicos sí influyen en el riesgo de mortalidad por COVID-19, dependiendo la ola de la pandemia que se analice y el tamaño del municipio donde se encuentre una persona. De forma concisa, se identificó que el factor socioeconómico con mayor influencia es el *grado promedio de estudios*, ya que a mayor grado de escolaridad la probabilidad de fallecer por COVID-19 es menor. Principalmente en los municipios medianos (entre 3,419 y 45,399 habitantes) en donde el riesgo de mortalidad disminuiría, en promedio, 3.75 % si se aumentara en una unidad el grado promedio de estudios.

En el trabajo de tesis que da origen a este artículo, también se hizo una predicción del riesgo de mortalidad para los municipios grandes de la cuarta ola de la pandemia. En este caso, se utilizó el modelo obtenido para los municipios grandes en la tercer ola. Esto no se presenta pues el objetivo principal era el de identificar factores socio-económicos que influyen en el riesgo de fallecer por COVID-19 que es el que se describe en este texto.

La pandemia representa un problema complejo ya que las condiciones entre cada ola fueron cambiando en el tiempo, sin embargo, las condiciones sociales y económicas no cambian a la misma velocidad. Por lo cual se necesitan enfocar esfuerzos en mejorar la educación a nivel nacional y la situación general en la que vive la población para que tenga una mejor resiliencia ante cualquier adversidad.

Referencias

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] U. de California en San Francisco. La respuesta de México al Covid-19: Estudio de caso. `https://globalhealthsciences.ucsf.edu/sites/globalhealthsciences.ucsf.edu/files/la_respuesta_de_mexico_al_covid_esp.pdf`.

[3] G. J. V.-L. M. G. González-Pérez, S. Romero-Valle, A. Vega-López, and C. E. Cabrera-Pivaral. Exclusión social e inequidad en salud en méxico: Un análisis socio–espacial. *Revista de salud pública*, 10(1), 2008.

[4] INEGI. Principales resultados por localidad (ITER). Censo de Población y Vivienda 2020. https://www.inegi.org.mx/programas/ccpv/2020/default.html#Datos_abiertos, 2021.

[5] D. Montgomery, E. Peck, and G. Vining. *Introduction to linear regression analysis*. Wiley, 2012.

[6] A. M. P. *Understanding Regression Analysis*. Plenum Press, 1997.

[7] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022.

[8] Secretaría de Salud, Gobierno de México. Datos Abiertos Dirección General de Epidemiología. https://www.gob.mx/salud/documentos/datos-abiertos-152127, 2020.

[9] S. V, S. Q. M, O. R. S, and R. D. J. E. Epidemiología de covid-19 en méxico: del 27 de febrero al 30 de abril de 2020. *Rev Clin Esp*, 220(8):463–471, 2020.

[10] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0.

[11] S. Wesberg. *Applied Linear Regression*. Wiley, 4 edition, 1980.

Información General

¿Quieres publicar artículos, información sobre eventos o noticias en el boletín?

La Sociedad Mexicana de Computación Científica y sus Aplicaciones A. C. (SMCCA), convoca a toda la comunidad interesada en el área de la Computación Científica y sus Aplicaciones, a presentar noticias, información sobre eventos, artículos de divulgación e investigación de alta calidad en el era, así como reportes de trabajos de tesis de nivel licenciatura y posgrado en Matemáticas Aplicadas.

Requisitos para la colaboración en el Boletín

I Artículos de Divulgación e Investigación.

a) Los artículos que se envíen para ser publicados deberán ser inéditos y no haber sido ni ser sometidos simultáneamente a la consideración en otras publicaciones.

b) Todos los artículos son sometidos a una revisión por expertos en estas áreas de instituciones nacionales e internacionales.

c) Los artículos a presentarse deben de ser enviados por medio de la página del Boletín: https://www.scipedia.com/sj/smcca

d) En la página de la sociedad se puede encontrar la plantilla de LaTeX para la correcta escritura de artículos.

II Información sobre eventos.

a) Los eventos cuya información quiera ser publicada para promocionarlos, deberán estar relacionados con el área de las Matemáticas Aplicadas y la Computación Científica.

b) La información debe enviarse en un archivo de imagen: PDF, JPG, PNG.

c) La información no deberá exceder una cuartilla.

d) Enviar la información con al menos 6 meses de anticipación a la fecha en que se llevaría a cabo.

III Noticias.

a) Las noticias a ser publicadas en el Boletín deben ser noticias relevantes de actividades de la SMCCA, Socios, Comunidad Científica interesada en las Matemáticas y Computación Científica.

b) La información de las noticias debe enviarse en un archivo de imagen: PDF, JPG, PNG.

c) La información no deberá exceder una cuartilla.

El material de colaboración, noticias e información de eventos, deberán ser dirigidos al Dr. Gerardo Tinoco Guerrero al correo electrónico de la SMCCA: smcca@smcca.org.mx.

Todos los artículos son sometidos a evaluación por especialistas de instituciones nacionales e internacionales y su publicación estará sujeta a la disponibilidad de espacio en cada número. Las demás colaboraciones se someterán a corrección de estilo y su publicación estará sujeta a la disponibilidad de espacio en cada número. Sólo se aceptará el material enviado que cumpla con todos los requisitos anteriormente señalados.

El envío de cualquier colaboración al Boletín implica no solo la aceptación de lo establecido en este documento, sino también la autorización al Comité Editorial del Boletín SMCCA para incluirlo en su página electrónica, reimpresiones, colecciones y cualquier otro medio que permita lograr una mayor y mejor difusión.

Sociedad Mexicana de Computación Científica y sus Aplicaciones

La Sociedad Mexicana de Computación Científica y sus Aplicaciones fue fundada el 16 de Mayo de 2013, para realizar actividades de investigación científica o tecnológica inscritas en el RENIECyT (Registro Nacional de Instituciones y Empresas Científicas y Tecnológicas), prestadas únicamente a los socios y asociados. Es una Asociación sin fines de lucro. Entre sus tareas fundamentales destacan: Conjuntar acciones e intereses comunes en los investigadores, profesores y estudiantes interesados en la Computación Científica y sus Aplicaciones, con el fin de fomentar la investigación de calidad, promover la actualización y el perfeccionamiento para el desarrollo científico, tecnológico y social; promover la creación, organización, acumulación y difusión de conocimientos referidos a la Computación Científica y sus Aplicaciones; promover la formación e interacción de redes y grupos de trabajo orientados hacia el desarrollo disciplinar, interdisciplinar y temático de la investigación; fomentar el desarrollo de la investigación sobre la Computación Científica y sus Aplicaciones en la República Mexicana; contribuir al mejoramiento de la enseñanza de la Computación Científica y sus Aplicaciones en la República Mexicana; promover y organizar toda clase de encuentros y eventos académicos orientados a la comunicación y discusión entre investigadores y profesores, así como también a la difusión del conocimiento hacia sectores interesados en la integración de la Computación Científica y sus Aplicaciones en los problemas de su sector.

smcca@smcca.org.mx
http://www.smcca.org.mx
https://www.scipedia.com/sj/smcca