



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

ALGORITMO DE OPTIMIZACIÓN POR ENJAMBRE
DE PARTÍCULAS EN COSMOLOGÍA
OBSERVACIONAL

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

FÍSICO

P R E S E N T A :

DANIEL MORALES HERNÁNDEZ

TUTOR

DR. JOSÉ ALBERTO VÁZQUEZ GONZÁLEZ



CIUDAD UNIVERSITARIA, CDMX, 2024



**CONSTANCIA DE
PRESENTACIÓN DE
EXAMEN PROFESIONAL**

En la Universidad Nacional Autónoma de México, en el AULA SOTERO PRIETO I de la FACULTAD DE CIENCIAS siendo las 11:00 horas del día 23 de septiembre de 2024, el alumno:

DANIEL MORALES HERNANDEZ

de nacionalidad MEXICANA con número de cuenta 315125669 se presentó con el fin de sustentar el examen oral para obtener el título de:

FÍSICO

en su modalidad de titulación por TESIS con el trabajo titulado: ALGORITMO DE OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULAS EN COSMOLOGÍA OBSERVACIONAL. El alumno cursó sus estudios en el periodo 2018-1 a 2023-2 habiendo obtenido un promedio de 8.60 y cumpliendo con los requisitos académicos señalados en el plan de estudios 1081 aprobado por el H. Consejo Universitario.

El Jurado designado por el Comité Académico integrado por:

PRESIDENTE: DR. SERGIO MENDOZA RAMOS
VOCAL: DR. MARCOS ALEJANDRO GARCIA GARCIA
SECRETARIO: DR. JOSE ALBERTO VAZQUEZ GONZALEZ
SUPLENTE: DR. FRANCISCO NETTEL RUEDA
SUPLENTE: DR. JORGE LUIS CERVANTES COTA

Tras el interrogatorio y deliberación resolvió otorgarle la calificación de:

Aprobado

Procediendo a informarle el resultado, tomarle la Protesta Universitaria y dar por concluido el acto.



PRESIDENTE DEL JURADO



SECRETARIO DEL JURADO



VOCAL DEL JURADO

El Titular de la entidad académica hace constar que las firmas electrónicas que anteceden son válidas y corresponden a los miembros del jurado.



**"POR MI RAZA HABLARÁ EL ESPÍRITU"
DR. VÍCTOR MANUEL VELÁZQUEZ AGUILAR
DIRECTOR DE LA ENTIDAD ACADÉMICA**

QR de verificación



Cadena de verificación digital

77646ffc006a8570f08786f9c0ca86dec6e78c538d987fb404f0ea485c61caa7f76003e0848d93b27cccf724499700bfb7c4782ce8b94028f27b88f8bb7102

1. Datos del alumno

Morales
Hernández
Daniel
55 18 61 73 55
Universidad Nacional Autónoma de México
Facultad de Ciencias
Física
315125569

2. Datos del tutor

Dr.
José Alberto
Vázquez
González

3. Datos del sinodal 1

Dr.
Sergio
Mendoza
Ramos

4. Datos del sinodal 2

Dr.
Marcos Alejandro
García
García

5. Datos del sinodal 3

Dr.
Francisco
Nettel
Rueda

3. Datos del sinodal 4

Dr.
Jorge Luis
Cervantes
Cota

7. Datos del trabajo escrito

Algoritmo de Optimización por Enjambre de Partículas en Cosmología Ob-
servacional
130 p
2024

*A mis padres, Gabriela y Roberto.
Gracias por todo su apoyo y cariño.*

Agradecimientos

Quisiera agradecer a mi asesor, el Dr. José Alberto Vázquez González por su mentoría durante la elaboración de este escrito, por compartir sus valiosos consejos y conocimientos que me han servido para mi desarrollo profesional. También agradecerle por impulsarme en la participación de congresos y otros eventos que siempre resultan en experiencias enriquecedoras.

También agradecer a mi comité sinodal por su apoyo y tiempo en la revisión de este trabajo. Particularmente agradecer al Dr. Francisco Nettel Rueda por también darme la oportunidad de apoyar en el curso de Relatividad.

Agradezco de todo mi corazón a mi madre Gabriela y a mi padre Roberto, por su apoyo, paciencia y buenos deseos durante esta etapa de mi vida como estudiante de física, que no fue fácil, pero que ustedes siempre creyeron en mí y sabían como motivarme a seguir adelante. Gracias por acompañarme en este camino. A mi familia por todos los momentos de diversión y porque su compañía siempre me ha hecho sentir comfortable.

A Aline, mi novia, por todo su cariño y compañía que siempre me hicieron sentir mejor en momentos difíciles, y porque también aprendí mucho de ella durante estos años. Sin duda eres una persona con un gran corazón y con una mente excepcional.

A mis amigos, por hacer de mi estancia en la facultad mucho más amena y por haberme compartido sus conocimientos y experiencias.

Investigación realizada gracias al Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica (PAPIIT) de la UNAM IN117723. Agradezco a la DGAPA-UNAM la beca recibida.

Resumen

El propósito principal de este trabajo es mostrar el uso del algoritmo de optimización por enjambre de partículas como un método eficiente para la estimación de parámetros cosmológicos siguiendo el marco de la estadística bayesiana. Para ello, es necesario incursionar en varios campos de la ciencia, como la cosmología, la estadística bayesiana y los algoritmos de optimización, en particular los algoritmos de enjambre.

En principio, estas ramas de la ciencia parecerían no tener una conexión entre sí, sin embargo, el desarrollo de la cosmología depende fuertemente de las observaciones realizadas, y a su vez, los datos obtenidos deben ser analizados siguiendo métodos estadísticos y computacionales.

Se han desarrollado diversos modelos cosmológicos que intentan explicar la expansión acelerada del universo, cada uno de estos modelos constan de un determinado número de parámetros que en conjunto podrían explicar el problema de la expansión acelerada y a su vez, acoplarse a los datos observados. Nos enfocamos en 3 modelos cosmológicos, tales como el modelo Λ CDM, ω CDM y el modelo Chevallier-Polarski-Linder (CPL). Estos modelos pueden expresarse por medio de la ecuación de Friedmann, de donde se obtiene la información teórica necesaria acerca de los parámetros cosmológicos. Para los fines de este trabajo consideramos que el modelo Λ CDM consta de dos parámetros a estimar, como el parámetro de Hubble en la actualidad H_0 y el parámetro de densidad de materia actual $\Omega_{m,0}$, mientras que el modelo ω CDM consta de un parámetro extra ω_0 . Finalmente, el modelo CPL consta de 4 parámetros: H_0 , $\Omega_{m,0}$, ω_0 y ω_a .

La estimación de estos parámetros se llevó a cabo minimizando la función χ^2 , ya que, según la estadística bayesiana el valor mínimo de χ^2 indica qué tan bien se ajusta un modelo a los datos obtenidos. Los datos utilizados en este trabajo para llevar a cabo el test chi-cuadrada consisten de los Cronómetros Cósmicos (CC) y Supernovas tipo Ia (SNIa), aunque mencionamos que los datos de Oscilaciones Acústicas Bariónicas suelen emplearse en los trabajos de estimación de parámetros cosmológicos ya que funcionan como reglas

estándar.

Debido a que cada modelo consta de diferente número de parámetros, utilizamos un método que pueda optimizar funciones de alta dimensionalidad y de características complejas que dificulten el proceso de minimización. Este método consiste en la implementación del algoritmo de optimización por enjambre de partículas, que es un algoritmo de optimización estocástico inspirado en el comportamiento de aves durante la búsqueda de alimento. Este algoritmo tiene la capacidad de hallar con gran precisión los valores óptimos de funciones que constan de varios óptimos (globales o locales) sin requerir de mucho coste computacional como los algoritmos de optimización deterministas.

Para la ejecución de este algoritmo se utilizó una librería de Python llamada PySwarms, la cual es fácil de implementar y modificar a nuestras necesidades. Con esta herramienta obtuvimos el mínimo de la función fitness (en este caso la función χ^2) y los valores donde se halla este mínimo, es decir, el valor de cada parámetro del modelo. El modelo cuyo valor χ^2 fue menor, considerando datos individuales de Cronómetros Cósmicos y supernovas fue el modelo CPL. Para los datos compuestos CC+SNIa, el valor χ^2 mínimo también se obtuvo con el modelo CPL.

Finalmente, se utilizaron los criterios de información Akaike (AIC) y Bayesiano (BIC) para hacer una comparación de modelos. Mientras menor sea el valor de estos criterios, menor penalización tendrá según el número de parámetros que contiene. De acuerdo a estos criterios, se obtuvo que el mejor modelo, considerando los datos de Cronómetros Cósmicos y Supernovas fue el modelo Λ CDM. Para los datos combinados, también obtuvimos que el modelo Λ CDM tiene los valores más bajos de los criterios AIC y BIC.

Índice general

Agradecimientos	V
Resumen	VII
1. Introducción	1
2. Cosmología observacional	5
2.1. Breve revisión histórica	5
2.1.1. Inicio de la cosmología moderna	7
2.2. Universo en expansión	8
2.2.1. Ley de Hubble	8
2.2.2. Principio cosmológico	9
2.3. Métrica FLRW y ecuaciones de Friedmann	10
2.4. Parámetros Cosmológicos	16
2.4.1. Materia Oscura	20
2.5. Modelos Cosmológicos	21
2.5.1. Modelo Λ CDM	22
2.5.2. Modelo ω CDM	22
2.5.3. Modelo Chevallier-Polarski-Linder (CPL)	23
2.6. Estimación de parámetros cosmológicos	24
2.6.1. Estadística Bayesiana	24
2.7. Criterios de información	29
2.8. Observaciones y datos	30
2.8.1. Cronómetros cósmicos	30
2.8.2. Supernovas	31
2.8.3. Otros datos: Oscilaciones Acústicas de Bariones	33
3. Optimización	37
3.1. Análisis de los problemas de optimización	37
3.2. Métodos de optimización	39

3.2.1.	Optimización determinista y optimización estocástica .	40
3.2.2.	Métodos heurísticos y meta-heurísticos	41
3.2.3.	Ejemplos de optimización determinista	42
4.	Algoritmos bio-inspirados	47
4.1.	Algoritmos evolutivos	48
4.1.1.	Ejemplo: algoritmo genético	49
4.2.	Algoritmos de enjambre	50
4.2.1.	Optimización por colonia de hormigas	52
5.	Optimización por enjambre de partículas	55
5.1.	Descripción del algoritmo PSO	56
5.1.1.	Dinámica de las partículas	56
5.1.2.	Topología de vecindades	59
5.1.3.	Búsqueda global y local	62
5.1.4.	¿Global best o Local best?	65
5.2.	Implementación de PSO en Python	65
5.2.1.	Funciones test	66
5.2.2.	Datos sintéticos para una recta	74
6.	Resultados	81
6.0.1.	Datos de cronómetros Cósmicos	81
6.0.2.	Datos de Supernovas	89
6.0.3.	Datos conjuntos CC+SN Ia	96
6.0.4.	Selección de modelos	98
7.	Conclusiones	101
Apéndice A.	Ecuaciones de campo de Albert Einstein	103
Apéndice B.	Distancias	109
B.0.1.	Distancia propia y comóvil	109
B.0.2.	Distancia de luz	110
B.0.3.	Distancia transversal y de ángulo	112
Apéndice C.	Criterio para la elección de parámetros PSO	115
Apéndice D.	Resultados del promedio y desviación estándar	117
Bibliografía		123

1 Introducción

A partir de las observaciones de Vesto Slipher y Edwin Hubble se descubrió que el universo se expande, sin embargo, tuvieron que pasar décadas de desarrollo tecnológico para que, a finales de la década de 1990, basándose en los datos recolectados de la luminosidad de supernovas tipo Ia, se descubriera que el universo se expande aceleradamente ([1] y [2]). Otras observaciones que reforzaron este descubrimiento fue la detección y análisis de las anisotropías del fondo cósmico de microondas (CMB) y las oscilaciones acústicas de bariones (BAO) [3]. Este importante descubrimiento fue posible debido a la gran cantidad de datos obtenidos de diversos telescopios y satélites, particularmente de los satélites Wilkinson Microwave Anisotropy Probe (WMAP), el satélite Planck y el telescopio espacial Hubble. Es así que la cosmología moderna debe afrontar dos problemas. El primero, es explicar la causa de la expansión acelerada del universo, para lo cual se han propuesto varios modelos cosmológicos que se pueden dividir en 2 tipos: materia modificada, que implica alterar y/o considerar nuevas formas de densidad de energía, como la quinta-esencia, el otro tipo es la gravedad modificada, que implica alterar la relatividad general, como las teorías $f(R)$ [4]. El segundo, es recolectar e interpretar una gran cantidad de datos por medio de métodos estadísticos para ajustar los parámetros cosmológicos de cada modelo a los observables y establecer algún criterio de validación entre los modelos cosmológicos y la información obtenida de los datos.

El modelo cosmológico estándar, también conocido como modelo Λ CDM, cuyas siglas se refieren a las principales componentes del universo: Λ (energía oscura) y CDM (materia oscura fría), este modelo considera que la energía oscura es la responsable de la expansión acelerada del universo, donde la constante cosmológica tiene presión negativa [5], y cuya ecuación de estado es $p = -\rho$, donde ρ es la densidad de energía. Aunque este modelo se ha ajustado bien a varias observaciones cosmológicas como los datos de supernovas, la edad del universo y el fondo cósmico de microondas [6], también presenta ciertos problemas. El primero de ellos se conoce como *el problema*

de la constante cosmológica, este se refiere a que el valor de la constante cosmológica Λ estimado por los físicos de partículas es varios órdenes de magnitud más grande que el observado [7]. El otro problema se refiere al *problema de coincidencia*, que refiere a que la densidad de energía oscura Ω_Λ y la densidad de materia Ω_m tienen, en la actualidad, el mismo orden de magnitud sin ninguna razón física aparente [4]. El modelo estándar también parece tener problemas a pequeñas escalas ($\sim$ kpc). Algunos de estos problemas son la discrepancia que hay entre el número de galaxias enanas observadas, con las predichas por simulaciones cosmológicas. Otro problema es el de núcleo cúspide, donde existe una diferencia entre las predicciones de Λ CDM respecto al pico de los perfiles de densidad de materia oscura y las observaciones de galaxias enanas que indican perfiles menos profundos [8].

Una manera de lidiar con estos problemas está en proponer modelos alternativos al modelo estándar. Existe una gran variedad de estos modelos, que se pueden dividir en modelos de quinta-esencia que consideran campos escalares, modelos de gravedad modificada, el modelo de gas de Chaplygin, entre otros. Tampoco pueden excluirse los modelos de energía oscura dinámica, donde el parámetro de la ecuación de estado ω varía con el tiempo y no es constante como se propone en el modelo estándar cosmológico. En este trabajo nos enfocamos en restringir el parámetro ω en los modelos ω CDM y el modelo Chevallier-Polarski-Linder (CPL), que considera a ω como una función lineal [4].

El modelo Λ CDM y los modelos ω CDM y CPL constan de distintos parámetros cosmológicos que deben estimarse a partir de los datos observacionales, la manera usual de hacerlo es por medio del enfoque bayesiano [9], ya que por el teorema de Bayes, puede asociarse una distribución de probabilidad posterior (es decir, la probabilidad de obtener los parámetros dado un conjunto de datos) con la función de verosimilitud, la cual se encarga de medir la probabilidad de hallar una medición u observación cuando se tiene información acerca de algún parámetro del modelo.

Una forma de estimar la máxima verosimilitud es por medio del test χ^2 , que se expresa en la ecuación (2.44), la cual indica qué tanto se ajusta un modelo a un conjunto de datos. Es así que la tarea de estimación de parámetros se vuelve en un problema de optimización [10]. Un método numérico común para la estimación de parámetros es el método de Monte Carlo basado en cadenas de Markov (MCMC), que consiste en construir una secuencia de puntos llamada *cadena* para evaluar la función de verosimilitud [11], también existen otros métodos para estimar parámetros cosmológicos basados en técnicas de inteligencia artificial, como redes neuronales [12] y computación evolutiva [13].

Actualmente se trabaja con una gran cantidad de datos en diferentes áreas de la ciencia y de la industria, es por ello que se han desarrollado distintos algoritmos para analizar y procesar grandes cantidades de datos, entre ellos destacan los algoritmos bio-inspirados. El propósito de los algoritmos bio-inspirados es resolver problemas específicos con base en el comportamiento de diversos fenómenos presentes en la naturaleza [14]. En este trabajo se pretende contribuir con los métodos de estimación de parámetros desde el enfoque de la optimización matemática utilizando un algoritmo bio-inspirado llamado *algoritmo de optimización por enjambre de partículas*. La razón principal de utilizar este algoritmo es que es fácil de implementar y puede hallar mínimos locales o globales en funciones que constan de muchos parámetros, a un bajo coste computacional [10]. Se espera entonces, que este algoritmo pueda estimar los parámetros cosmológicos de los diversos modelos mencionados anteriormente con relativa rapidez y considerando conjuntos de datos individuales y compuestos.

Para la elección del modelo cosmológico que mejor se haya ajustado a los datos se deben satisfacer algunos modelos de selección estadística, como el criterio de información Akaike (AIC) y Bayesiano (BIC), los cuales, con el fin de evitar un sobre ajuste, tienden a favorecer modelos que se hayan ajustado bien y que no consistan de un gran número de parámetros [4].

La estructura de este trabajo es la siguiente: en el capítulo 2 hacemos una breve revisión histórica de la cosmología y su evolución hasta nuestros días, pasando por la ecuaciones de campo de Einstein y las observaciones de Edwin Hubble acerca de la expansión del universo. También mencionamos las implicaciones del principio cosmológico, que junto con la teoría de la relatividad general dan lugar a la métrica Friedmann-Lemaître-Robertson-Walker y las ecuaciones de Friedmann. Revisamos brevemente 3 modelos cosmológicos (Λ CDM, ω CDM y CPL), y los parámetros que los componen, así como el análisis bayesiano utilizado para la estimación de parámetros y los datos observacionales tratados en este trabajo. Mencionamos también los criterios de información Akaike y Bayesiano.

En el capítulo 3 se explica en qué consisten los problemas de optimización y los distintos métodos que hay para resolver estos problemas, destacando los métodos deterministas y estocásticos. Además, se dan un par de ejemplos de algoritmos deterministas. En el capítulo 4 nos enfocamos en los algoritmos bio-inspirados y sus dos ramas principales: los algoritmos evolutivos y de enjambre, dando un ejemplo para cada uno. En el capítulo 5 hablamos concretamente de uno de los algoritmos de enjambre más representativos, a saber, el algoritmo de optimización por enjambre de partículas (PSO) y explicamos las características que debe tener este algoritmo para llevar a

cabo una tarea de optimización eficaz. Damos varios ejemplos de funciones test de una y 2 dimensiones que fueron optimizadas con el algoritmo PSO utilizando la librería PySwarms de Python. En el capítulo 6 mostramos los resultados obtenidos de la optimización con PSO para los diferentes conjuntos de datos y los resultados de la comparación de modelos por medio de los criterios Akaike y Bayesiano. Finalmente, en el capítulo 7 escribimos las conclusiones de este trabajo, comparando los resultados propios utilizando PSO con los de otro trabajo, donde utilizaron el algoritmo genético para los mismos propósitos.

2 Cosmología observacional

Desde que el ser humano tiene razón de su existencia y comenzó a contemplar el cielo durante las noches, fue inevitable tener curiosidad por saber qué son esas luces brillantes en un cielo oscuro. Mirar al Sol, la Luna y a las estrellas dio origen a innumerables mitologías e historias que a su vez trataban de explicar el origen de todo lo que uno pudiera cuestionarse.

Si bien las primeras culturas y civilizaciones no tenían idea del lugar que ocupa la Tierra en el universo, ni de la lejanía que hay entre las estrellas y la Tierra, su aportación, por el simple hecho de buscar una respuesta a lo que observaban es invaluable. Pronto, mirar al cielo no solo inspiraba historias, sino que empezó a ser de gran utilidad para las técnicas de navegación y exploración, por lo que el estudio del cielo empezó a tener relevancia, seriedad y con ello, progreso. Es así, que comenzaron a surgir preguntas acerca del origen del universo, su forma y el lugar que ocupamos en él.

Por lo tanto, es importante hacer una breve revisión histórica acerca de las ideas que contribuyeron al desarrollo de la cosmología moderna hasta nuestros días. Así como mencionar algunos modelos cosmológicos que pretenden explicar la dinámica y evolución del universo.

Desde el punto de vista observacional, vale la pena mencionar los conjuntos de datos que usaremos en este trabajo y los métodos estadísticos que se utilizarán para hallar los parámetros cosmológicos que mejor se ajusten a ellos, y también los criterios empleados para hallar no solo los mejores parámetros, sino el mejor modelo cosmológico que explique los datos observados.

2.1. Breve revisión histórica

Actualmente la cosmología es una rama de la física que estudia el universo a grandes escalas, considerándolo como un todo, sin embargo, el intento de estudiar el universo y el lugar que ocupamos en él tiene varios siglos de desarrollo y se puede dividir en dos categorías: la teoría geocéntrica postulada por Aristóteles y siendo refinada por Ptolomeo para explicar el movimiento

de algunos cuerpos celestes, y la teoría heliocéntrica postulada por Nicolás Copérnico.

En la teoría heliocéntrica de Ptolomeo se proponía que la Tierra es el centro del universo y que los planetas giraban alrededor de ella en círculos y, a su vez, estos planetas giraban en círculos a lo largo de su trayectoria. Por otro lado, en el modelo heliocéntrico de Copérnico publicado en el año 1543, se consideraba que el Sol es estático y que la Tierra, así como los otros planetas conocidos en aquella época giraban alrededor de él.

Años después, con las observaciones de Júpiter y sus satélites hechas por Galileo se empieza a descartar la teoría geocéntrica de Ptolomeo que aún era fuertemente defendida por la iglesia, ya que pone en evidencia que otros cuerpos celestes pueden girar alrededor de otros planetas diferentes a la Tierra [15].

Fue en el año 1687 que Isaac Newton publica una de las obras más fundamentales en la historia de la ciencia, *Philosophiae Naturalis Principia Mathematica* en la que se da una explicación matemáticamente elegante para el movimiento de los planetas y en general, de los cuerpos celestes. Así, la razón de que Galileo observó a los pequeños satélites de Júpiter girando a su alrededor fue por efecto de la **Ley de gravitación universal** formulada como:

$$\vec{F} = -G \frac{m_1 \cdot m_2}{r^2} \vec{u}_r. \quad (2.1)$$

La fórmula anterior establece que la fuerza de atracción $\vec{F}$ entre 2 cuerpos, depende únicamente de sus respectivas masas m_1 y m_2 , y es inversamente proporcional al cuadrado de la distancia que los separa (r^2). El signo menos indica el carácter atractivo de la fuerza, cuya dirección es la misma que la del vector unitario $\vec{u}_r$ pero de sentido opuesto. La relación se convierte en igualdad gracias a la constante gravitacional G . Considerando esta aportación de Newton y las aportaciones, tanto de Hipatia como de Kepler acerca de las órbitas elípticas que hacen los planetas al girar alrededor del Sol, se concluyó que la gravedad provoca el giro elíptico de la Luna alrededor de la Tierra y de los planetas alrededor del Sol.

La mecánica Newtoniana fue un éxito para el conocimiento humano ya que con las ecuaciones de movimiento -derivadas de las 3 Leyes de Newton- se puede predecir el movimiento de los planetas y en general de cualquier objeto de dimensiones macroscópicas. Sin embargo, en el universo de Newton el tiempo es absoluto, esto quiere decir que el tiempo fluye de manera continua en un espacio que para él era infinito e inmóvil [16]. Esta idea de tiempo y espacio fue, por varios siglos, bien recibida por la comunidad científica, ya

que además de intuitiva, funcionaba para resolver diversos problemas por medio de la mecánica Newtoniana.

2.1.1. Inicio de la cosmología moderna

La teoría electromagnética de James Clerk Maxwell y los experimentos de Michelson y Morley, llevaron a que Albert Einstein propusiera en 1905 su teoría de la *relatividad especial*, en la que concluye que el tiempo no es absoluto sino que las medidas de tiempo y espacio dependen del observador.

Una década después de publicada esta teoría, Einstein publicó su más célebre obra, *The foundation of the general theory of relativity*, donde se considera al espacio y el tiempo como un solo concepto: espacio-tiempo. Además, la teoría general de la relatividad da una descripción gravitacional diferente a la de Newton, donde la gravedad no es típicamente una fuerza, más bien es una estructura del propio espacio-tiempo que se puede deformar por la presencia de materia o energía.

Ecuación de campo de Einstein.

La deformación del espacio-tiempo da como resultado la interacción gravitacional que ejercen los cuerpos entre sí. Esta relación entre gravedad y materia se refleja en una de las ecuaciones más importantes de la física, cuya deducción puede verse en el apéndice A. La ecuación de campo fue originalmente expresada como:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu}, \quad (2.2)$$

en el lado izquierdo de la ecuación (2.2) se encuentra el tensor de Ricci ($R_{\mu\nu}$), y el escalar de curvatura de Ricci (R) junto con el tensor métrico $g_{\mu\nu}$. Es decir, el lado izquierdo describe la geometría del espacio-tiempo. Por otro lado, en el lado derecho se halla el tensor de energía-momento $T_{\mu\nu}$, la constante gravitacional G y la velocidad de la luz c , que en conjunto representan la presencia de materia y energía en el espacio-tiempo. En realidad, las ecuaciones de campo están conformadas por 10 ecuaciones diferenciales no lineales y acopladas [17], cuyas soluciones describen la distribución de materia en una región del espacio, acarreada por la curvatura local de este. Este resultado nos indica que el universo es dinámico, sin embargo, se creía que era estático, por lo que en 1917 Einstein incluyó una constante, conocida como la constante Λ . Esta constante actuaría como un efecto repulsivo que compense y evite la tendencia gravitacional del colapso, obteniendo así un

universo estático [18]. Por lo tanto, la ecuación de campo quedó finalmente expresada como:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu}. \quad (2.3)$$

2.2. Universo en expansión

2.2.1. Ley de Hubble

A partir de sus observaciones, el astrónomo Vesto Melvin Slipher descubrió la velocidad radial de la nebulosa de Andrómeda utilizando un espectroscopio, donde se reveló un alto corrimiento al rojo (*redshift*) de las líneas de absorción en su espectro [19].

Una forma de medir el corrimiento al rojo de un objeto es considerando que la luz que un cuerpo emite a un tiempo t tiene una longitud de onda λ , conforme t aumenta, la longitud de onda va aumentar también según las observaciones. Esta razón de cambio se puede expresar como:

$$z = \frac{\Delta\lambda}{\lambda} = \frac{\lambda_{obs} - \lambda_{em}}{\lambda_{em}}, \quad (2.4)$$

donde z es el redshift, λ_{em} es la longitud de onda de la luz emitida por la galaxia y λ_{obs} es la longitud de onda de esa misma luz emitida pero medida en la Tierra [5]. Cuando una galaxia se aleja de nosotros, las líneas de absorción del espectro tienen un mayor corrimiento al rojo o un alto redshift, por el contrario, cuando las galaxias se acercan a nosotros, las líneas de absorción tienen un mayor corrimiento al azul.

Posteriormente, las observaciones del astrónomo Edwin Hubble acerca del corrimiento al rojo de las galaxias lo llevaron a establecer en el año 1929 una relación entre las velocidades radiales de las galaxias (usando el redshift) y la distancia de separación que hay entre ellas [20].

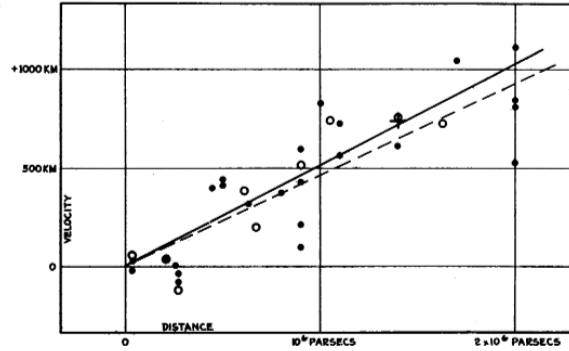


Figura 2.1: Relación de velocidades y distancias de nebulosas extragalácticas obtenidas por Edwin Hubble. Obtenido de [20].

La gráfica 2.1 obtenida por Hubble, muestra que la relación entre la distancia de separación de las galaxias y la velocidad radial es lineal. Los puntos oscuros y la línea sólida representan los resultados usando galaxias individuales, mientras que los círculos y la línea punteada representan los resultados combinando las nebulosas en grupos. A partir de esa gráfica se establece la **Ley de Hubble**, expresada como:

$$cz = H_0 D, \quad (2.5)$$

donde H_0 es la constante de Hubble a tiempo actual t_0 , y su propósito es medir el ritmo con el que se expande el espacio, c es la velocidad de la luz y D es la distancia de separación entre las galaxias. Este descubrimiento llevó a que Albert Einstein quitara la constante Λ de sus ecuaciones.

Con el descubrimiento de que el universo se expande, el sacerdote y astrofísico George Lemaitre propone que el origen del universo proviene de una gran explosión de singularidad finita de energía; esta idea surge de que si el universo se ha estado expandiendo, entonces en algún momento el universo debía estar concentrado en un solo punto, dando lugar a la teoría del Big Bang. Observaciones posteriores a las de Hubble, donde se consideraron redshifts más altos, sugirieron que el universo a grandes escalas es homogéneo e isótropo. A esta conclusión se le conoce como “Principio Cosmológico”.

2.2.2. Principio cosmológico

El principio cosmológico asegura que a escalas particularmente grandes (aproximadamente 100 Mpc [21]), el universo es *homogéneo* e *isotrópico*. La homogeneidad del universo asegura que el universo luce igual en cualquier punto, mientras que la isotropía establece que el universo luce igual

en cualquier dirección [5], por lo que no existen ni puntos, ni direcciones privilegiados.

Debido a que el principio cosmológico se supone válido para “distancias muy grandes” se tiene por la teoría de la relatividad general que también se trata de “tiempos muy largos”, por lo que se define un tiempo t cuya evolución refleje la evolución del universo, a este tiempo se le conoce como *tiempo cósmico*.

2.3. Métrica FLRW y ecuaciones de Friedmann

El desarrollo teórico de esta sección está inspirado principalmente de [21], [22], [23] y [5]. La bibliografía complementaria se citará en el mismo enunciado donde sea requerida.

Como hemos visto, la cosmología moderna se sustenta de la relatividad general, de la expansión del universo y del principio cosmológico; esto dio lugar a buscar una métrica $g_{\mu\nu}$ que sea compatible con las ecuaciones de campo de Einstein, es decir, una métrica que describa la deformación del espacio tiempo en presencia de materia, además de considerar homogeneidad e isotropía espacial.

Los primeros modelos cosmológicos consideraron un universo no dinámico que vive en el espacio de Minkowski con un número infinito de partículas de prueba sin masa y volumen que salieron disparadas a todas direcciones en un evento único ε . En un sistema de referencia S , este evento ocurre en su origen a un tiempo $t = 0$, por lo que todas las partículas se moverán radialmente del origen a velocidades menores que la de la luz. Esto dio pie a considerar que en modelos dinámicos, la distribución de materia y energía tuviera simetría esférica.

Sin embargo, para tener modelos más reales, se debe tomar en cuenta un universo espacialmente homogéneo e isótropo (a esto se le conoce como condición de máxima simetría), que evolucione con el tiempo. Para ello también es importante considerar las coordenadas comóviles, ya que permiten que las partículas mantengan sus coordenadas espaciales fijas a medida que el espacio se expande.

Por lo tanto, una métrica que considere esto tendrá la siguiente forma:

$$ds^2 = c^2 dt^2 - dr^2 - R^2(t) d\sigma^2, \quad (2.6)$$

donde $R(t)$ es una función conocida como el *factor de escala* y σ una función espacial definida como:

$$d\sigma^2 = g_{ij}(x)dx^i dx^j,$$

con $i, j = 1, 2, 3$ y g_{ij} la métrica espacial tal que cumpla la condición de máxima simetría. En estas condiciones acerca de la simetría espacial, el tensor de Riemann se puede simplificar a una expresión que depende únicamente de la métrica Euclideana 3-dimensional:

$$R_{ijkl} = k (g_{ik}g_{jl} - g_{il}g_{jk}),$$

donde definimos k como una constante de curvatura. De aquí podemos hallar el tensor de Ricci y el escalar de Ricci, que son respectivamente $R_{jl} = 2kg_{jl}$ y $R = 6k$. Todo esto nos dice que la curvatura depende de la constante k y nos permite obtener una expresión general de la métrica para describir el universo, conocida como la métrica FLRW (Friendman-Lemaitre-Robertson-Walker) y se expresa como:

$$ds^2 = cdt^2 - a^2(t) \left(\frac{dr^2}{1 - kr^2} + r^2 d\theta^2 + r^2 \sin^2(\theta) d\phi^2 \right). \quad (2.7)$$

Las coordenadas θ y ϕ son coordenadas angulares y r la coordenada radial. El termino $a(t)$ es el factor de escala adimensional, mientras que k es el factor de curvatura, el cual puede tomar 3 valores: $-1, 0, +1$, correspondientes a geometrías abiertas, planas o cerradas respectivamente (ver imagen 2.2). Veremos más adelante que dichas geometrías dependen de la densidad de energía ρ que contenga el universo.

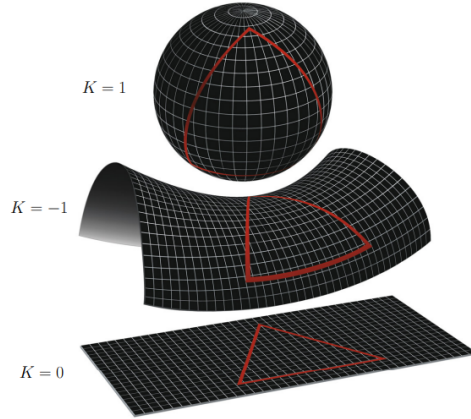


Figura 2.2: Geometrías espaciales del universo. De arriba hacia abajo: geometría cerrada, abierta y plana. Obtenido de [24].

El factor de escala $a(t)$ es una función que depende del tiempo y sirve como un indicador de qué tanto ha variado la distancia entre 2 puntos tomando 2 instantes de tiempo distintos. El factor de escala se relaciona con el redshift como:

$$1 + z = \frac{a_0}{a(t)}, \quad (2.8)$$

donde a_0 es el factor de escala en la actualidad ($t = 0$) y toma el valor de uno, mientras que en las primeras épocas del universo $a(t) < 1$. Algo importante a recalcar es que si uno estudia la evolución del factor de escala estaría estudiando directamente la evolución del universo.

Una vez introducido el factor de escala es posible definir 2 tipos de distancias: **la distancia comóvil** y **la distancia propia**, cuyas descripciones pueden verse en el Apéndice B.

Para resolver la ecuación (2.3) con la métrica FLRW es necesario tener una expresión del tensor de energía-momento $T_{\mu\nu}$, el cual describe el flujo de energía y momento lineal de una distribución continua de materia. Si consideramos que el flujo es producido por un fluido perfecto (es decir, un fluido que carece de viscosidad, homogéneo, no conduce calor y cuya velocidad es constante en el tiempo), se debe cumplir para un observador comóvil, que las componentes del tensor satisfacen $T_{00} = \rho$, $T_{0i} = 0$ y $T_{ij} = p\delta_{ij}$. Así, el tensor de energía momento se puede expresar en términos de la densidad de energía ρ , la presión del fluido p y la cuadrivelocidad u_μ :

$$T_{\mu\nu} = \left(\rho + \frac{p}{c^2}\right) u_\mu u_\nu - pg_{\mu\nu}.$$

Se suele decir que la ecuación anterior está expresada en coordenadas comóviles y esto es justo de la suposición de un universo homogéneo e isótropo [25]. Más adelante detallaremos brevemente acerca del significado de las expresiones ρ y p .

Dado que el fluido solo depende del tiempo, la única componente de la cuadrivelocidad es la componente temporal, $u_\mu = (1, 0, 0, 0)$ y recordando las condiciones de los elementos del tensor $T_{\mu\nu}$ se obtiene fácilmente que $T_{\mu\nu} = \text{diag}(\rho, p, p, p)$. Introduciendo esta forma del tensor de energía momento en la ecuación (2.3) obtenemos:

$$R_{\mu\mu} - \frac{1}{2}Rg_{\mu\mu} + \Lambda g_{\mu\mu} = 8\pi G\rho u_\mu u_\mu. \quad (2.9)$$

Al resolver las ecuaciones de Einstein con la métrica FLRW y sustituyendo las componentes del tensor métrico se obtienen las *ecuaciones de Friedmann*:

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G\rho}{3} - \frac{kc^2}{a^2} + \frac{\Lambda c^2}{3}, \quad (2.10)$$

$$\left(\frac{\ddot{a}}{a}\right) = \frac{\Lambda c^2}{3} - \frac{4\pi G}{3} (\rho + 3p/c^2). \quad (2.11)$$

La ecuación (2.10) muestra que el cambio en la expansión del universo depende de la densidad de energía ρ y que esta razón de cambio incrementa conforme el valor de ρ . Mientras que la ecuación (2.11) describe la aceleración con la que se expande el universo. Esta aceleración depende de la presión p y la densidad ρ , además debido al signo negativo, se entiende que la aceleración de la expansión disminuye cuando la densidad de materia y la presión incrementan.

Las ecuaciones de Friedmann describen cómo cambia el factor de escala $a(t)$ con base en $\rho(t)$, p y Λ . El término de densidad de energía ρ está asociado con el tipo de materia contenida en el universo, el término de presión p es justamente la presión de dicho contenido de materia y finalmente, la constante cosmológica Λ representa la energía oscura y está estrechamente relacionada con la expansión del universo debido a que corresponde a una presión negativa.

La evolución del universo depende de cómo ha cambiado la energía contenida en él, por lo que será útil buscar una relación explícita entre la densidad de energía y el factor de escala. Para encontrar esta relación usaremos la *ecuación de continuidad* (2.12), la cual describe qué tanto cambia la densidad de energía del universo conforme evoluciona $a(t)$. Para deducir esta ecuación debemos tomar la divergencia covariante del tensor energía-momento y asumir que dicha cantidad es cero, de lo contrario se interpretaría que hay un cambio de energía en el sistema.

$$\nabla_\mu T^{\mu\nu} = 0.$$

Dado que la componente $T^{00} = c^2\rho(t)$ es la que describe la energía, podemos expresar la conservación como $\nabla_\mu T^{\mu 0} = 0$ y fijarnos que la única componente distinta de cero es la componente T^{00} . Por lo que usando la definición de divergencia covariante se llega a la siguiente igualdad:

$$\begin{aligned} \nabla_\mu T^{\mu 0} &= \dot{\rho}(t) + 3\frac{\dot{a}(t)}{a(t)} \left(\rho(t) + \frac{p(t)}{c^2} \right), \\ &= 0. \end{aligned} \quad (2.12)$$

El lado derecho de la igualdad es la ecuación de continuidad y es válida para todas las formas de energía: radiación, materia y energía oscura.

Para conocer qué tipo de energía predominó en el universo durante un tiempo t se propone una ecuación de estado constante:

$$p(t) = \omega \rho(t). \quad (2.13)$$

Donde p es la presión de energía y ρ la densidad de energía. A ω se le conoce como *parámetro de la ecuación de estado* y contiene la información necesaria del tipo de energía con la que evoluciona el factor de escala. Para comprobar esto, sustituiremos la ecuación de estado en la ecuación de continuidad (2.12) y escogiendo $c = 1$ obtendremos el siguiente resultado:

$$\frac{d}{dt} \ln(\rho(t)) + 3(1 + \omega) \frac{d}{dt} \ln(a(t)) = 0.$$

Que al integrarlo resulta en:

$$\begin{aligned} \ln(\rho(t)) + 3(1 + \omega) \ln(a(t)) + cte &= 0, \\ \ln(\rho(t)) + \ln(a(t)^{3(1+\omega)}) + cte &= 0, \\ \rho(t) \cdot a(t)^{3(1+\omega)} \exp(cte) &= 1. \end{aligned}$$

De esta manera obtenemos la siguiente relación:

$$\rho(t) \propto a(t)^{-3(1+\omega)}. \quad (2.14)$$

La expresión anterior indica que a medida que el universo se expande, la densidad de energía se diluye a diferentes potencias del factor de escala y esto aplica para la densidad de energía de la materia, radiación y energía oscura [26].

A continuación mostraremos una lista de cómo se relaciona la densidad de energía para diferentes valores de ω con el factor de escala, es decir, cómo se diluyen los tipos de densidad de energía a diferentes potencias de $a(t)$:

- Para radiación: $\omega = 1/3 \rightarrow \rho \propto a^{-4}$.
- Para materia: $\omega = 0 \rightarrow \rho \propto a^{-3}$.
- Para energía de vacío: $\omega = -1 \rightarrow \rho \propto cte$.

Si usamos la ecuación (2.10) se puede estudiar la evolución del factor de escala. Para ello, sea $\rho_0 = \rho a^{3(1+\omega)}$ la solución a la ecuación (2.12), donde ρ_0

es la densidad de energía actual [17], se reescribe la ecuación de Friedmann en términos de la densidad actual como:

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3} \frac{\rho_0}{a^{3(1+\omega)}} - \frac{k}{a^2}. \quad (2.15)$$

Hemos omitido el término donde se encuentra la constante Λ ya que este no depende del factor de escala. La ecuación (2.15) nos ayuda a obtener las siguientes equivalencias:

$$\left(\frac{\dot{a}}{a}\right)^2 \propto \rho \propto \rho_0 a^{-3(1+\omega)}.$$

La equivalencia anterior nos conduce al siguiente resultado:

$$\frac{da}{dt} \propto a^{-\frac{1}{2}(1+3\omega)}.$$

Una vez más tenemos una relación que depende del parámetro ω por lo que si sustituimos los valores asociados a los tipos de energía y resolvemos la ecuación diferencial obtendremos los siguientes resultados:

- Para radiación: $\omega = 1/3 \rightarrow a(t) \propto t^{1/2}$.
- Para materia: $\omega = 0 \rightarrow a(t) \propto t^{2/3}$.
- Para energía del vacío: $\omega = -1 \rightarrow a(t) \propto e^t$.

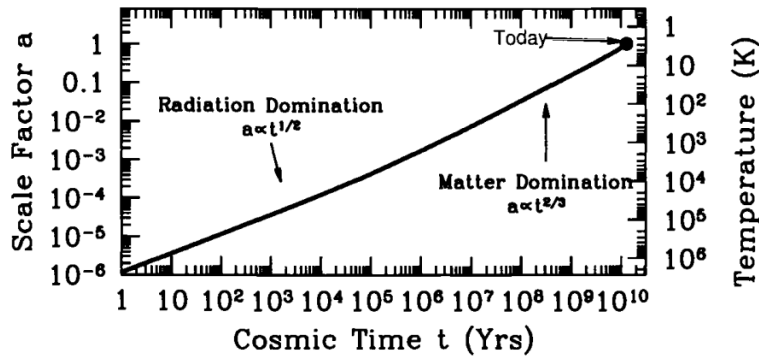


Figura 2.3: Evolución del factor de escala a lo largo del tiempo cósmico. Obtenido de [22].

La figura 2.3 muestra el contenido energético del universo a diferentes etapas del universo a lo largo del tiempo. Al inicio, el universo estaba dominado por radiación y el factor de escala aumentaba a razón de $t^{1/2}$, después la materia fue dominante y el factor de escala se relacionaba con el tiempo por el factor $t^{2/3}$.

2.4. Parámetros Cosmológicos

Una vez obtenido un número suficiente de ecuaciones y relaciones, queda una tarea aún más importante: *obtener cantidades que se puedan medir*. Para ello necesitamos definir parámetros en función de constantes conocidas o que por su propia definición se puedan medir con relativa sencillez.

Dentro de las ecuaciones de Friedmann se encuentra definido el primer parámetro a considerar, el cuál es el parámetro de Hubble:

$$H = \frac{\dot{a}}{a}.$$

Las mediciones de anisotropías del fondo cósmico de microondas (CMB) realizadas por el satélite Planck concluyeron que el valor actual del parámetro de Hubble es de $H_0 = (67.4 \pm 0.5) \text{ km s}^{-1} \text{ Mpc}^{-1}$ ([27]), mientras que las observaciones de estrellas cefeidas variables realizadas por el telescopio Hubble estimaron un valor de $H_0 = (72 \pm 8) \text{ km s}^{-1} \text{ Mpc}^{-1}$ ([28]). Sin embargo, también se han estimado valores intermedios ([29] y [30]), como en [31], donde:

$$H_0 = (70 \pm 2) \text{ km s}^{-1} \text{ Mpc}^{-1}. \quad (2.16)$$

Dado que la geometría del universo se determina por la densidad de energía que contiene, es conveniente definir un parámetro que relacione estos conceptos en función de un valor crítico. Con esto en mente definimos el *parámetro de densidad* Ω :

$$\Omega = \frac{\rho}{\rho_c}. \quad (2.17)$$

La densidad crítica ρ_c se obtiene al asumir que el universo es plano ($k = 0$) y sin considerar Λ [5], por lo que de la ecuación (2.10) se obtiene:

$$\rho_c = \frac{3H^2(t)}{8\pi G}. \quad (2.18)$$

Cuyo valor actualmente es $\rho_{c,0} = 1.88h^2 \times 10^{-26} \text{ kg m}^{-3}$, donde h es un número adimensional, cuyo valor es $H_0/100$. Se pueden obtener las geometrías

del universo en función del contenido de energía actual. Si $\Omega > 1$ (es decir, si la densidad actual es mayor que la densidad crítica) el universo tendrá la suficiente cantidad de materia para que la expansión se frene, el resultado de ello es un universo cerrado (una esfera). Si $\Omega < 1$ la densidad de materia no será lo suficientemente grande para detener la expansión por lo que el universo será abierto (hiperboloide). Finalmente, si $\Omega = 1$ la densidad de materia será similar a la densidad crítica, densidad para la cuál el universo es plano y se encuentra en expansión.

Se pueden reescribir las ecuaciones de Friedmann en términos de los parámetros de densidad, para ello hay que definir previamente el parámetro de densidad de la constante cosmológica [5] como:

$$\Omega_\Lambda = \frac{\Lambda}{3H^2}. \quad (2.19)$$

De esta manera:

$$\begin{aligned} H^2 &= \frac{8\pi G\rho}{3} - \frac{k}{a^2} + \frac{\Lambda}{3}, \\ &= \frac{8\pi G}{3}\Omega\rho_c - \frac{k}{a^2} + \frac{\Lambda}{3}, \\ &= \frac{8\pi G}{3}\Omega\left(\frac{3H^2}{8\pi G}\right) - \frac{k}{a^2} + \frac{\Lambda}{3}. \end{aligned}$$

Dividiendo entre H^2 :

$$\begin{aligned} 1 &= \Omega - \frac{k}{a^2H^2} + \frac{\Lambda}{3H^2}, \\ &= \Omega - \frac{k}{a^2H^2} + \Omega_\Lambda. \end{aligned}$$

Donde es posible definir un “parámetro de densidad” asociado a la curvatura:

$$\Omega_k = -\frac{k}{a^2H^2} \quad (2.20)$$

Por lo que la ecuación de Friedmann (2.10) queda reescrita en términos de los parámetros cosmológicos como:

$$\Omega_m + \Omega_k + \Omega_\Lambda = 1. \quad (2.21)$$

Donde Ω_m es el parámetro de densidad de materia. Los distintos parámetros de densidad se pueden expresar en términos de sus valores actuales a tiempo $t = 0$, y se suelen denotar como $\Omega_{m,0}$ para el parámetro de densidad actual, $\Omega_{k,0}$ para el parámetro de curvatura actual y $\Omega_{\Lambda,0}$ para el parámetro de densidad de energía oscura.

Suele considerarse que la densidad de energía total depende de la materia y de la radiación, es decir, $\rho = \rho_m + \rho_r$. Por lo que, análogamente al parámetro de densidad de materia se define el parámetro de densidad de radiación como:

$$\Omega_r = \frac{8\pi G \rho_r}{3H^2}. \quad (2.22)$$

Así, la expresión completa de la ecuación de Friedmann en términos de los parámetros de densidad es:

$$\Omega_m + \Omega_r + \Omega_k + \Omega_\Lambda = 1. \quad (2.23)$$

Esta expresión simplificada de la ecuación de Friedmann expresa la evolución del universo en términos de parámetros de densidad.

Ecuaciones de Friedmann en términos del corrimiento al rojo

Los parámetros de densidad expresados en la ecuación (2.23) nos dan información valiosa acerca de la distribución de contenido energético en el universo, por lo que es necesario conocer sus valores. Podemos medir estos parámetros en la edad actual del universo (sustituyendo $t = t_0$) y expresar la ecuación de Friedmann en estas condiciones.

Primero, notemos que a partir de la definición (2.17) podemos obtener $\rho_i = \Omega_i \rho_c$, con $i = \{m, r, k, \Lambda\}$. Con esto, podemos escribir la ecuación (2.10) como:

$$H^2(z) = \frac{8\pi G}{3} (\rho_r + \rho_m + \rho_k + \rho_\Lambda). \quad (2.24)$$

Sin embargo sabemos que existe una razón de proporcionalidad entre las densidades ρ para materia, radiación y el factor Λ , y de la ecuación (2.20) también podemos relacionar $\rho_k \propto a^{-2}(t)$. Diremos que la igualdad se va a obtener si:

$$\begin{aligned}
\rho_r(t) &= \rho_{r,0} a^{-4}. \\
\rho_m(t) &= \rho_{m,0} a^{-3}. \\
\rho_k(t) &= \rho_{k,0} a^{-2}. \\
\rho_\Lambda(t) &= \rho_{\Lambda,0}.
\end{aligned}$$

Por lo que podemos sustituir los parámetros de densidad ρ_i con $i = \{r, m, k, \Lambda\}$ utilizando las expresiones anteriores, y sustituyendo también la expresión de la densidad crítica actual $\rho_{c,0}$ obtenemos:

$$\begin{aligned}
H^2(z) &= \frac{H_0^2}{\rho_{c,0}} (\rho_{r,0} a^{-4} + \rho_{m,0} a^{-3} + \rho_{k,0} a^{-2} + \rho_{\Lambda,0}), \\
&= H_0^2 (\Omega_{r,0} a^{-4} + \Omega_{m,0} a^{-3} + \Omega_{k,0} a^{-2} + \Omega_{\Lambda,0}).
\end{aligned} \tag{2.25}$$

Finalmente, de la expresión (2.8), considerando $a_0 = 1$ obtendremos:

$$H^2(z) = H_0^2 \left(\Omega_{r,0} (1+z)^4 + \Omega_{m,0} (1+z)^3 + \Omega_{k,0} (1+z)^2 + \Omega_{\Lambda,0} \right). \tag{2.26}$$

De esta manera, hemos obtenido una expresión en términos de parámetros medibles, como la constante de Hubble H_0 , cuyo valor se mencionó anteriormente (ver 2.16).

Los resultados de estos parámetros se obtuvieron de las mediciones del espectro de anisotropías del fondo cósmico de microondas (CMB por sus siglas en inglés) obtenidas por el satélite Planck [27], cuyos resultados son:

$$\begin{aligned}
\Omega_{m,0} &= 0.315 \pm 0.007. \\
\Omega_{k,0} &= 0.001 \pm 0.002. \\
\Omega_{\Lambda,0} &= 0.684 \pm 0.007.
\end{aligned}$$

La densidad de radiación $\Omega_{r,0}$ se puede medir a partir de la radiación de fondo de microondas, el experimento COBE Far Infrared Absolute Spectrophotometer (FIRAS) obtuvo un valor para este parámetro de $\Omega_{r,0} h^2 = 2.47 \times 10^{-5}$ [32].

De estos resultados se puede observar que el parámetro de densidad de curvatura contribuye prácticamente con un 0% de la energía del universo, esto quiere decir que la curvatura espacial del universo es plana. Por otro

lado, el parámetro de densidad de materia contribuye con un 32 % de la energía del universo, sin embargo esta contribución se divide para dos tipos de materia: materia oscura no-relativista Ω_c y materia bariónica Ω_b , la cual contribuye con aproximadamente 5 % del contenido de energía mientras que la *materia oscura* contribuye con el 27 % del contenido energético. Finalmente, el resto de la densidad de energía actual en el universo está dada por la energía de vacío, cuya contribución es de aproximadamente 68 %.

La energía de vacío o energía oscura es un misterio al día de hoy, por lo que se han tenido que proponer diversos modelos cosmológicos que traten de explicar la expansión acelerada de universo. Algunos modelos cosmológicos, entre ellos, el modelo cosmológico estándar consideran que la materia oscura, está ampliamente contenida en el universo, por lo que es importante dar una breve explicación de este concepto.

2.4.1. Materia Oscura

La naturaleza de la materia oscura es un misterio, incluso su existencia es debatible. Dado que la materia oscura no interactúa con la radiación electromagnética, es sumamente difícil tener observaciones directas de su existencia, sin embargo, la manera en que sabemos que está ahí es debido a sus efectos gravitacionales.

Uno de los efectos gravitacionales más importantes observados que soportan la existencia de la materia oscura son las observaciones de curvas de rotación de galaxias espirales.

Se ha observado que en el centro de las galaxias espirales hay visiblemente una mayor cantidad de materia que disminuye conforme se aleja del centro por lo que se espera que la velocidad de las estrellas más lejanas al centro de la galaxia [33] sea:

$$v(R) = \sqrt{\frac{G \cdot M(R)}{R}}, \quad (2.27)$$

donde $M(R)$ es la masa total contenida en una esfera de radio R . Así, la relación entre la velocidad y el radio viene dada por $v(R) \propto R^{-1/2}$, por lo que la velocidad disminuye conforme el radio aumenta.

Sin embargo, se ha observado que para este tipo de galaxias la velocidad de las estrellas se vuelve constante a distancias lejanas del centro (ver imagen 2.4). Este resultado conlleva a que, para una velocidad constante v_0 , la masa contenida dentro de una esfera de radio R es:

$$M(R) = \frac{Rv_0^2}{G}. \quad (2.28)$$

Por lo tanto, la masa incrementa conforme uno se aleje más allá del disco visible [33]. Esto quiere decir que hay un tipo de materia que contribuye con la masa total de la galaxia pero que no aparece visiblemente en las observaciones.

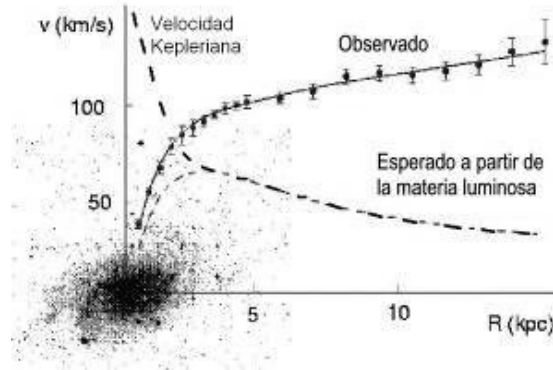


Figura 2.4: Los puntos representan la curva de rotación de la galaxia M33 [34] y la línea continua representa el mejor ajuste. La línea punteada representa la curva de rotación esperada siguiendo la dinámica Newtoniana. Obtenido de [35].

Debido a que la materia oscura interactúa sólo gravitacionalmente es probable que las partículas que la constituyen no colisionen entre sí. Es por ello que si la materia oscura está constituida por un nuevo tipo de partículas, estas tendrán que clasificarse de acuerdo a su velocidad relativista y no relativista como *materia oscura caliente* (hot dark matter) y *materia oscura fría* (cold dark matter), respectivamente.

2.5. Modelos Cosmológicos

El modelo Λ CDM o modelo cosmológico estándar es el modelo cosmológico mayormente aceptado debido a la concordancia teórica y observacional de los parámetros que lo describen. Sin embargo, este modelo consta de algunos problemas como el problema de la constante cosmológica en el cual, valor de la constante Λ es mucho más pequeño de lo esperado. Además, el modelo Λ CDM tiene fallas a escalas más pequeñas, dentro de las galaxias. Una de

estas se debe a que los perfiles de densidad predicho por CDM aumentan abruptamente en la densidad central en contraste con las curvas de rotación de galaxias pequeñas [36].

Algunos modelos cosmológicos alternativos a Λ CDM son el modelo ω CDM y el modelo CPL, sin embargo, en este trabajo no excluirémos a Λ CDM debido a la concordancia que tiene con los datos observacionales. Para nuestro análisis omitiremos estimar el parámetro de radiación Ω_r y el parámetro de curvatura Ω_k . La razón de esto es que, para el parámetro de densidad de radiación se han realizado mediciones con bastante precisión, por lo que no vale la pena ajustar este parámetro [11], mientras que para el parámetro de densidad de curvatura se han estimado valores muy próximos a cero, como en [27].

2.5.1. Modelo Λ CDM

A lo largo de la sección pasada hemos estado describiendo el modelo estándar, sin embargo, a manera de resumen, denominamos al modelo estándar como el modelo que mejor describa la dinámica del universo a partir de las siguientes hipótesis: homogeneidad e isotropía espacial, validez de la relatividad general y densidades de energía que persisten durante toda la historia del universo [7].

Como resultado de esto, se considera que el modelo consta de una constante cosmológica Λ , que es responsable de la expansión acelerada del universo, y la ecuación de estado que obedece es considerando a $\omega = -1$ para toda la historia del universo. La ecuación de Friedmann para este modelo en términos de z es:

$$\frac{H^2(z)}{H_0^2} = \Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0}). \quad (2.29)$$

Donde $\Omega_{\Lambda,0} = 1 - \Omega_{m,0}$. Por lo que los parámetros a estimar son la constante de Hubble H_0 y $\Omega_{m,0}$, el cual consiste mayoritariamente de materia oscura fría, mientras que $\Omega_{\Lambda,0}$ acuñe a la presencia de densidad de energía de vacío o energía oscura, y se puede determinar a partir del parámetro de densidad de materia.

2.5.2. Modelo ω CDM

A diferencia del modelo Λ CDM que asume que el parámetro de estado es constante ($\omega = -1$), el modelo ω CDM propone que dicho parámetro ω debe

tener un valor ω_0 que se ajuste a los datos observacionales [4]. La ecuación de Friedmann de este modelo es:

$$\frac{H^2(z)}{H_0^2} = \Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0})(1+z)^{3(1+\omega_0)}. \quad (2.30)$$

La ecuación anterior sugiere que la densidad de energía de vacío viene encapsulada por el parámetro ω y no exclusivamente por Λ cuando $\omega = -1$, como lo sugiere el modelo 2.5.1. Sin embargo, ambos modelos concuerdan en la existencia de materia oscura y energía oscura.

El modelo ω CDM consta de 3 parámetros principales a estimar: H_0 , Ω_m y ω_0 .

2.5.3. Modelo Chevallier-Polarski-Linder (CPL)

Este modelo consiste en considerar una parametrización de ω en términos del factor de escala, cuyo propósito sea modelar la evolución temporal de $\omega(z(t))$ con base en datos observacionales [37]. Para ello, se parte de una aproximación por series de Taylor alrededor de $a = 1$:

$$\omega(a) = \sum_{i=0}^N (1-a)^i \omega_i.$$

El caso $N = 1$ resulta en:

$$\omega = \omega_0 + \omega_a (1 - a(t)), \quad (2.31)$$

o bien, en función del corrimiento al rojo z [38]:

$$\omega(z) = \omega_0 + \omega_a \frac{z}{1+z}. \quad (2.32)$$

donde ω_0 y ω_a son constantes. Esta parametrización de la constante ω varía con el tiempo, de tal manera que se pueda describir el valor de ω desde el inicio (con $a(t) = 0$), obteniendo $\omega = \omega_0 + \omega_a$ hasta la actualidad en que $a(t) = 1$, resultando en $\omega = \omega_0$. Por lo tanto, podemos afirmar que el modelo CPL describe la evolución de la energía oscura a lo largo del tiempo en términos del corrimiento al rojo z .

El parámetro ω_a se expresa como $\omega_a = \frac{d\omega(0)}{dz}$, cuyo signo indica si ω aumenta o decrece con el corrimiento al rojo z [38].

La ecuación de Friedmann para un parámetro de estado variable en el tiempo se expresa como:

$$\frac{H^2(z)}{H_0^2} = \Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0}) \cdot \exp \left[3 \int_0^z \frac{1 + \omega(z')}{1 + z'} \right]. \quad (2.33)$$

Sustituyendo la expresión (2.32) en la ecuación de Friedmann anterior se obtiene como resultado:

$$\frac{H^2(z)}{H_0^2} = \Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0}) (1+z)^{3(1+\omega_0+\omega_a)} \cdot \exp \left(\frac{-3\omega_a z}{1+z} \right). \quad (2.34)$$

A diferencia de los modelos Λ CDM y ω CDM, el modelo CPL consta de 4 parámetros principales: H_0 , $\Omega_{m,0}$, ω_a y ω_0 [4].

2.6. Estimación de parámetros cosmológicos

Un modelo es un marco teórico que se asume como verdadero y que consta de un conjunto de parámetros θ que deben determinarse [9] a partir de datos $\mathbf{x}$ recolectados. Una herramienta matemática útil para lograr esto es la estadística Bayesiana, por lo que daremos un breve resumen acerca de los conceptos que conforman esta área. Para ello, seguiremos principalmente la siguiente bibliografía [11], [39], [9] y [40].

2.6.1. Estadística Bayesiana

La principal diferencia entre la estadística Bayesiana y la estadística clásica (o frecuentista) recae en la noción de probabilidad que cada una de ellas adopta. La probabilidad, según la estadística frecuentista, está asociada con la frecuencia de resultados de acuerdo a:

$$P = \frac{n}{N}, \quad (2.35)$$

donde n es el número de resultados u ocurrencias de un evento y N el número de ensayos realizados. Sin embargo, esta noción de probabilidad no es del todo útil para cierto tipo de problemas más complejos, por ejemplo: ¿Cuál es la probabilidad de que el día de mañana el cielo esté nublado?. Este tipo de problemas responden a otra noción de probabilidad, usada en la estadística bayesiana, en la cual la probabilidad expresa un grado de creencia basada en el conocimiento del experimentador [39].

Sin embargo existen algunas reglas que la estadística frecuentista y bayesiana obedecen, tales como:

1. La probabilidad $P(x) \geq 0 \forall x \in \Omega$. Donde Ω es el espacio muestral.
2. Regla de la suma de probabilidades:

$$P(x_1 \cup x_2) = P(x_1) + P(x_2).$$

3. Probabilidad conjunta, es la probabilidad de que 2 eventos x_1 y x_2 ocurran simultáneamente:

$$P(x_1, x_2) = P(x_1|x_2)P(x_2).$$

Donde $P(x_1|x_2)$ denota la probabilidad condicional, es decir, la probabilidad de que ocurra x_1 una vez ocurrido x_2

A las variables x se les llama *variables aleatorias* (rv) y son funciones que mapean el espacio muestral Ω a los números reales:

$$x : \Omega \rightarrow \mathbb{R}.$$

A las variables aleatorias se les puede asociar una distribución de probabilidad $\rho(x)$, la cual da la probabilidad de cada resultado en un intervalo x y $x + dx$. La distribución de probabilidad debe cumplir:

$$\int_{-\infty}^{\infty} \rho(x) dx = 1.$$

Del inciso (3) se puede deducir el *Teorema de Bayes*. De la definición de probabilidad conjunta se puede concluir que $P(x_1, x_2) = P(x_2, x_1)$ (por la simultaneidad), esto da como resultado:

$$\begin{aligned} P(A|B) &= \frac{P(A, B)}{P(B)}, \\ &= \frac{P(B|A)P(A)}{P(B)}. \end{aligned}$$

Para la estimación de parámetros cosmológicos es necesario contar con un conjunto de observaciones representada por un conjunto de datos $X = \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N\}$, la meta de estos datos es que puedan interpretarse en términos de un modelo [9] que consta de parámetros θ . En la estadística Bayesiana es natural asociar distribuciones de probabilidad a los parámetros θ [11] dado un conjunto de datos X , es decir, queremos encontrar la probabilidad condicional $P(\theta|X)$.

Siguiendo el teorema de Bayes obtenemos dicha probabilidad condicional como:

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}, \quad (2.36)$$

donde $P(\theta|X)$ es la distribución de probabilidad *posterior de los parámetros* y como mencionamos anteriormente, encapsula la probabilidad de que los parámetros tomen ciertos valores después de hacer el experimento [9]. El término $P(X|\theta)$ es la función de verosimilitud o **likelihood**, $P(\theta)$ es el prior o la información disponible que tenemos de los parámetros antes de obtener los datos. Finalmente, el término $P(X)$ expresa la evidencia.

La existencia de los modelos cosmológicos nos dan cierta información teórica previa que es útil para nuestras observaciones. Además, han existido varias observaciones cosmológicas que arrojaron información acerca de la distribución de probabilidad de diferentes parámetros cosmológicos. Estos resultados teóricos y observacionales actúan como hipótesis sobre los parámetros. De acuerdo a esto escribiremos la distribución de probabilidad posterior de los parámetros partiendo de una hipótesis o modelo I , de tal modo que todas las probabilidades sean condicionales de esta:

$$P(\theta|X, I) = \frac{P(X|\theta, I)P(\theta|I)}{P(X|I)}. \quad (2.37)$$

Si antes de obtener los datos no existe alguna información previa del modelo, entonces se puede tomar un prior que sea proporcional a una constante k , esto es:

$$P(\theta|I) \equiv P(\theta) \propto k. \quad (2.38)$$

La interpretación de esta equivalencia es que si no existe algún conocimiento previo del modelo, el valor de cada parámetro tiene, a priori, la misma probabilidad [11]. Por otro lado, el término de evidencia o evidencia Bayesiana $P(X|I)$ juega un papel de normalización cuando se trata de estimación de parámetros:

$$P(X|I) = \int P(X|\theta, I)P(\theta|I) d^N\theta,$$

donde N es el número de parámetros θ que conforman al modelo. Como la evidencia no depende de los parámetros θ , suele despreciarse en la tarea de estimación de parámetros.

Considerando que el prior es proporcional a una constante y que la evidencia es despreciable, concluimos que la distribución de probabilidad posterior será proporcional al likelihood, (denotada comúnmente como $\mathcal{L}(X|\theta)$), esto es:

$$P(\theta|X, I) \propto \mathcal{L}(X|\theta). \quad (2.39)$$

Máxima verosimilitud

El hecho de que la distribución de probabilidad posterior sea equivalente a la verosimilitud implica hallar el pico máximo de la distribución para la estimación de parámetros. Por lo tanto, la tarea de estimación de parámetros cosmológicos se convierte en un problema de optimización [10].

Podemos expresar las ideas anteriores formalizando los conceptos de función de verosimilitud y máxima verosimilitud por medio de la siguientes definiciones, obtenidas de [39].

Definition 2.6.1 (Función de verosimilitud) *Dada una distribución de probabilidad $P(x|\theta)$, donde x representa una variable aleatoria y θ una colección de parámetros que describen la forma de la distribución de probabilidad y los datos observados $X = \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N\}$, la función de verosimilitud $\mathcal{L}$ se define como:*

$$\mathcal{L}(\theta) = P(x = X|\theta). \quad (2.40)$$

Definition 2.6.2 (Máxima verosimilitud) *El principio de máxima verosimilitud indica que, dada la función de verosimilitud $\mathcal{L}(\theta)$ y buscando el parámetro θ , debemos hallar el valor de θ de tal manera que el valor de $\mathcal{L}$ sea máximo. A este valor de θ que cumple dichas condiciones se le llama estimador de máxima verosimilitud (maximum likelihood) denotado por θ_{ML} :*

$$\max_{\theta} \mathcal{L}(\theta) \equiv \theta_{ML}.$$

Una vez entendido el papel que juega la función de verosimilitud en la estimación de parámetros, es necesario expresar explícitamente la función $\mathcal{L}$ en términos de θ . Para ello, asumamos que tenemos una distribución que consta solamente de un pico, de tal manera que denotemos la media de dicha distribución como $\hat{\theta}$:

$$\hat{\theta} = \int P(\theta|X) \cdot \theta d\theta.$$

Denotamos como θ_0 al valor esperado de $\hat{\theta}$:

$$\langle \hat{\theta} \rangle \equiv \theta_0.$$

Esto quiere decir que el modelo está bien definido y que el valor esperado de $\hat{\theta}$ corresponde al valor real (pico de la distribución) $\theta_0 = \theta_{ML}$ [11].

Cerca del pico podemos aproximar a la función de verosimilitud por medio de una serie de Taylor del logaritmo de $\mathcal{L}$:

$$\ln \mathcal{L}(X|\theta) = \ln \mathcal{L}(X|\theta_0) + \frac{1}{2} \Delta_i \frac{\partial^2 \ln \mathcal{L}}{\partial \theta_i \partial \theta_j} \Delta_j, \quad (2.41)$$

donde $\Delta_i = \theta_i - \theta_{0i}$ y $\Delta_j = \theta_j - \theta_{0j}$. Aplicando la función exponencial a la ecuación (2.41) obtenemos:

$$\mathcal{L}(X|\theta) = \mathcal{L}(X|\theta_0) \cdot \exp \left[-\frac{1}{2} \Delta_i H_{ij} \Delta_j \right], \quad (2.42)$$

donde H_{ij} es la matriz Hessiana:

$$H_{ij} = -\frac{\partial^2 \ln \mathcal{L}}{\partial \theta_i \partial \theta_j},$$

esta matriz nos dice si las estimaciones de distintos parámetros θ_i, θ_j de un modelo están correlacionados o no.

Chi cuadrada (χ^2)

La expresión (2.42) resulta en que la función de verosimilitud se puede aproximar por una distribución Gaussiana, esto tiene ciertas implicaciones, una de ellas es que maximizar la función $\mathcal{L}$ implica minimizar la cantidad χ^2 . Donde χ^2 se expresa como:

$$\chi^2 = \Delta_i H_{ij} \Delta_j. \quad (2.43)$$

Por lo tanto, la función de verosimilitud se relaciona con χ^2 como:

$$\mathcal{L} = \mathcal{L}_0 \cdot \exp \left(-\frac{\chi^2}{2} \right). \quad (2.44)$$

Una definición más práctica de la cantidad χ^2 se puede apreciar con el siguiente ejemplo obtenido de [39]:

Ejemplo. Supongamos que tenemos N mediciones independientes de una cantidad distribuida de manera Gaussiana, denotemos estas mediciones por $\{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N\}$. Los parámetros que nos interesan determinar son la media de la distribución μ y la desviación estándar de la distribución σ , por lo tanto $\theta = \{\mu, \sigma\}$. Así, la función de verosimilitud está dado por:

$$\mathcal{L}(\mu, \sigma) = \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2} \frac{(\hat{x}_i - \mu)^2}{\sigma^2}\right),$$

donde la χ^2 se define como:

$$\chi^2 = \sum_{i=1}^N \frac{(\hat{x}_i - \mu_i)^2}{\sigma_i^2}. \quad (2.45)$$

Obteniendo así a $\mathcal{L}$ en términos de χ^2 :

$$\mathcal{L} = \mathcal{L}_0 \cdot \exp\left(-\frac{\chi^2}{2}\right).$$

En conclusión, para la estimación de parámetros cosmológicos será necesario minimizar la cantidad χ^2 . El modelo que mejor se acople a los datos observacionales (2.8) será aquél con los parámetros que mejor dan la prueba χ^2 .

2.7. Criterios de información

Como se mencionó anteriormente, la prueba χ^2 no es suficiente para saber qué modelo es el mejor de todos ya que pueden existir modelos que se ajusten a los datos pero que tienen muchos parámetros y otros que se ajusten bien con pocos parámetros, para hacer esta exclusión se utilizan los criterios de información (IC's) que son modelos de selección estadísticos [4]. En cosmología los criterios de información más utilizados son el criterio de información Akaike (AIC) y el criterio de información Bayesiana (BIC). El AIC se expresa como:

$$AIC = -2 \ln \mathcal{L}_{max} + 2k, \quad (2.46)$$

donde k es el número de parámetros y $\mathcal{L}_{max}$ es la verosimilitud máxima. Por su parte el BIC se expresa:

$$BIC = -2 \ln \mathcal{L}_{max} + k \ln(N), \quad (2.47)$$

donde N es el número de datos observados. Utilizando estos criterios se concluye que el mejor modelo es aquel que minimice los criterios BIC y AIC.

2.8. Observaciones y datos

Actualmente la cosmología consta de varias observaciones y de varios experimentos que han aprovechado el avance tecnológico, con el fin de tener telescopios y satélites que proporcionen datos confiables.

En esta sección se pretende dar una descripción de las observaciones cosmológicas utilizadas en este trabajo para la estimación de parámetros, así como dar a conocer los proyectos observacionales que hicieron posible la toma de datos y sus correspondientes resultados obtenidos.

2.8.1. Cronómetros cósmicos

Uno de los parámetros a estimar, además de los parámetros de densidad Ω , es el parámetro de Hubble $H(z)$. Para lograr esto es útil utilizar el *método de edad diferencial* [41] ya que se pueden hacer mediciones directas de $H(z)$ considerando la siguiente relación:

$$\begin{aligned} H(z) &= -\frac{1}{1+z} \frac{dz}{dt}, \\ &\approx -\frac{1}{1+z} \frac{\Delta z}{\Delta t}. \end{aligned} \tag{2.48}$$

Como la cantidad Δz ha sido bien estudiada y su proceso de medición es precisa, solo queda por determinar Δt dada una cantidad infinitesimal del corrimiento al rojo.

Por lo tanto, como se menciona en [41] es necesario obtener mediciones de la diferencia de edad (Δt) entre dos galaxias de evolución pasiva, que se formaron al mismo tiempo pero están separadas por una cantidad Δz . Varios estudios han revelado que este tipo de galaxias se han formado a redshifts muy altos ($z \geq 2$) en escalas de tiempo cortas ($< 1\text{Gyr}$), agotando casi por completo su reserva de gas en las primeras etapas de su vida, provocando que evolucionen pasivamente hasta el presente [42].

Para los propósitos de este trabajo utilizaremos un conjunto de 26 mediciones del parámetro de Hubble obtenido de [43] considerando un rango de $0.09 < z < 2$. El gráfico de los datos se puede observar en la imagen de abajo.

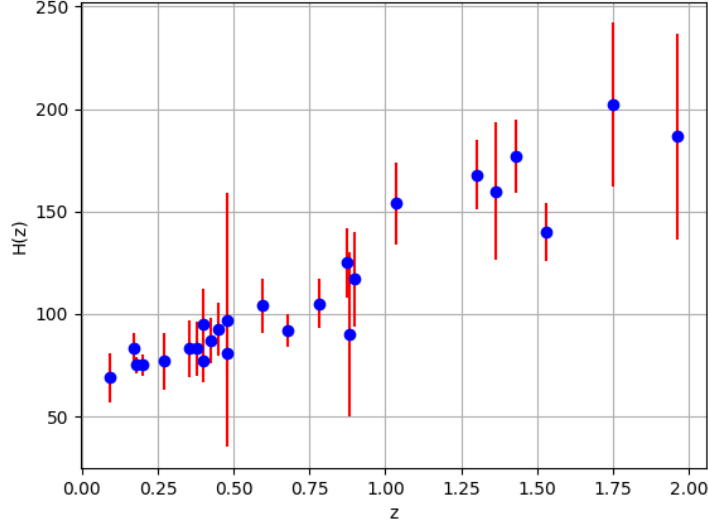


Figura 2.5: Datos obtenidos del parámetro de Hubble $H(z)$ en función del corrimiento al rojo z en un intervalo $0.09 < z < 2$.

2.8.2. Supernovas

Las SNIa son una clase de supernovas que ocurren en sistemas binarios de estrellas, donde una enana blanca altamente densa atrae masa de la capa externa de otra estrella (una gigante roja en particular), volviendo a la enana blanca más masiva. Esto provoca que surjan reacciones nucleares en la superficie de la enana blanca, y cuando alcance una masa límite (masa Chandrasekhar) equivalente a 1.4 masas solares, la enana blanca explotará.

Este tipo de supernovas emiten aproximadamente la misma cantidad de luz al estallar por lo que pueden estandarizarse y así, ser usadas como *candelas estándar* [11]. Una candela estándar es un objeto que produce una luminosidad conocida y se usan para medir distancias muy grandes. Podemos determinar la distancia que hay entre un objeto que emite luz y la Tierra a partir de su brillo, usando la ecuación:

$$F = \frac{L}{4\pi D_L^2} \rightarrow D_L = \sqrt{\frac{L}{4\pi F}}, \quad (2.49)$$

donde F es la potencia recibida por unidad de área, L es la luminosidad de la fuente y D_L es la distancia entre la Tierra y dicha fuente.

Para los datos de SNIa utilizamos la compilación de 31 puntos del proyecto Joint Light-Curve Analysis (JLA) [44]. La relación entre estos datos observacionales y los parámetros cosmológicos se obtiene de la siguiente fórmula [11]:

$$D_L = D_M(1 + z), \quad (2.50)$$

donde D_M es el diámetro de distancia angular comóvil, el cual se puede relacionar con la ecuación de Friedmann (ver apéndice B), obteniendo así:

$$D_L = \frac{c(1+z)}{H_0} \text{sinn} \left(\int_0^z \frac{H_0}{H(z')} dz' \right). \quad (2.51)$$

Con

$$\text{sinn}(x) = \begin{cases} \sinh(\sqrt{\Omega_k}x)/\sqrt{\Omega_k}, & \text{si } \Omega_k > 0, \\ x, & \text{si } \Omega_k = 0, \\ \sin(\sqrt{-\Omega_k}x)/\sqrt{-\Omega_k}, & \text{si } \Omega_k < 0. \end{cases}$$

Como convención para las observaciones cosmológicas se define el módulo de distancia μ como:

$$\mu(z) \equiv 5 \log \left(\frac{D_L(z)}{10 \text{pc}} \right).$$

El módulo de distancia representa la diferencia de magnitud luminosa entre un objeto observado y si el objeto observado a una distancia de 10 pc [45]. De esta manera definimos al módulo de distancia como una diferencia entre magnitud aparente y magnitud absoluta:

$$\mu(z) = m - M. \quad (2.52)$$

Los resultados obtenidos de la compilación Union2.1 se muestra en la figura 2.6

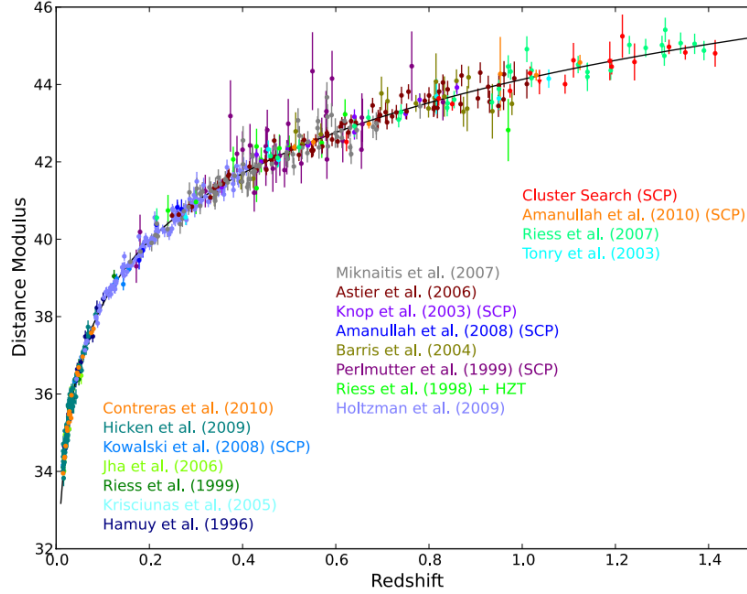


Figura 2.6: Diagrama de Hubble para la compilación Union2.1. La línea sólida representa el mejor ajuste de un modelo Λ CDM para un universo plano. Obtenido de [46].

2.8.3. Otros datos: Oscilaciones Acústicas de Bariones

Debido a la gran temperatura a la que se encontraba el universo unos instantes después del Big Bang, el universo primigenio estaba compuesto por un plasma de radiación electromagnética y materia bariónica. La radiación electromagnética era lo suficientemente energética como para prevenir la formación de átomos neutros [47] permitiendo que los electrones libres actúen como un pegamento entre los fotones y los bariones. Por otra parte, los fotones arrastran a los electrones por dispersión Thomson y viceversa, y los electrones arrastran a los bariones mediante atracción electrostática y viceversa [7]. Esta competencia entre presión de radiación y gravedad generaron ondas esféricas (también llamadas ondas acústicas) en el plasma [11], cuyas oscilaciones provocaron que haya regiones con mayor densidad. La materia oscura, al no interactuar con los fotones y por ende, no ser afectada por la presión de radiación, se acumuló entorno a las inhomogeneidades iniciales del plasma por efectos puramente gravitacionales [47], [48].

Posteriormente, con la expansión y enfriamiento del universo, la temperatura del plasma disminuyó, provocando que se formaran los primeros átomos

de hidrógeno, helio y litio (etapa de recombinación). Como consecuencia del inicio de la recombinación, las sobre-densidades de materia oscura concentradas debido a las inhomogeneidades iniciales del plasma y las zonas concentradas de materia bariónica arrastradas por la onda acústica [47] solo crecieron debido a la inestabilidad gravitatoria [7]. El resultado de esta interacción instantes antes del desacople, se puede observar actualmente. Debido a que los bariones fueron arrastrados por las ondas de sonido, estos se encuentran, ahora en forma de galaxias, en la frontera de varios cascarones esféricos, todos de radio constante r_d (ver imagen 2.7), llamado horizonte sónico y en cuyo centro hay una sobre-densidad de materia oscura, sin embargo, debido a efectos gravitacionales también se pueden hallar algunas galaxias en los centros de estos cascarones.

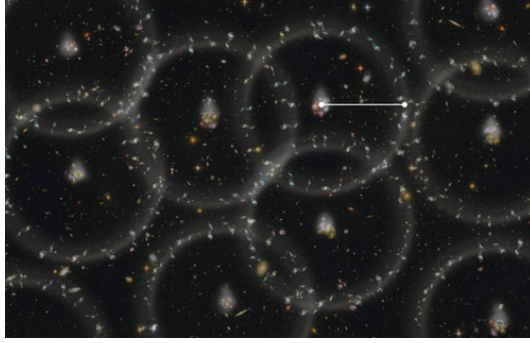


Figura 2.7: Imagen demostrativa de los cascarones esféricos, en cuya frontera se encuentra materia bariónica y en el centro, sobre-densidades de materia oscura, junto con materia bariónica. Créditos: proyecto Baryon Oscillation Spectroscopic Survey (BOSS).

El horizonte sónico funge como una regla característica ya que representa las zonas conectadas causalmente por ondas de presión en lugar de luz [7] y cuyo valor es aproximadamente 150 Mpc [48]. Esta regla característica se expresa como:

$$r_d = \int \frac{c_z}{H(z)} dz, \quad (2.53)$$

con $c_z^2 = \frac{1}{3} \frac{1}{1+R}$ y $R = \frac{3\rho_b}{4\rho_\gamma}$.

Dado que las oscilaciones acústicas de bariones (BAO por sus siglas en inglés) determinan la distribución de zonas donde queda una mayor densidad de galaxias, sus mediciones a partir de catálogos de galaxias pueden ser

útiles para medir la geometría del universo a través de la relación distancia-redshift [49]. Además, de los resultados obtenidos del proyecto SDSS [48], las BAO's pueden ser usadas para medir parámetros cosmológicos. Para ello es necesario que las mediciones BAO tengan influencia a través del radio de horizonte sónico r_d , así como de la distancia de Hubble $D_H(z)$, el diámetro de distancia angular comóvil $D_M(z)$ y la distancia angular volumétrica $D_V(z)$.

La siguiente figura 2.8 muestra las mediciones BAO de los distintos proyectos observacionales mostrados en la leyenda.

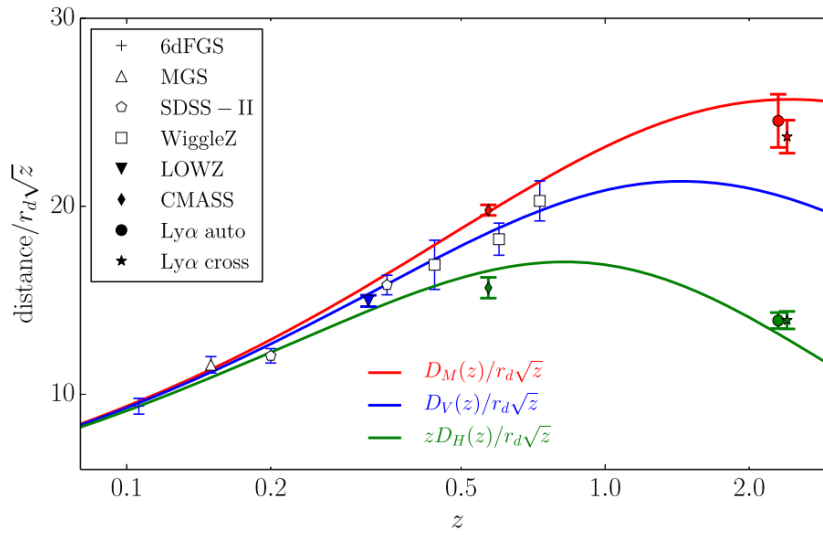


Figura 2.8: Mediciones BAO determinadas de la relación distancia-redshift. Los puntos de colores azul, rojo y verde muestran las mediciones BAO de $D_V(z)/r_d$, $D_M(z)/r_d$ y $zD_H(z)/r_d$, respectivamente, a partir de los proyectos mostrados en la leyenda. Obtenido de [50].

3 Optimización

La idea principal de la optimización es encontrar, de entre toda una gama de soluciones a un problema, la mejor solución de este.

Inherentemente los seres humanos y algunos grupos de animales buscan optimizar algún proceso que les traiga cierto beneficio, en este sentido, optimizar es tomar decisiones y tomar las mejores decisiones conllevan a mejores beneficios. Sin embargo, para la toma de decisiones es necesario modelar el problema para el cual se obtendrá la mejor solución, el campo encargado de modelar estos problemas es la *optimización matemática*, ya que se encarga de modelar por medio de ecuaciones un problema dadas ciertas restricciones. Un ejemplo clásico de la optimización matemática es el de disminuir el coste de producción de un producto considerando restricciones en la cantidad o coste de materiales para la elaboración de dicho producto, de tal manera que la producción, la ganancia o ambas sea lo más eficaz posible.

La optimización matemática tiene diversas aplicaciones en la industria, ciencia, finanzas y por su puesto en el ajuste de datos [51]. Debido a esta gran cantidad de aplicaciones se han desarrollado diversos métodos de optimización, adaptados a la complejidad del problema en cuestión. Entre ellos se destacan la optimización estocástica, optimización no estocástica, optimización continua y optimización discreta, entre otras [52].

En este trabajo nos enfocaremos principalmente en los algoritmos estocásticos y no estocásticos para justificar el uso del algoritmo PSO como algoritmo de optimización estocástico en la cosmología.

3.1. Análisis de los problemas de optimización

Para entender con mayor claridad la optimización matemática es necesario ahondar en algunas definiciones clave de esta área. El primer paso para resolver un problema de optimización es, una vez entendido el problema, modelarlo por medio de una función matemática, digamos: $f(\mathbf{x})$. A esta función se le conoce como *función objetivo* y depende de valores vectoria-

les $\mathbf{x}=(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ llamadas *variables*, las entradas de estos vectores representan características cuantitativas del problema planteado. Así, el objetivo de resolver un problema de optimización es maximizar o minimizar una función f con el fin de encontrar los valores de las variables donde se hallen dichos óptimos [52].

Dependiendo de la naturaleza del problema a optimizar, es común que se consideren ciertas restricciones en la función objetivo que son representadas por ecuaciones o desigualdades, es decir, una función objetivo puede estar sujeta a funciones del tipo $h(\mathbf{x}) = 0$ o $g(\mathbf{x}) \leq 0$.

El esquema 3.1 representa las curvas de nivel de una función ($f(\mathbf{x}) = \text{cte}$) restringida por funciones $c_i(\mathbf{x})$, $i \in \{1, 2\}$. A los puntos que satisfacen las restricciones se les llama *solución factible*.

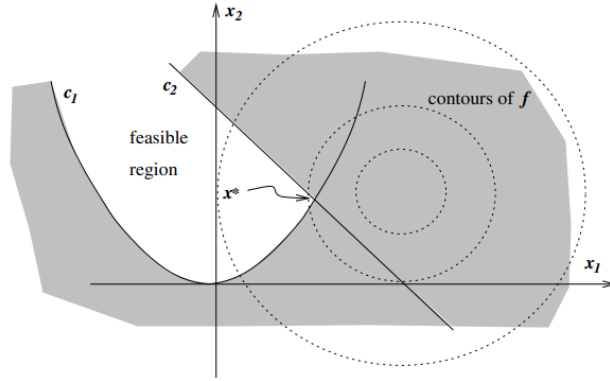


Figura 3.1: Ejemplificación geométrica de una función $f(\mathbf{x})$ restringida por funciones $c_i(\mathbf{x})$. Obtenido de [52].

Así, un punto $\mathbf{x}$ es solución factible si cumple:

$$\begin{aligned} g_i(\mathbf{x}) &= 0, i \in \mathcal{I}, \\ g_i(\mathbf{x}) &\leq 0, i \in \mathcal{J}. \end{aligned}$$

Donde $\mathcal{I}$ y $\mathcal{J}$ son conjuntos de índices asociados al número de funciones de restricción a considerar. A la región Ω que consta de todos los puntos que son solución factible se le conoce como *región factible*. La región factible representa a lo que llamamos anteriormente como soluciones eficaces, sin

embargo en un problema de optimización se busca encontrar un punto $\mathbf{x}^* \in \Omega$ que sea la mejor de las soluciones factibles.

El criterio de la mejor solución factible depende de maximizar o minimizar la función objetivo según el problema. En este sentido, un *máximo global* es una solución factible $\mathbf{x}^* \in \Omega$ tal que $f(\mathbf{x}^*) \geq f(\mathbf{x})$ para toda $\mathbf{x} \in \Omega$, $\mathbf{x} \neq \mathbf{x}^*$. Similarmente, un *mínimo global* es una solución factible $\mathbf{x}^* \in \Omega$ tal que $f(\mathbf{x}^*) \leq f(\mathbf{x})$ para toda $\mathbf{x} \in \Omega$, $\mathbf{x} \neq \mathbf{x}^*$. A los máximos y mínimos globales se les conoce como *soluciones óptimas* ya que representan el mejor resultado posible según los criterios del problema y van de acuerdo a si el problema consiste en minimizar o maximizar la función objetivo.

La evaluación de las soluciones óptimas en la función objetivo f garantiza la obtención de un *valor óptimo*.

Por practicidad suele ser más útil encontrar soluciones óptimas *locales* en lugar de soluciones óptimas globales como las mencionadas anteriormente. De esta manera se dice que $\mathbf{x}^* \in \Omega$ es un *mínimo local* si existe una vecindad $\mathcal{V}$ centrada en $\mathbf{x}^*$ tal que $f(\mathbf{x}^*) \leq f(\mathbf{x})$ para todo $\mathbf{x} \in \mathcal{V}$. Similarmente, se dice que $\mathbf{x}^*$ es un *máximo local* si existe una vecindad $\mathcal{V}$ centrada en $\mathbf{x}^*$ tal que $f(\mathbf{x}^*) \geq f(\mathbf{x})$ para todo $\mathbf{x} \in \mathcal{V}$. Computacionalmente es recomendable encontrar máximos o mínimos locales (extremos locales) en lugar de extremos globales debido a que la mayoría de algoritmos de optimización no pueden recorrer todos los puntos de una función objetivo pero sí ciertas regiones de la función.

Con estos conceptos fundamentales que se consideran en los problemas de optimización se tiene una idea más clara de los criterios que se deben tomar en cuenta para resolver esta clase de problemas. Sin embargo, aún no hemos mencionado de qué manera se pueden resolver, para ello veremos algunos métodos de optimización comúnmente utilizados en la resolución de estos problemas ya que no existe una forma única de resolverlos, si no que hay métodos más óptimos para una clase de problemas que para otros.

3.2. Métodos de optimización

Las definiciones mostradas anteriormente son muy generales ya que sobre ellas se construye todo el campo de la optimización matemática, esto significa que de estas definiciones se despliegan muchos otros conceptos que han dado lugar a la existencia de diversos métodos de optimización. Sin embargo, para propósitos de este trabajo podemos clasificar varios de estos métodos en dos: optimización determinista y optimización estocástica.

3.2.1. Optimización determinista y optimización estocástica

Los métodos de optimización determinista tienen la capacidad de encontrar el mejor resultado global de un problema de optimización [53], es decir, esta clase de algoritmos garantizan que al menos se encontrará un óptimo global. Estos algoritmos se aplican a problemas cuyo modelo tiene características (variables) fijas, por lo que las componentes del vector $\mathbf{x}$ no cambian con el tiempo. Para que estos algoritmos logren su cometido deben cumplir requisitos de robustez, eficiencia y precisión [52].

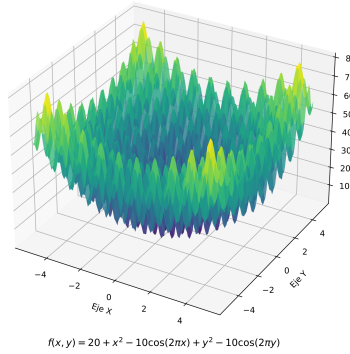


Figura 3.2: La función Rastrigin consta de múltiples extremos locales.

En el caso de funciones con alta dimensionalidad o que constan de un gran número de extremos locales, los algoritmos deterministas pueden no ser eficaces desde el punto de vista computacional, ya que el tiempo de cómputo para encontrar los óptimos globales puede incrementar de acuerdo a la complejidad de la función, un ejemplo de este tipo de funciones es la función *Rastrigin* mostrada en 3.2. Es por ello, que para esta clase de problemas se empezó a estudiar la optimización estocástica, la cuál busca encontrar soluciones adecuadas empleando procesos aleatorios, esto ayuda a conocer la probabilidad de diferentes escenarios posibles de un problema de coste incierto. La ventaja de la optimización estocástica recae en la búsqueda de una solución óptima considerando parámetros desconocidos [54], lo que amplía el espacio de búsqueda. Sin embargo, pese a que existe la probabilidad de encontrar un óptimo global, la optimización estocástica no asegura al 100 % encontrarlo, esto depende una vez más del tiempo de cómputo y de la complejidad del problema. Dado que la optimización estocástica no solo depende de las variables $\mathbf{x}$ consideradas en el problema de optimización sino que también depende de variables aleatorias w_i , se garantiza que los

algoritmos estocásticos no queden intrincados en algún óptimo local. Por lo tanto, los algoritmos de optimización estocástica no garantizan hallar el óptimo global pero si garantizan hallar una buena solución con un menor coste computacional que los algoritmos deterministas.

3.2.2. Métodos heurísticos y meta-heurísticos

Los algoritmos de optimización estocástica pertenecen a una clase de métodos conocidos como *heurísticos* y *meta-heurísticos*. La heurística es un método de aproximación que asegura una solución de confianza [55] y utiliza técnicas basadas en experiencia y aprendizaje para la solución de problemas. En la optimización matemática, una heurística es una técnica diseñada para resolver problemas más rápidamente cuando los métodos clásicos son muy lentos, o bien, para hallar una solución aproximada cuando los métodos clásicos fallan en hallar la solución exacta [56]. Algunas técnicas heurísticas consisten de los siguientes algoritmos:

- Algoritmo de recocido simulado (Simulated annealing).
- Búsqueda tabú (Tabu search).
- Redes neuronales (Neural networks).

Por su parte, los métodos meta-heurísticos se definen como métodos de exploración y búsqueda.

Las técnicas metaheurísticas son una mejora en los procesos heurísticos ya que prometen resolver diversos problemas de optimización utilizando el método heurístico previo existente a dicho problema. Los algoritmos meta-heurísticos hacen pocas o incluso ninguna suposición del problema a optimizar, además, estos algoritmos se pueden clasificar en métodos basados en población y basados en trayectoria [57], los cuales utilizan técnicas de exploración y búsqueda, por lo que pueden buscar en espacios que contengan un gran número de soluciones candidatas [56]. Algunas técnicas meta-heurísticas consisten de los siguientes algoritmos:

- Algoritmo genético (genetic algorithm).
- Colonia de abejas artificiales (artificial bee colony).
- Optimización por enjambre de partículas (particle swarm optimization).

Algo sumamente interesante de los métodos heurísticos y meta-heurísticos es que adoptan conceptos de varios campos de la ciencia como la física, la biología, entre otros. A la mayoría de los algoritmos que adoptan conceptos de la biología se les conoce como *algoritmos bio-inspirados*.

3.2.3. Ejemplos de optimización determinista

En el siguiente capítulo explicaremos con mayor detalle los algoritmos bio-inspirados, que como mencionamos anteriormente, pertenecen a la clase de algoritmos estocásticos. Por lo tanto, vale la pena concluir esta sección con dos métodos de optimización deterministas que han sido ampliamente estudiados y eficaces, como el método de programación lineal (algoritmo simplex) y el método de gradientes (algoritmo de gradiente descendente determinista).

Descenso de gradiente determinista

Algunos algoritmos de optimización determinista son algoritmos iterativos cuyo desarrollo está inspirado en métodos analíticos desarrollados por el cálculo diferencial para la obtención de máximos y mínimos de una función, un ejemplo analítico de esto es el método de *multiplicadores de Lagrange* [58], ya que estos retoman algunos conceptos matemáticos como el gradiente.

El método de descenso de gradiente, es una de las primeras técnicas utilizadas para la minimización de funciones objetivo multidimensionales [59] por lo que ha sido ampliamente estudiado, convirtiéndolo en un método base para otros algoritmos más sofisticados como el método de descenso de gradiente estocástico.

El descenso de gradiente es un algoritmo que busca encontrar las variables para las cuales se halla un mínimo, y la forma de hallarlas es por medio de pasos, donde cada paso apunta en la dirección donde la función decrece. Se parte un punto inicial $\mathbf{x}_0$ y se ocupa un método iterativo sobre este vector para hallar la solución óptima $\mathbf{x}^*$ [59]. Dicho método iterativo está expresado por:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k), \quad (3.1)$$

donde k expresa el número de iteraciones generadas, α representa el tamaño de cada paso y $\nabla f(\mathbf{x})$ representa (por definición de gradiente) la dirección donde la función objetivo decrece y con ello garantizar que $f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$.

La forma en que se calcula el gradiente es por métodos numéricos. Para ello es necesario implementar otro vector $\tilde{\mathbf{x}}_i \in \mathbb{R}^n$ cuya entrada i -ésima x_i se

sustituye por $x_i + h$, así $\tilde{\mathbf{x}}_i = (x_1, x_2, \dots, x_i + h, \dots, x_n)$. Con ello se procede a calcular el gradiente de f en el punto $\tilde{\mathbf{x}}_i$ mediante la fórmula:

$$\nabla f|_{x_i} = \frac{f(\tilde{\mathbf{x}}_i) - f(\mathbf{x})}{h}.$$

El propósito de este algoritmo es que el valor de $\nabla f(\mathbf{x}^k)$ tienda a cero en cada una de sus entradas para asegurar que se ha alcanzado un mínimo, por ello se debe implementar una condición de parada considerando que la norma del gradiente en la k -ésima iteración sea menor o igual a un número arbitrariamente pequeño S :

$$\|\nabla f(\mathbf{x}^k)\| \leq S.$$

El pseudocódigo que ejemplifica la implementación de este algoritmo se muestra abajo.

Algorithm 1 *Descenso de gradiente determinista*

Require: *inicializa:* $f, x^0 = x_{init} \in \mathbb{R}^d$, $\text{stop}=S$.

```

1: while  $k > 0$  do
2:    $x^{k+1} = x^k - \alpha \nabla f(\mathbf{x}^k)$ 
3:   if  $\|\nabla f(\mathbf{x}^k)\| \leq S$  then
4:     break
5:   else
6:     pass
7: return  $x^k$ 

```

El descenso de gradiente es un algoritmo muy utilizado para hallar mínimos de funciones no lineales debido a la sencilla implementación del algoritmo (ver 1). Sin embargo, la precisión y el tiempo de convergencia depende del parámetro α que puede tomar un valor constante positivo durante toda la ejecución del programa o ser recalculado en cada iteración (α_k) [60]. Para ambos casos existen diferentes técnicas de cómo obtener un parámetro α o α_k adecuado para que el tiempo de convergencia no sea tan extenso y para que no exista una divergencia en el tamaño del paso al estar cerca del mínimo. Más información relacionada al cálculo de α_k se puede hallar en [52].

Programación lineal - Método Simplex

La programación lineal es el nombre de una rama de las matemáticas aplicadas que lidian con los problemas de optimización de una forma particular [61] ya que, si los métodos de gradientes utilizan herramientas del

cálculo diferencial, la programación lineal utiliza herramientas del álgebra lineal.

La programación lineal se encarga de hallar máximos y mínimos de funciones objetivo lineales cuyas constricciones están expresadas como desigualdades. La forma canónica de expresar un problema de este tipo es usando la notación matricial dada por el álgebra lineal:

$$\begin{aligned} &\text{maximizar} && f(\mathbf{x}) = \mathbf{c}^T \mathbf{x} \\ &\text{sujeto a} && \mathbf{Ax} \leq \mathbf{b}, \\ &&& \mathbf{x} \geq 0. \end{aligned} \tag{3.2}$$

Donde $\mathbf{A}$ es una matriz de $m \times n$, $\mathbf{x}$ es el vector solución al problema, $\mathbf{b}$ es el vector de restricciones para el sistema de inecuaciones y $\mathbf{c}$ es un vector de coeficientes. Podemos reescribir la expresión anterior en términos de sumas:

$$\begin{aligned} \mathbf{c}^T \mathbf{x} &= \sum_{j=1}^n c_j x_j, \\ \mathbf{Ax} \leq \mathbf{b} &\rightarrow \sum_j a_{ij} x_j \leq b_i, \\ \mathbf{x} \geq 0 &\rightarrow x_j \geq 0. \end{aligned}$$

Con $i \in \{1, \dots, m\}, j \in \{1, \dots, n\}$. El primer paso para resolver un problema de programación lineal es llevar la expresión (3.2.3) a la forma estándar, esto se hace añadiendo *variables de holgura* a las restricciones para obtener una igualdad estricta. Así, la expresión para resolver un problema de programación lineal queda expresada como:

$$\begin{aligned} &\text{maximizar} && z = \sum_{j=1}^n c_j x_j \\ &\text{sujeto a} && \sum_{j=1}^n a_{ij} x_j + s_i = b_i, \\ &&& x_j \geq 0, \\ &&& s_i \geq 0. \end{aligned} \tag{3.3}$$

En caso de que la restricción sea del tipo $\geq$ se deberán restar las variables de holgura s_i . Otra consideración para obtener la forma estándar

de un problema de programación lineal es considerando la restricción de no negatividad para todas las variables, es decir, $x_j \geq 0 \forall j \in \{1, \dots, n\}$ [62].

Cualquier vector $\mathbf{x}$ que satisfaga las restricciones del problema de programación lineal se le conoce como *solución factible del problema*. Cabe resaltar que un conjunto de soluciones factibles de cualquier problema de programación lineal se puede representar mediante un poliedro convexo y más aún, la solución óptima se encuentra en un vértice de dicho poliedro, para ello es necesario también que la solución factible sea acotada. Estas afirmaciones son resultados de 2 teoremas que se pueden consultar en [63].

Teniendo en cuenta lo anterior, el primer paso para resolver un problema de programación lineal es obtener la forma estándar del problema, una vez hecho eso es posible implementar el algoritmo Simplex. Este algoritmo recorre iterativamente algunos vértices del poliedro (conjunto factible) de tal manera que en cada iteración o recorrido haya una mejora en la función objetivo (ver 3.3), para ello es necesario evaluar la función objetivo en cada solución factible. La mejora depende de si el propósito es maximizar o minimizar la función.

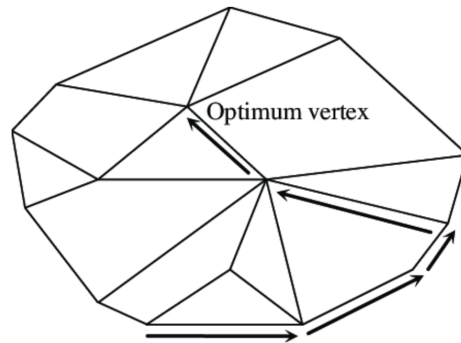


Figura 3.3: Evolución de la optimización del algoritmo Simplex. Obtenido de [64].

El punto de partida del algoritmo Simplex es considerando la solución trivial $\mathbf{x}=0$, para lo cual se obtendrán soluciones a las variables de holgura:

$$\begin{aligned}
a_{11}x_1 + \dots + c_{1n}x_n + 1s_1 + 0s_2 \dots + 0s_m &= b_1, \\
a_{21}x_1 + \dots + a_{2n}x_n + 0s_1 + 1s_2 \dots + 0s_m &= b_2, \\
&\vdots \\
&\vdots \\
&\vdots \\
a_{m1}x_1 + \dots + a_{mn}x_n + 0s_1 + 0s_2 \dots + 1s_m &= b_m.
\end{aligned}$$

Es claro que con esta solución, la función objetivo toma el valor $z = 0$. La iteración se realiza entonces encontrando una variable que se pueda incrementar a expensas de que otra variable del problema (en su versión estándar) disminuya con el fin de que se produzca una mejora general en la función objetivo [61].

La condición de parada se obtiene cuando la función objetivo no incrementa al evaluar la solución factible obtenida de alguna iteración, es así que se ha encontrado la solución óptima. Se puede encontrar información más detallada de la programación lineal y del algoritmo simplex en [65] y [62].

La programación lineal tiene aplicaciones en la economía, en la industria aérea y militar, entre otras, por lo que dentro del campo de la optimización, la programación lineal ha sido fuertemente estudiada a lo largo de los años desde su implementación en 1947, por lo que desde el punto de vista computacional el método simplex se ha convertido en una herramienta eficaz para resolver problemas de programación lineal [66].

4 Algoritmos bio-inspirados

Debido a que la optimización matemática es útil en muchas ramas de la ciencia y de la industria, existe un desarrollo amplio en los problemas y soluciones que este campo puede tratar. Es por ello que los problemas de optimización son cada vez más complicados de resolver con los métodos y/o algoritmos más comunes. Considerando este inevitable problema, los investigadores de la optimización han volteado a ver hacia la naturaleza, ya que si de optimización se trata, la naturaleza es capaz de encontrar la mejor solución posible.

Hablar del comportamiento de la naturaleza puede ser algo ambiguo debido a la complejidad, extensión y falta de conocimiento o comprensión de esta, por lo que uno se puede preguntar: ¿De qué forma la optimización se ha inspirado de la naturaleza para implementar algoritmos de optimización que resuelvan problemas complejos?. Pues bien, principalmente la *optimización bio-inspirada* busca emular la manera en que la naturaleza resuelve problemas que involucren la **supervivencia** y **adaptabilidad** de los seres vivos. Es así, que los algoritmos bio-inspirados se pueden dividir en 2 principales bloques: *algoritmos evolutivos* y *algoritmos de enjambre*, el primero inspirado en la evolución natural y el segundo en el comportamiento que tienen los grupos de animales para sobrevivir [67].

Los algoritmos evolutivos y los algoritmos de enjambre se consideran meta-heurísticos ya que su implementación e iteración requieren de aleatoriedad, teniendo así un amplio espacio de búsqueda y mayor rapidez en encontrar soluciones locales de las cuales alguna de ellas puede ser el óptimo global. Dado que los algoritmos meta-heurísticos utilizan una función heurística, su exploración se basa en el aprendizaje y experiencia [14] [68], pudiendo así emular el comportamiento de animales o procesos evolutivos [69].

En lo que resta del capítulo veremos de manera más detallada la consistencia de los algoritmos evolutivos y los algoritmos de enjambre junto con un breve ejemplo para cada uno de ellos.

4.1. Algoritmos evolutivos

La teoría evolutiva de Darwin ha servido como fuente de inspiración para la creación de un grupo de algoritmos bio-inspirados, los *algoritmos evolutivos*, ya que desde el punto de vista de la optimización, hallar la mejor solución a un problema sería el equivalente a obtener los individuos mejor adaptados a un ambiente. Para emular este proceso se tiene que inicializar aleatoriamente un conjunto de puntos en el espacio de soluciones (una población), y por medio de métodos de selección, mutación y recombinación obtener una evolución de los individuos hasta llegar a los mejores adaptados al problema. Aunque algunos de estos métodos mencionados anteriormente no los comparten todos los algoritmos evolutivos, la idea de utilizar procesos para obtener mejores soluciones a partir de una población inicializada estocásticamente, prevalece [67]. La idea general para implementar un algoritmo evolutivo es la siguiente:

- Generar aleatoriamente un conjunto de soluciones, es decir, un conjunto factible Ω . Dado que cada elemento del conjunto es una solución factible se debe cuantificar la calidad de dicha solución, para ello se aplica una **función fitness** al conjunto Ω . El valor del fitness incrementará en cada elemento conforme su fitness se acerque al de un valor óptimo. El valor de la función fitness puede depender del entorno y para ello se considera la diferencia que tiene el valor fitness en un punto del espacio con respecto a un óptimo [70].
- El proceso de selección escoge los individuos (elementos) con mejor fitness y usando los procesos de **mutación** y **recombinación** se obtiene una nueva generación de individuos mejor adaptados, o bien, con un fitness mayor que los de la generación pasada. El proceso de recombinación se aplica como un operador binario $\mathcal{R}$ que toma 2 individuos del conjunto factible (padres) y da lugar a un nuevo individuo (hijo). Por otro lado, el proceso de mutación se aplica como un operador unitario $\mathcal{M}$ sobre un solo individuo y de igual manera da lugar a un nuevo individuo (hijo) [71].
- Dado que el proceso es iterativo se obtendrán diferentes generaciones de soluciones, donde cada nueva generación de individuos tendrá un mejor valor fitness que la anterior. El proceso termina considerando un criterio de término, como fijar un número de iteraciones o un tiempo máximo de cálculo.

Existe toda una familia de algoritmos evolutivos como el Algoritmo Genético (AG), Programación Genética (PG), Programación Evolutiva (PE), entre otras. Cada uno de estos algoritmos ocupan el mismo principio mencionado anteriormente, sin embargo sus principales diferencias recaen en la estructura de su implementación. Esto conlleva a su vez una preferencia para ocupar un determinado algoritmo en la resolución de problemas específicos.

4.1.1. Ejemplo: algoritmo genético

Los artículos [67], [72], [70], [14] y [73] sirvieron como inspiración para este apartado, se pueden consultar para obtener más información del algoritmo genético y otros algoritmos evolutivos.

Los algoritmos genéticos son probablemente los algoritmos evolutivos más famosos. Fue Jhon Holland quien, en los 70's se dedicaba al estudio de los procesos adaptativos generales, concentrándose en la idea de que un sistema recibe entradas sensoriales del medio ambiente por medio de detectores binarios [70].

Los algoritmos genéticos originales comienzan inicializando una población de soluciones (individuos), su representación se expresa como un bit vectorial [67] de longitud fija l , es decir, $I = \{0, 1\}^l$. Luego, para cada individuo se evalúa una función fitness Φ apropiada para el problema. Para aplicar los algoritmos genéticos canónicos a problemas de optimización de la forma:

$$f : \prod_{i=1}^n [u_i, v_i] \rightarrow \mathbb{R},$$

con $u_i < v_i$. Se debe dividir el bit vectorial en n segmentos, usualmente de misma longitud ($l = n \cdot l_x$), así, cada segmento es interpretado como el código binario de la correspondiente variable $x_i \in [u_i, v_i]$. Cada segmento de bit se puede decodificar de diferentes maneras. Normalmente, una función de decodificación $\Gamma^i : \{0, 1\}^{l_x} \rightarrow [u_i, v_i]$ tiene la siguiente forma:

$$\Gamma^i(a_{i1} \dots a_{il_x}) = u_i + \frac{v_i - u_i}{2^{l_x} - 1} \left(\sum_{j=1}^{l_x} a_{ij} 2^{j-1} \right), \quad (4.1)$$

donde $(a_{i1} \dots a_{il_x})$ denota el i -ésimo segmento de un individuo $\vec{a} = (a_{11} \dots a_{nl_x}) \in I^l$. En términos genéticos, llamamos **cromosomas** a la población de soluciones y podemos asociar cada segmento de bit con el genotipo del individuo y obtener el fenotipo por medio de un mapeo sobre el espacio de parámetros.

Los tres principales operadores en el algoritmo genético involucran selección, combinación y mutación para dar lugar a un nuevo conjunto de soluciones.

La probabilidad de que un cromosoma sea seleccionado para la combinación depende de su fitness $\Phi(\vec{a})$. Una vez seleccionado a los mejores cromosomas, se utiliza el operador de combinación, en el que se intercambian fragmentos de vectores entre cromosomas. El operador de mutación cambia aleatoriamente los valores de los cromosomas. Finalmente, se forma una nueva generación de cromosomas como resultado y el proceso se repite hasta obtener una generación de soluciones óptimas. Esta información se encapsula en el pseudocódigo 2.

Algorithm 2 *Algoritmo genético*

Require: *inicializa:* $t = 0$, $P(0)$, evaluar $P(0)$, función_fitness= Φ .

- 1: **while** not (condición de término) **do**
 - 2: $P'(t)$ =seleccionar($P(t)$) usando Φ
 - 3: $P'(t)$ =aplicar operadores de cruce ($P(t)$)
 - 4: $P(t + 1)$ = reemplazar ($P(t), P'(t)$)
 - 5: evaluar $P(t + 1)$
 - 6: $t = t + 1$
 - 7: **return** Mejor solución encontrada
-

El algoritmo genético es eficaz cuando se tiene poca información del espacio de búsqueda o cuando el espacio es muy complejo, ya que obtiene buenas soluciones en poco tiempo. Si bien este algoritmo es útil cuando no se tiene un análisis matemático del problema, las matemáticas asociadas a este algoritmo pueden llegar a ser complejas (ver [70]).

Un par de inconvenientes del algoritmo genético es que no es directamente adecuado para resolver problemas de optimización con restricciones ya que su propósito inicial fue para búsqueda adaptativa y diseño de sistemas adaptativos [71]. Otro problema es que el algoritmo genético tiende a converger en óptimos locales en lugar del óptimo global si la función fitness no se define apropiadamente.

4.2. Algoritmos de enjambre

El comportamiento de algunos insectos o grupos de animales en la naturaleza, tales como las colonias de hormigas, parvadas de aves, enjambre de abejas o bancos de peces han llamado la atención de varios investigadores

para resolver diferentes problemas en ciencia e ingeniería [69]. Del estudio de estos comportamientos nacen las técnicas de *inteligencia de enjambre*, las cuales se basan en el comportamiento colectivo de esta clase de individuos. En particular, son algunos grupos de insectos, como las hormigas y abejas, quienes pueden resolver de manera flexible y robusta los problemas cotidianos que enfrentan, tales como la búsqueda de comida, construcción de hormigueros o panales y la reparación de estos, haciendo una división de trabajo eficiente [74]. Con esto se asegura cumplir tareas de búsqueda, recolección y comunicación. Otras características representativas de estos algoritmos son la rapidez, posición del individuo, adaptabilidad, escalabilidad y auto organización, ya que cada individuo es responsable de una tarea en particular.

Existen dos familias de algoritmos que destacan dentro de los métodos de inteligencia de enjambre, uno es el algoritmo de optimización por colonia de hormigas (ACO por sus siglas en inglés) y el otro es el algoritmo de optimización por enjambre de partículas (PSO por sus siglas en inglés). La primer familia está inspirada por el rastro de feromonas, siguiendo el comportamiento de especies de hormigas. La otra está inspirada de los rebaños y su comportamiento colectivo [75].

El término “enjambre” acuñe al movimiento irregular de las partículas en el espacio de búsqueda, por lo que es necesario que los métodos de inteligencia de enjambre cumplan ciertos principios (obtenidos de [67]):

1. Principio de proximidad: Los individuos del enjambre deben ser capaces de realizar cálculos simples relacionados con el entorno que los rodea.
2. Principio de calidad: los individuos deben ser capaces de responder a factores de calidad en el entorno, tales como comida y seguridad.
3. Principio de respuesta diversa: los individuos no deben realizar sus actividades en regiones estrechas.
4. Principio de estabilidad y adaptabilidad: los individuos no deben adaptarse cada vez que el entorno cambie ya que su cambio de comportamiento tiene un costo computacional.

La inteligencia de enjambre tiene ventajas sobre los algoritmos evolutivos, ya que su implementación consta de menos parámetros, menos operadores y son más fáciles de implementar [68].

4.2.1. Optimización por colonia de hormigas

En este apartado discutiremos brevemente el algoritmo ACO, ya que representa uno de los algoritmos de enjambre más importantes, para ello ocuparemos como inspiración los siguientes artículos [67], [76] y el libro [77].

Como su nombre lo indica, el algoritmo de optimización por colonia de hormigas está inspirado en el comportamiento que realizan las colonias de hormigas para la búsqueda de alimento, creación y protección de hormigueros. Para lograr esta coordinación colectiva, las hormigas hacen uso de una forma de comunicación indirecta llamada *estigmergia*. Esta forma de comunicación se produce al modificar el entorno dejando rastros de feromonas desde la colonia hasta el lugar donde haya fuentes de alimento.

La importancia de la stigmergia se debe a que las hormigas que recorran caminos más cortos para hallar alimento dejarán un mayor rastro de feromonas debido a que el tiempo de recorrido también es menor. Así, mientras mayor sea el rastro de feromonas, mayor será la elección de las hormigas por seguir ese camino como se puede ver en la imagen 4.1. Esta habilidad hace que las hormigas sean capaces de encontrar siempre el camino más corto entre su colonia y la fuente de alimento, un claro ejemplo de optimización.

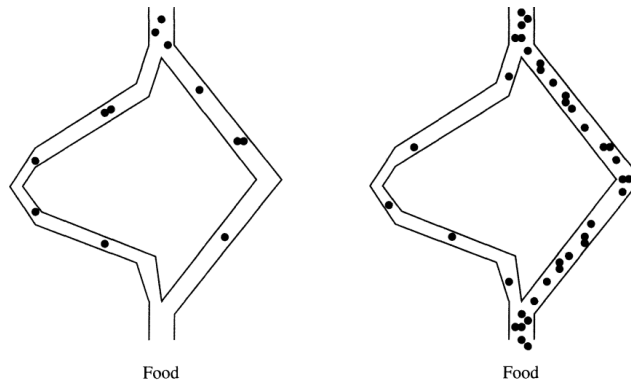


Figura 4.1: Ilustración del camino de feromonas seguido por las hormigas considerando 2 caminos distintos, uno más corto que otro. Obtenido de [77].

Este algoritmo de optimización se estructura en cuatro funciones principales:

- **AntSolutionsConstruct**: esta función construye distintas soluciones, donde cada hormiga se mueve entre diferentes puntos del problema siguiendo reglas de transición, construyendo diferentes soluciones de forma iterativa.

- PheromoneUpdate: esta función actualiza el rastro de feromonas. Se puede realizar una vez que todas las hormigas hayan recorrido todos los puntos o actualizarse en cada recorrido/iteración.
- EvaporationUpdate: se incluye una función de evaporación, la cual se encarga de reducir los rastros de feromonas y evitar que las hormigas se concentren en malas soluciones.
- DeamonAction: es una función opcional, cuyo objetivo es hacer un refuerzo de feromonas en la mejor solución.

El siguiente pseudocódigo obtenido de [78] ejemplifica la implementación del algoritmo ACO clásico. Los parámetros definidos en el pseudocódigo 3 se pueden revisar en [77].

Algorithm 3 *Algoritmo de optimización por colonia de hormigas*

Require: *inicializa:* $t = 0$, número de hormigas (N), parámetros ACO (τ, α, β y η).

```

1: while not (condición de término) do
2:   for  $k=0$  to número de hormigas do
3:     la hormiga  $k$  escoge un nodo de inicio.
4:     while la solución no esté construida do
5:       la hormiga  $k$  selecciona el nodo de mayor probabilidad.
6:     end while
7:   end for
8:   Actualiza el rastro de feromonas.
9: end while

```

5 Optimización por enjambre de partículas

El algoritmo de optimización por enjambre de partículas (PSO) fue propuesto en el año 1995 por Kennedy y Eberhart [79] como un método para optimizar funciones continuas no lineales. Como se ha mencionado anteriormente, este algoritmo es el mayor representante de los algoritmos de enjambre inspirados en rebaños y comportamientos sociales.

Este algoritmo está inspirado principalmente en el comportamiento social que tienen las parvadas de aves para buscar alimento. El vuelo aparentemente aleatorio de las parvadas de aves tiene un propósito importante, cada miembro del grupo puede hacer hallazgos propios y compartirlos con el resto de la parvada. Por ello, en este algoritmo se resaltan los descubrimientos personales y la comunicación entre los miembros de la parvada para lograr rápida y eficientemente un objetivo común, en este caso, encontrar alguna fuente de comida [10].

A diferencia de otros algoritmos bio-inspirados o de algunos otros métodos de optimización, el PSO es un algoritmo que se ha estudiado y aplicado ampliamente debido a su sencilla implementación y comprensión del algoritmo. Por ejemplo, se puede simular con relativa sencillez y precisión el vuelo de una parvada de aves, simplemente manteniendo una distancia objetivo entre cada ave y sus vecinos inmediatos [80]. Por supuesto, en condiciones reales o no simuladas, esto se convierte en un sistema complejo ya que influye el razonamiento personal de cada individuo.

Las diversas aplicaciones del algoritmo de optimización por enjambre de partículas y sus posteriores variantes abarcan diversos campos, tales como el de la inteligencia artificial ya que el PSO puede ser útil para entrenar redes neuronales y apoyar en el diagnóstico de enfermedades degenerativas como el Parkinson [81]. También es útil en área de ingeniería, metalurgia y economía. En física, el algoritmo PSO ha mostrado tener diversas aplicaciones, particularmente en astrofísica y cosmología ([82], [10], [83] y [84]).

5.1. Descripción del algoritmo PSO

Para desarrollar el algoritmo PSO debemos empezar por definir el espacio de búsqueda del enjambre. Se debe comenzar por definir un número de partículas inicial, la posición de cada partícula debe ser inicializada aleatoriamente, en particular para los casos en que el problema no aporte alguna información previa [85]. Debido a que la información debe transmitirse, asignaremos velocidades de desplazamiento a las partículas, dichas velocidades iniciales también se implementan aleatoriamente y se irán ajustando de acuerdo a la dinámica del movimiento de las partículas.

La intención del PSO consiste en encontrar el óptimo global partiendo de una cierta comunicación entre cada una de las partículas generadas, comunicando la posición en la que están y qué tan buena es esta posición. Una forma tradicional de modelar esta red de comunicación entre los individuos es por medio de un grafo, donde cada vértice representa un individuo y el arco que los conecta representa el enlace de información entre los individuos. Así, podemos ver a este conjunto de partículas conectadas por enlaces de información como un grupo de receptores de información [85]. Cabe resaltar que cada partícula puede pertenecer de forma simultánea a diferentes grupos de información, de esta manera, cada partícula se comunicaría con diferentes informantes acerca de los mejores o peores sitios. Por lo tanto, es importante definir un método para que las partículas se desplacen una vez que la información haya sido transmitida.

5.1.1. Dinámica de las partículas

Para que la comunicación entre partículas sea efectiva, es necesario definir una función llamada **función fitness**, que evaluará la calidad de la posición actual donde se encuentre la partícula. Para que las partículas calculen su siguiente desplazamiento, deben ser influenciadas por la mejor posición encontrada hasta el momento por cada una de ellas individualmente, y debido al factor del número de informantes que tiene cada partícula, también influirá la mejor posición encontrada por sus vecinos [85].

Podemos formalizar matemáticamente el propósito del algoritmo PSO (inspirado de [82]) para hallar el mínimo global de una función escalar $f(\vec{x})$, $\vec{x} \in D \subset \mathbb{R}^n$. D representa el espacio de búsqueda y si implementamos N partículas de enjambre, podemos asociar a cada una con la posición que ocupe. Para una partícula arbitraria este vector se expresa como: $\vec{x}_i = (x_{i1}, \dots, x_{in})$, con $i \in \{1, 2, \dots, N\}$, donde N es un parámetro fijo y representa el número de partícula, mientras que n representa la dimensión del espacio.

Para el proceso iterativo del algoritmo, fijamos el número de iteraciones t . Para cambiar la posición de la partícula de una iteración o tiempo $t - 1$ a t añadimos el factor de velocidad $\vec{v}$ a la posición actual que lleve cada partícula, esto es:

$$\vec{x}_i(t) = \vec{x}_i(t - 1) + \vec{v}_i(t). \quad (5.1)$$

Durante cada iteración se evaluará el fitness en la posición actual de cada partícula. También, cada partícula tienen asociado un resultado histórico de la posición donde ha encontrado el mejor valor fitness a lo largo de su trayectoria. A este resultado se le conoce como **personal best** denotado por $pbest$ o $\vec{p}_i(t)$:

$$\min_{j=t, t-1, \dots, 0} f(\vec{x}_i(j)) = f(\vec{p}_i(t)). \quad (5.2)$$

Por otro lado, debido a la interacción colectiva de las partículas se puede obtener un resultado histórico de la mejor posición hallada por todas las partículas en una determinada iteración t . A este resultado se le conoce como **general best** denotado por $gbest$ o $\vec{g}(t)$.

$$\min_{j=1, \dots, N} f(\vec{p}_j(t)) = f(\vec{g}(t)). \quad (5.3)$$

Con estas definiciones podemos escribir la ecuación para actualizar la velocidad e implementarla en (5.1):

$$\begin{aligned} \vec{v}_i(t) = & \omega \vec{v}_i(t - 1) + r_1 c_1 [\vec{p}_i(t - 1) - \vec{x}_i(t - 1)] + \\ & r_2 c_2 [\vec{g}(t - 1) - \vec{x}_i(t - 1)], \end{aligned} \quad (5.4)$$

donde c_1 y c_2 representan coeficientes de aceleración positivos, r_1 y r_2 son valores aleatorios uniformes en el rango $[0, 1]$, de tal manera que el producto $r_i c_i$ sea, en esencia, menor a 1. Mientras que al parámetro ω se le conoce como peso de inercia de las partículas.

Coeficientes de aceleración

Los coeficientes de aceleración c_1 y c_2 , también conocidos como parámetros cognitivo y social respectivamente [80], son factores de peso, es decir que pueden afectar la habilidad de búsqueda de PSO. El parámetro cognitivo se

encarga de controlar cuánta importancia debe tener el resultado de búsqueda individual de la partícula, mientras que el parámetro social se encarga de cuánta importancia debe tener el resultado de la búsqueda de todo el enjambre.

La elección de estos valores dependerá de si se requiere una buena exploración global o una precisa búsqueda local alrededor de las mejores posiciones obtenidas. Para el primer caso se recomiendan valores grandes de c_1 y c_2 , mientras que para el segundo se deben elegir valores pequeños. Para funciones objetivo unimodales convexas es más efectivo definir los coeficientes de aceleración de tal manera que se le dé peso al vector $gbest$, es decir $c_2 > c_1$ y para funciones multimodales más complejas es mejor utilizar una componente cognitiva mayor, $c_1 > c_2$.

Kennedy ha estudiado los efectos de estas variables en las trayectorias de las partículas y concluyo que si $c_1 + c_2 \geq 4$, las velocidades y posiciones van hacia infinito [77].

Peso de inercia

El parámetro conocido como peso de inercia ω controla la influencia de la velocidad previa ($\vec{v}_i(t-1)$) sobre la velocidad actualizada $\vec{v}_i(t)$. El peso de inercia es un parámetro que se debe fijar, se ha demostrado que si el valor de $\omega \geq 1$, la velocidad de las partículas puede aumentar y por lo tanto el enjambre podría diverger. Por otro lado, si $\omega < 1$, la exploración de las partículas podría concentrarse en regiones pequeñas. Se ha demostrado que para asegurar la convergencia (5.1.1) se debe cumplir la siguiente relación:

$$\omega > \frac{1}{2}(c_1 + c_2) - 1, \quad (5.5)$$

ya que según algunos estudios, la convergencia no se asegura para cualquier combinación de estos parámetros, exhibiendo comportamientos cíclicos o divergentes [77].

Velocidad máxima

La velocidad máxima es un parámetro que previene que las partículas se muevan rápidamente a través del espacio de búsqueda e impulsando su divergencia.

$$\vec{v}_i(t) = \begin{cases} \vec{V}_{max}, & \text{si } \vec{v}_i(t) > \vec{V}_{max}, \\ -\vec{V}_{max}, & \text{si } \vec{v}_i(t) < -\vec{V}_{max}. \end{cases}$$

Es común que la velocidad máxima $\vec{V}_{max}$ sea proporcional al rango de la función. Por ejemplo, si el rango es $[-50, 50]$, entonces $\vec{V}_{max} \propto 50$. Sin embargo, se ha demostrado [86] que la implementación de la velocidad máxima no es necesario si se cumple:

$$\vec{v}_i(t) = \kappa \{ \omega \vec{v}_i(t-1) + \rho_1 [\vec{p}_i(t-1) - \vec{x}_i(t-1)] + \rho_2 [\vec{g}(t-1) - \vec{x}_i(t-1)] \}, \quad (5.6)$$

donde $\rho_1 = r_1 c_1$, $\rho_2 = r_2 c_2$ y

$$\kappa = 1 - \frac{1}{\rho} + \frac{\sqrt{|\rho^2 - 4\rho|}}{2}.$$

Con $\rho = \rho_1 + \rho_2 > 4$. Al factor κ se le conoce como coeficiente de constricción.

Convergencia

Para que el algoritmo PSO logre la convergencia en algún punto o el fin de su búsqueda, se debe fijar un número máximo de iteraciones. Otro criterio de término es mediante el cambio en las velocidades de todas las partículas. Si el cambio tiende a cero significa que ninguna partícula está moviéndose más, por lo que se asegura que las partículas han alcanzado algún punto.

5.1.2. Topología de vecindades

El parámetro ω y los coeficientes c_1, c_2 tienen el propósito de garantizar la convergencia del algoritmo PSO si se implementan de acuerdo a los criterios mencionados anteriormente. Sin embargo, esto no es suficiente para incrementar la eficiencia del algoritmo en problemas más complejos donde el espacio de búsqueda sea amplio y multimodal. Este tipo de funciones, al ser multimodales, constan de un gran número de óptimos locales y la exploración del enjambre podría no abarcar todo el espacio y concentrarse únicamente en regiones locales. En este caso, la convergencia se podría dar en una vecindad donde se halle la mejor de todas las posiciones de dicha región. A cada partícula se le asocia un conjunto de partículas vecinas y en cada iteración se comunica la mejor posición sólo a este grupo de partículas vecinas y posteriormente al resto de los conjuntos vecinos [80].

Es importante entonces, discutir la configuración de las vecindades ya que sus características tienen un impacto en la dinámica del enjambre [53].

Concepto de vecindad

Sea x_i la i -ésima partícula del enjambre $S = \{x_1, x_2, \dots, x_N\}$. Se dice que una vecindad de x_i es el subconjunto:

$$B_i = \{x_{n_1}, x_{n_2}, \dots, x_{n_s}\} \subset D,$$

donde $\{n_1, n_2, \dots, n_s\} \subseteq \{1, 2, \dots, N\}$ es el conjunto de índices de las vecindades [80]. El estudio de la topología está fuertemente relacionado con el concepto de vecindad, por lo que al estudio de las vecindades de cada partícula se le conoce como *topología de vecindad*. Las topologías de vecindad pueden determinarse de dos maneras, la primera es por medio de los índices de las partículas, esta forma no requiere de ningún concepto de distancia. La otra se determina de acuerdo con la distancia topológica entre las partículas considerando alguna métrica.

En este apartado nos enfocaremos en tres principales tipos de topologías de vecindad que se han estudiado, todas ellas consideran los índices de las partículas sin ninguna información acerca de las distancias que hay entre ellas para llevar a cabo la comunicación y diversificación del enjambre. Utilizaremos como guía a [77].

- Topología estrella: en este tipo de estructura, todas las partículas están intercomunicadas por lo que es posible que cada una de ellas pueda comunicarse con el resto (ver imagen 5.1). En este caso, cada partícula está atraída hacia la mejor solución encontrada por el resto del enjambre. El algoritmo PSO implementado siguiendo esta topología dio lugar al *gbest* PSO. Es útil y eficaz para problemas unimodales y ha mostrado tener una rápida convergencia en comparación con otras estructuras topológicas. Sin embargo, es susceptible a que las partículas queden atrapadas en mínimo locales.

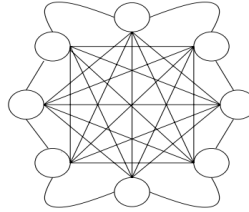


Figura 5.1: Ejemplo de la estructura de topología estrella. Obtenido de [77].

- Topología anillo: cada partícula puede comunicarse con su número n

de vecinos más cercanos. En la imagen 5.2 se tiene como ejemplo el caso $n = 2$, por lo que cada partícula se comunica con sus 2 vecinos inmediatos, esto da lugar a que cada partícula se mueva cerca de la partícula que haya encontrado la mejor posición. La imagen mostrada abajo tiene una forma circular, la ventaja de esta geometría es la de abarcar la mayor cantidad de área a costa de una convergencia lenta. El algoritmo PSO implementado siguiendo esta topología dio lugar al *lbest* PSO.

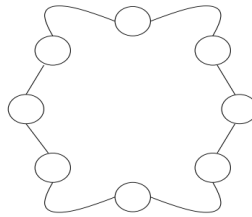


Figura 5.2: Ejemplo de la estructura de topología anillo. Obtenido de [77].

- Topología rueda: la estructura rueda consiste en que una partícula, llamada partícula principal esté conectada al resto, tal como se observa en la imagen 5.3. Esto permite que toda la información pase a través de la partícula principal y lo que hace esta partícula es comparar los resultados del resto de partículas vecinas y modificar su posición hacia una partícula con mejor desempeño. Si la nueva posición de la partícula principal resulta en un mejor fitness se le comunicará al resto de partículas vecinas. La estructura rueda permite una propagación lenta del enjambre.

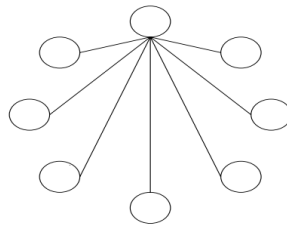


Figura 5.3: Ejemplo de la estructura de topología rueda. Obtenido de [77].

Una de las razones por las que se prefiere utilizar vecindades basadas en índices y no en distancias es que no es computacionalmente costoso. Por

ejemplo, el cálculo de la distancia Euclideana tiene complejidad $O(n^2)$ por lo que el costo computacional incrementa si se quiere implementar para formar vecindades entre todos los pares de partículas. Además de que ayuda a promover la propagación de información acerca de las mejores soluciones a todas las partículas sin importar la posición actual de las partículas [75].

5.1.3. Búsqueda global y local

Global best

Para hallar la mejor posición global (*Global best*), la vecindad de cada partícula es el enjambre entero (topología estrella) y la ecuación de velocidad se sigue de (5.4). Los pasos para implementar este algoritmo pueden ser deducidos de la sección 5.1.1, sin embargo, en este apartado los escribiremos explícitamente:

1. Inicializamos el enjambre de partículas $P(t)$ considerando que la posición inicial $\vec{x}_i(t)$ de cada partícula P_i es aleatoria en el hiperespacio.
2. Evaluamos la función fitness F en cada partícula usando como parámetro la posición actual $\vec{x}_i(t)$ de dicha partícula.
3. Si la función fitness es tal que $F(\vec{x}_i(t)) < pbest_i$ entonces de manera recurrente sucederá que $pbest_i = F(\vec{x}_i(t))$ y la posición pasará a ser: $\vec{p}_i = \vec{x}_i(t)$
4. Comparamos el fitness de cada partícula con el promedio del mejor valor anterior de la población. Si el valor actual es mejor que el $gbest$, se reajusta el $gbest$ al valor del fitness de la partícula actual [77]. Esto es, $F(\vec{x}_i(t)) = gbest$ y la posición $\vec{g} = \vec{x}_i(t)$.
5. Cambiamos el vector velocidad para cada partícula de acuerdo a la ecuación:

$$\begin{aligned} \vec{v}_i(t) = & \vec{v}_i(t-1) + \rho_1(\vec{p}_i - \vec{x}_i(t)) \\ & + \rho_2(\vec{g} - \vec{x}_i(t)). \end{aligned} \quad (5.7)$$

6. Movemos cada partícula a una nueva posición usando la ecuación (5.1), con el cambio $t = t + 1$.
7. Se regresa al paso 2 y se repite el proceso hasta obtener la convergencia.

Algorithm 4 *Algoritmo Global Best, obtenido de [75].*

Require: *inicializa:* Tamaño de la población n . Número máximo de iteraciones max_iter .

```

1: while not (condición de término) do
2:   for  $i = (1 \text{ to } n)$  do
3:      $\vec{x}_i \leftarrow$  inicializa la posición de la  $i$ -ésima partícula.
4:     if  $f(\vec{x}_i) < f(\vec{p}_i)$  then
5:        $\vec{p}_i = \vec{x}_i$ 
6:     end if
7:     if  $f(\vec{p}_i) < f(\vec{g})$  then
8:        $\vec{p}_i = \vec{g} \leftarrow$  establecer la mejor posición global.
9:     end if
10:  end for
11:  for  $i = (1 \text{ to } n)$  do
12:     $\vec{v}_i \leftarrow$  inicializa la velocidad de la  $i$ -ésima partícula usando 5.4.
13:     $\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i$ 
14:  end for
15: until se cumpla la condición de término

```

Local Best

Para la búsqueda local del algoritmo PSO (*Local best*) se definen pequeñas vecindades para cada partícula (topología anillo), intercambiando información dentro de la vecindad, que se refleja en un conocimiento local del espacio [75]. A diferencia de la búsqueda global, la ecuación de velocidad para este algoritmo es:

$$\vec{v}_i(t) = \omega \vec{v}_i(t-1) + r_1 c_1 [\vec{p}_i(t-1) - \vec{x}_i(t-1)] + r_2 c_2 [\vec{g}_i(t-1) - \vec{x}_i(t-1)], \quad (5.8)$$

donde el factor $\vec{g}_i$ representa la mejor posición encontrada por la vecindad de la i -ésima partícula. Es decir, la diferencia que hay entre las ecuaciones de velocidad para la búsqueda global (5.4) y local (5.8), está en que la primera no requiere del subíndice i en $\vec{g}$ mientras que la otra sí. La mejor posición $\vec{g}_i$ de una vecindad B_i se define como [75]:

$$\vec{g}_i(t) \in \{B_i | f(\vec{g}_i(t)) = \min\{f(\vec{x})\} \cdot \forall \vec{x} \in B_i\}.$$

La diferencia principal entre el algoritmo Global best y el algoritmo Local best está en la manera en que cada partícula se relaciona con el enjambre.

En el global best, cada partícula compara su fitness con el mejor fitness hallado por alguna partícula del enjambre, promoviendo la búsqueda cercana a esa partícula y por lo tanto, a ese valor. En el local best, cada partícula perteneciente a una vecindad compara su fitness con el resto de partículas vecinas, esto da lugar a que en la implementación de este algoritmo se deba inicializar un número k de vecindades, que en su totalidad contengan a todas las partículas, por lo que k debe ser menor que el número de partículas iniciadas [87].

Los pasos para la implementación de este algoritmo se siguen de los mostrados en 5.1.3, con la diferencia de que en los pasos 4 y 5 cambiemos $gbest$ por $lbest$ y $\vec{g}$ por $\vec{g}_i$.

Algorithm 5 *Algoritmo Local Best, obtenido de [75].*

Require: *inicializa:* Tamaño de la población n . Número de vecindades κ .
Número máximo de iteraciones max_iter .

```

1: while not (condición de término) do
2:   for  $i = (1 \text{ to } n)$  do
3:      $\vec{x}_i \leftarrow$  inicializa la posición de la  $i$ -ésima partícula.
4:     if  $f(\vec{x}_i) < f(\vec{p}_i)$  then
5:        $\vec{p}_i = \vec{x}_i$ 
6:     end if
7:     if  $f(\vec{p}_i) < f(\vec{g}_i)$  then
8:        $\vec{p}_i = \vec{g}_i \leftarrow$  establecer la mejor posición de la vecindad.
9:     end if
10:  end for
11:  for  $i = (1 \text{ to } n)$  do
12:     $\vec{v}_i \leftarrow$  inicializa la velocidad de la  $i$ -ésima partícula usando 5.8.
13:     $\vec{x}_i(t) = \vec{x}_i(t - 1) + \vec{v}_i$ 
14:  end for
15: until se cumpla la condición de término

```

Mencionamos que tanto para la búsqueda local como la búsqueda global, la implementación de topología de vecindades es por medio de los índices de partículas. Sin embargo, no hay que confundir este concepto con la vecindad de una partícula, las topologías de vecindad definen una estructura para que la comunicación de las partículas pueda diversificarse mientras que la vecindad de una partícula se puede definir de la siguiente manera: una partícula P_a pertenece a la vecindad de una partícula P_b si

$$\|\vec{x}_a - \vec{x}_b\| < \epsilon, \quad (5.9)$$

donde $\vec{x}_a$ y $\vec{x}_b$ representan las posiciones de las partículas y ϵ representa un número apropiado mayor que cero. En algunos trabajos, como el de Suganthan [88], la elección de ϵ se considera a partir de:

$$\epsilon = \frac{3t + 0,6t_{max}}{t_{max}}.$$

Considerando a t_{max} como el número máximo de iteraciones. De esta manera, el tamaño de las vecindades incrementan conforme aumenta el número de iteraciones hasta que cada vecindad contenga a todas las partículas del enjambre.

La ecuación (5.9) muestra la utilidad de implementar una métrica que calcule la distancia que hay entre una partícula y los vecinos más cercanos a ella para definir la estructura de su comunicación.

5.1.4. ¿Global best o Local best?

Hacer una comparación entre las dos clases de algoritmos PSO, es hacer una comparación entre las topologías que ambas requieren para transmitir la información de las partículas. En el artículo [89] se hace una revisión de distintos artículos que pusieron a prueba a los algoritmos PSO Global best y PSO Local best a diferentes problemas de optimización, en su mayoría se concluye que el algoritmo Global best converge prematuramente en óptimos que no son los deseados. Se sugiere que la razón de esto es porque la comunicación entre las partículas se da tan rápido, que suelen atraerse hacia una única buena posición. A diferencia del algoritmo Local best, cuya convergencia es más lenta a costa de una mayor exploración del espacio.

Sin embargo, en ese mismo artículo se hace un análisis empírico entre estos dos algoritmos, comparando cada uno de ellos con diferentes funciones test y concluyendo que Global best puede converger más lentamente que Local best para menos de la mitad de las funciones test. Acerca de la precisión de soluciones, Global best proporcionó soluciones más precisas para tantas funciones como hizo Local best. Por estas razones se considera que ninguno de los algoritmos podría considerarse mejor que otro.

5.2. Implementación de PSO en Python

Actualmente existen variaciones y modificaciones del algoritmo PSO, esto conlleva que existan diferentes formas de implementarlo en distintos lengua-

jes de programación. Sin embargo, en este trabajo utilizaremos una librería de Python llamada PySwarms, cuya documentación puede revisarse en [87]. Utilizar PySwarms facilita la implementación del algoritmo PSO, ya que cuenta con módulos para la visualización de funciones en 2 y 3 dimensiones, también cuenta con diferentes topologías de vecindad (mencionadas en 5.1.2) y la visualización del historial de coste (*cost history*) el cuál representa la convergencia de las partículas hacia el óptimo a lo largo de las iteraciones. Otra característica importante es que el algoritmo PSO implementado en PySwarms se puede poner a prueba por medio de las *funciones test* multidimensionales, sin embargo, se puede modificar esta librería para definir funciones test unidimensionales.

5.2.1. Funciones test

Debido a la necesidad de aplicar diferentes métodos matemáticos o algoritmos encargados de encontrar el óptimo global de funciones objetivo, es importante considerar los diferentes retos que deben superar estos métodos de optimización. En general, en problemas del mundo real, los problemas de optimización son multimodales, esto quiere decir que en una región dada de la función pueden existir muchos óptimos locales. Las funciones multimodales funcionan como un filtro para varios algoritmos de optimización ya que varios de ellos fracasan al momento de buscar un óptimo global, y es por esa razón que se consideran como *funciones test*, sin embargo las funciones test pueden tener otras características como no suavidad y discontinuidades.

Este tipo de funciones marcan una bandera roja o verde para muchos métodos de optimización debido a que varios de ellos no son capaces de lograr resolver el problema de manera eficaz. Por lo tanto, considerando las características necesarias que deben afrontar los algoritmos de optimización, se clasificaron ciertas funciones de optimización según sus características y formas específicas [90]. Las funciones test han sido ampliamente estudiadas, por lo que son bien conocidos el valor del óptimo global y la posición de este. En esta sección pondremos a prueba al algoritmo PSO con funciones test 1-dimensional y 2-dimensional.

Funciones unidimensionales

Primeramente, utilizaremos PySwarms para optimizar cuatro funciones unidimensionales usando el algoritmo Global Best, posteriormente haremos lo mismo usando el algoritmo Local Best y compararemos los resultados de la búsqueda. La elección de la topología se escogerá de acuerdo a lo mencionado

en la sección 5.1.2, donde se recomienda usar la topología estrella para Global Best y la topología anillo para Local Best, de hecho PySwarms considera esto en la implementación de cada uno de los algoritmos. Por otro lado, la elección de los parámetros del algoritmo se seguirán de los criterios mencionados en la sección 5.1.1.

A continuación definimos las cuatro funciones test unidimensionales y sus gráficas se muestran en la imagen 5.4.

■

$$f_1(x) = x^2.$$

Con óptimo global en $x^* = 0$ y valor $f_1(x^*) = 0$.

■

$$f_2(x) = \sin(x) + \sin((10/3)x).$$

Con óptimo global en $x^* = 5,14735$ y valor $f_2(x^*) = -1,9$.

■

$$f_3(x) = \begin{cases} x^2, & \text{si } x \geq 1, \\ 2 - x, & \text{si } x < 1. \end{cases}$$

Con óptimo global en $x^* = 1$ y valor $f_3(x^*) = 1$.

■

$$f_4(x) = (x + (\text{randn} \cdot 0,3x))^2.$$

Con óptimo global en $x^* = 0$ y valor $f_4(x^*) = 0$.

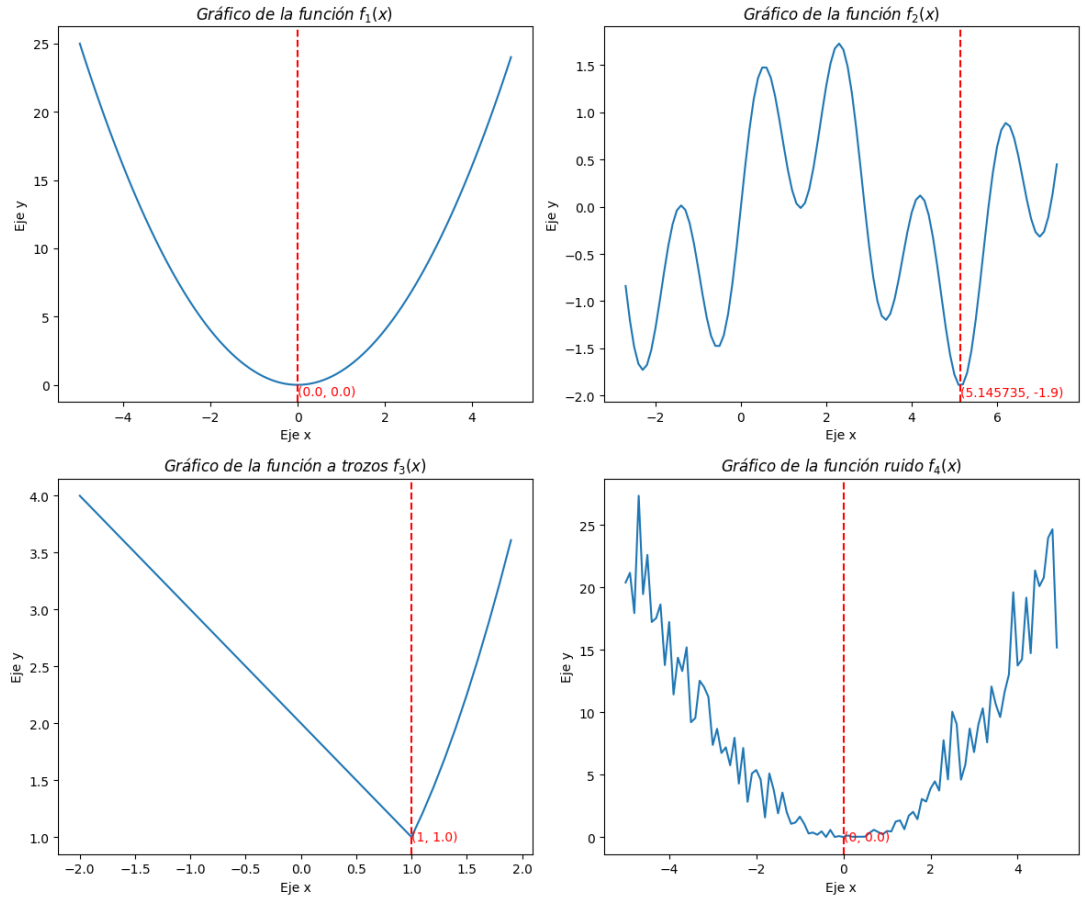


Figura 5.4: Funciones test unidimensionales. La recta vertical roja señala la posición donde se encuentra el óptimo global.

Resultados Global best y Local best para funciones unidimensionales

En las tablas 5.1 y 5.2 mostraremos respectivamente los parámetros definidos para la implementación Global best y Local best de PySwarms, y en la tabla 5.3 mostramos los resultados obtenidos de cada algoritmo.

Para el algoritmo Local best, PySwarms tiene la opción de escoger un tipo de distancia, a saber, la distancia valor absoluto o la distancia Euclideana. Por lo tanto, el algoritmo Local best requiere de 2 parámetros extra que el Global best, estos son: k que representa el número de vecindades y p , que representa el tipo de distancia ($p = 1$ para valor absoluto y $p = 2$ para

Global best. Topología estrella.

Función	c_1	c_2	ω	Iteraciones	Partículas
$f_1(x)$	0.3	0.5	0.4	20	10
$f_2(x)$	0.7	0.3	0.9	30	90
$f_3(x)$	0.2	0.7	0.7	20	25
$f_4(x)$	0.5	0.5	0.9	15	55

Cuadro 5.1: Parámetros algoritmo Global best para optimizar funciones test 1D.

Local best. Topología anillo.

Función	c_1	c_2	ω	Iteraciones	Partículas	Vecindades (k)	Distancia (p)
$f_1(x)$	0.2	0.5	0.5	30	20	5	Euclideana
$f_2(x)$	0.8	0.3	0.5	112	20	10	Euclideana
$f_3(x)$	0.4	0.9	0.7	33	10	7	Valor absoluto
$f_4(x)$	0.8	0.3	0.9	50	10	5	Valor absoluto

Cuadro 5.2: Parámetros algoritmo Local best para optimizar funciones test 1D.

distancia Euclideana).

Los resultados mostrados en la tabla 5.3, nos muestran que el algoritmo PSO es eficaz en hallar el óptimo global para funciones suaves y convexas como f_1 , así como de funciones multimodales (f_2), no suaves (f_3) y una combinación de ambas (f_4). Además, se puede apreciar que, al menos para esta clase de funciones, no hay gran diferencia entre usar los algoritmos Global best o Local best, ya que los resultados son muy aproximados entre sí, y en algunos casos ambos algoritmos hallan el mismo fitness y la misma posición de este. La diferencia principal entre estos algoritmos radica principalmente en la elección de iteraciones y número de partículas. Por ejemplo, el algoritmo Local best requiere de un menor número de partículas, pero de más iteraciones a diferencia del Global best, esto va de la mano con su convergencia, debido a la estructura topológica que define este algoritmo, sea más lenta.

Una comparación visual entre estos dos algoritmos se muestra en la figura 5.5, donde se graficaron los historiales de costo utilizando el módulo `cost_history` de Pyswarms.

Resultados Global best vs Local best.

Función	Algoritmo	Mejor posición	Mejor fitness
$f_1(x)$	Global best	-6.0493×10^{-6}	3.6594×10^{-11}
	Local best	6.1944×10^{-6}	3.8371×10^{-11}
$f_2(x)$	Global best	5.1461	-1.8996
	Local best	5.1457	-1.8996
$f_3(x)$	Global best	0.9999	1.0000
	Local best	0.9999	1.0000
$f_4(x)$	Global best	0.1293	7.9225×10^{-7}
	Local best	0.0789	2.07564×10^{-6}

Cuadro 5.3: Resultados obtenidos del algoritmo Global best y Local best implementados en PySwarms para las 4 funciones test 1D.

Funciones multidimensionales

Una vez probado el algoritmo PSO en funciones unidimensionales es natural probar dicho algoritmo en funciones de prueba de más dimensiones. Este tipo de funciones son por lo general mayormente utilizadas para probar los diversos algoritmos de optimización debido a que su complejidad es mucho mayor que con las funciones test unidimensionales por lo que, si un algoritmo es capaz de encontrar el óptimo de una o más funciones de este tipo, sería un buen candidato para usarlo en problemas de optimización aplicados a problemas reales.

Particularmente, las funciones test 2-dimensionales también tienen un carácter visual muy importante, ya que al ser funciones del tipo $z = f(x, y)$, sus gráficas se representan como superficies en un espacio 3-Dimensional y de esta manera es visualmente intuitivo encontrar el óptimo global, e incluso representarlas como curvas de nivel o mapas de calor, lo que ayudaría a representar los óptimos locales y/o globales como colinas o valles.

Para esta clase de funciones también haremos una comparación entre los algoritmos Global best y Local best PSO, con el fin de intentar hallar una diferencia aún más significativa en cuanto a precisión y convergencia de los dos algoritmos, y con ello obtener nuestra propia conclusión en cuanto a qué algoritmo puede ser el más adecuado para nuestro propósito, el cuál es minimizar la χ^2 de diferentes modelos cosmológicos.

Escogeremos 3 funciones multidimensionales de diferente complejidad cada una (ver imagen 5.6), las cuales son:

■

$$f_5(\mathbf{x}) = x^2 + y^2.$$

Con óptimo global en $\mathbf{x}^* = (0, 0)$ y valor $f_5(\mathbf{x}^*) = 0$.

■

$$f_6(\mathbf{x}) = -\cos(x) \cdot \cos(y) \exp(-(x - \pi)^2 - (y - \pi)^2).$$

Con óptimo global en $\mathbf{x}^* = (\pi, \pi)$ y valor $f_6(\mathbf{x}^*) = -1$.

■

$$f_7(\mathbf{x}) = 20 + (x^2 - 10 \cos(2\pi x) + y^2 - 10 \cos(2\pi y)).$$

Con óptimo global en $\mathbf{x}^* = (0, 0)$ y valor $f_7(\mathbf{x}^*) = 0$.

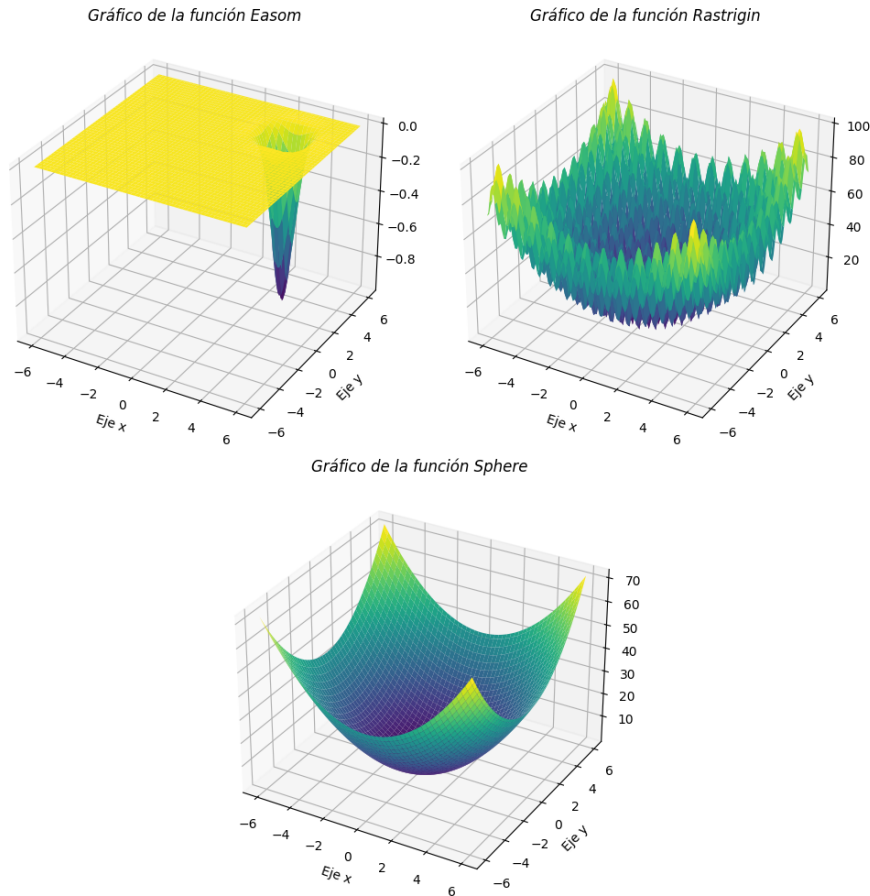


Figura 5.6: Funciones test 2-dimensionales.

Global best. Topología estrella.

Función	c_1	c_2	ω	Iteraciones	Partículas
Sphere	0.3	0.6	0.7	60	30
Easom	0.3	0.9	0.9	73	20
Rastrigin	0.8	0.4	0.9	150	50

Cuadro 5.4: Parámetros del algoritmo Global best para optimizar funciones test 2D.

Local best. Topología anillo.

Función	c_1	c_2	ω	Iteraciones	Partículas	Vecindades (k)	Distancia (p)
Sphere	0.3	0.6	0.7	30	20	5	Euclideana
Easom	0.5	0.8	0.7	45	30	9	Valor absoluto
Rastrigin	0.8	0.4	0.7	200	45	15	Euclideana

Cuadro 5.5: Parámetros del algoritmo Local best para optimizar funciones 2D.

Resultados Global best y Local best para funciones 2-dimensionales

Al igual que en el caso de las funciones test unidimensionales, mostraremos en las tablas 5.4 y 5.5 los parámetros utilizados para cada función y algoritmo respectivamente, posteriormente, en la tabla 5.6 mostraremos el resultado de la mejor posición y mejor fitness obtenido.

La tabla 5.6 muestra, de igual manera que el algoritmo PSO es eficaz en hallar el óptimo global de cada una de las funciones test 2-dimensiones. Puede notarse que para una función unimodal convexa, como la función Sphere,

Resultados Global best vs Local best.

Función	Algoritmo	Mejor posición	Mejor fitness
Sphere	Gbest	$(5.5544 \times 10^{-7}, 6.3143 \times 10^{-7})$	7.0611×10^{-13}
	Lbest	(0.0003,-0.0003)	1.9899×10^{-7}
Easom	Gbest	(3.1441,3.1416)	-0.9999
	Lbest	(3.1416,3.1415)	-0.9999
Rastrigin	Gbest	$(-4.3188 \times 10^{-6}, 3.0282 \times 10^{-5})$	1.8563×10^{-7}
	Lbest	$(-3.2772 \times 10^{-9}, 6.3069 \times 10^{-9})$	0

Cuadro 5.6: Resultados obtenidos del algoritmo Global best y Local best para funciones 2D utilizando PySwarms.

la precisión es mayor utilizando Global best, mientras que para una función multimodal como la función Rastrigin se tiene una mejor precisión utilizando Local best. Respecto a las posiciones halladas tanto en x como en y , también se tiene una mayor precisión en la función Sphere utilizando Global best, mientras que para el resto de funciones es mejor con Local best. Aunque en algunos casos, la precisión del resultado de mejor fitness difiere en algunos órdenes de magnitud, el resultado general sigue siendo bastante óptimo a lo requerido, por ejemplo, si el mejor fitness para la función Rastrigin $f_7(\mathbf{x}^*)$ es igual a cero, y el algoritmo Local best fue capaz de hallar este valor pero Global best halló un valor proporcional a 10^{-7} , podríamos considerar ambos resultados como válidos y tener en cuenta entonces otros factores como el tiempo de convergencia para escoger qué algoritmo conviene utilizar. Por supuesto, hay problemas donde se requiere una mayor precisión, es decir, que los órdenes de magnitud del fitness sea mayor que en otros problemas donde puede que un orden de magnitud pequeño sea suficiente, por lo que la elección del algoritmo PSO Global best o Local best dependerá de las características del problema.

También podemos destacar que el número de iteraciones y partículas implementadas son mayores para esta clase de funciones que para las funciones unidimensionales, sin embargo, no se presentó ninguna complicación durante la ejecución de PySwarms para la optimización de funciones 2D. A continuación presentamos una comparación visual (ver imagen 5.8) de los historiales de costo de cada algoritmo para cada función. Como mencionamos anteriormente, PySwarms permite visualizar el movimiento de las partículas sobre la función objetivo, en este caso, la imagen 5.7 ejemplifica esto, mostrando el movimiento de las partículas sobre la función Rastrigin.

La manera en que escogimos los parámetros de diseño para hallar los óptimos globales de las funciones test se siguieron de acuerdo a la forma de su gráfica. Esto nos dio una idea de cómo debería ser c_1 respecto a c_2 y viceversa. Por otro lado, la elección de w también está inspirada de su gráfica, por ejemplo, para funciones complejas, el valor de este valor era más grande que para funciones más sencillas. Para el caso particular del algoritmo Local best, se siguieron los mismos criterios de los parámetros c_1 , c_2 y ω , sin embargo, la elección del número de vecindades fue tal, que sea menor al número de partículas. Estas funciones nos ayudan a explorar una manera de elegir los parámetros de diseño con base en sus características geométricas, y como resultado de ello, obtuvimos valores precisos del óptimo teórico que tiene cada función. Notamos también que, asumiendo un número de partículas e iteraciones adecuados al problema, y mientras se cumplan las condiciones que tienen c_1 y c_2 para cada función, los valores del óptimo

no divergirán de la vecindad donde se encuentra el óptimo global teórico. Lo que podría cambiar en este caso son otros aspectos como el tiempo de convergencia. Esto significa que dados otros valores de c_1 o c_2 , los valores hallados deben ser cercanos al óptimo.

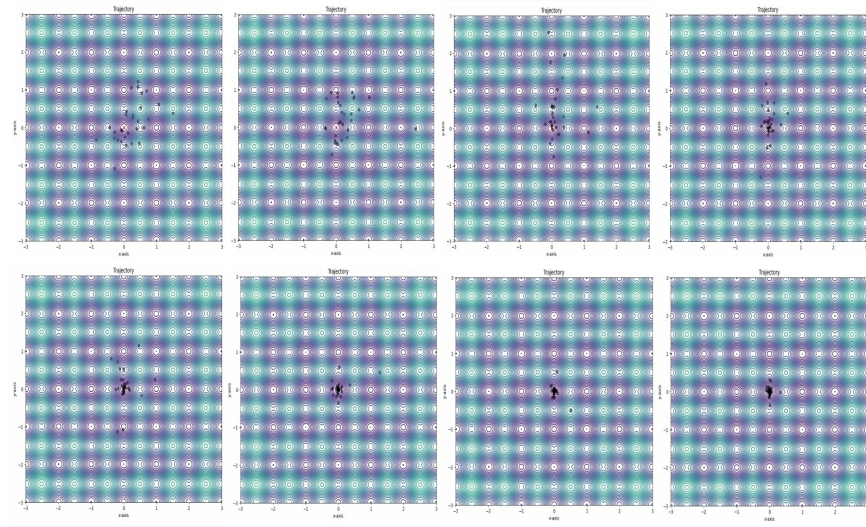


Figura 5.7: De izquierda a derecha: partículas de enjambre moviéndose a través de la función Rastrigin hasta alcanzar el valor de mejor fitness marcado con una X.

5.2.2. Datos sintéticos para una recta

Ya que hemos aplicado el algoritmo PSO a funciones de una y más dimensiones, procedemos a aplicarlo a un conjunto de datos sintéticos, es decir, datos que podemos generar nosotros mismos y que pueden imitar ciertos comportamientos. En este caso, generamos una recta ($y = mx + b$) de 30 puntos, y a cada punto le asociamos ruido con el fin de emular datos dispersados. El propósito de este test es minimizar la función χ^2 de la recta para hallar los valores de la pendiente m y la ordenada al origen b que logren el mejor ajuste de los datos.

La función χ^2 en este ejemplo se expresa como sigue:

$$\chi^2 = \sum_{i=1}^{30} \frac{[(mx + b)_{\text{teo}} - y_{\text{data}_i}]^2}{\sigma_i^2},$$

donde σ representa la desviación estándar, en este caso, el ruido asociado a los datos, y_{data} representa el conjunto de 30 puntos de los datos sintéticos y la expresión $(mx + b)_{\text{teo}}$ representa la recta teórica, es decir, nuestro modelo. La función χ^2 depende de los parámetros m y b , por lo que al minimizar esta función, podremos hallar los valores de m y b donde se halle el mínimo y con ellos, los valores más óptimos de los parámetros para obtener el mejor ajuste.

Para inicializar con los datos sintéticos, generamos una recta de 30 puntos, cuya ecuación es de la forma $y = 2x + 7$, donde $m = 2$ y $b = 7$. Sin embargo, para que esta recta pueda considerarse como datos sintéticos, necesitamos generar ruido aleatoriamente a cada punto de la recta. El resultado de esto se puede ver en la imagen 5.9 junto con las barras de error en cada punto.

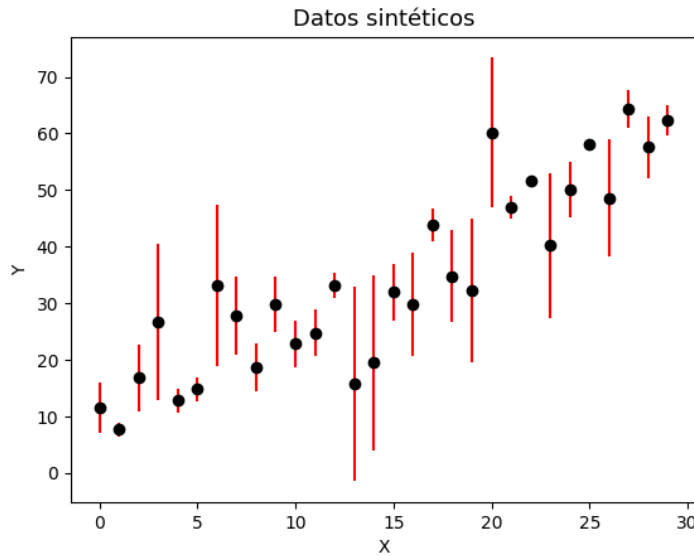


Figura 5.9: Datos sintéticos de una recta, con errores asociados a cada punto.

Debido al ruido asociado a los datos, la recta que mejor se ajuste a ellos ya no será la implementada originalmente, por lo que procedemos a utilizar PySwarms para optimizar la función (5.2.2) y hallar los valores óptimos de m y b . Para llevar a cabo la optimización, utilizaremos el algoritmo Global best, ya que de los resultados anteriores vemos que la diferencia entre este algoritmo y el algoritmo Local best ha sido, por el momento, insignificante.

A diferencia de los ejemplos anteriores, en este caso (y en adelante) consideraremos un parámetro extra en el algoritmo PSO que también se encuentra implementado en PySwarms. Este nuevo parámetro conocido como *bounds* se encarga de acotar el espacio de búsqueda.

Para tener una idea de cómo elegir los parámetros del algoritmo PSO, podemos graficar la función χ^2 y saber si es unimodal, multimodal, o si tiene alguna otra característica que nos de información para la elección adecuada de parámetros. Lo único que sabemos por el momento es que la función es continua y 2-dimensional.

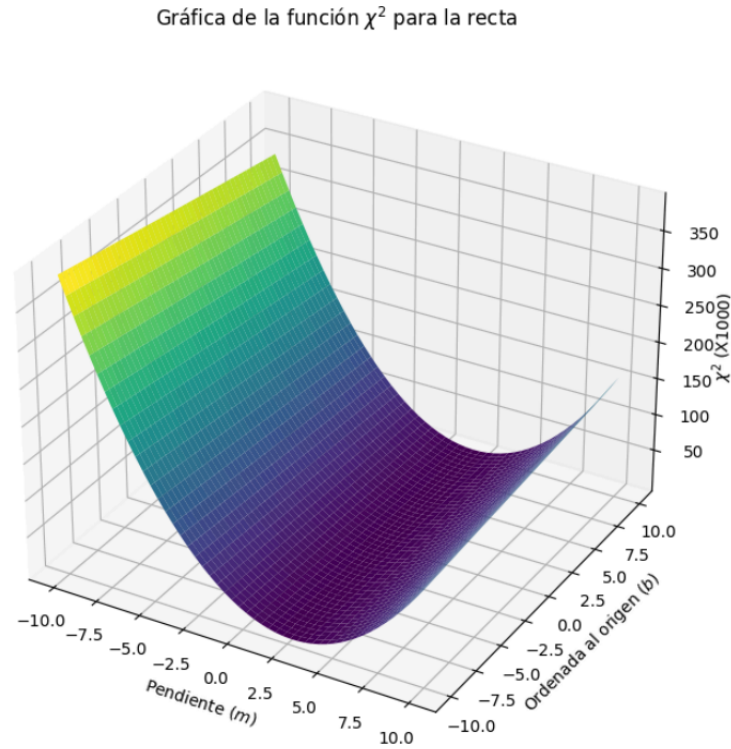


Figura 5.10: Gráfico de la función 5.2.2, en el espacio de parámetros.

La figura 5.10 nos ayuda a visualizar que la función es unimodal, por lo que, en primera instancia podríamos utilizar unos coeficientes de aceleración similares a los de la función Sphere.

En la tabla 5.7 se muestran los parámetros implementados en PySwarms para optimizar la función (5.2.2).

Como resultado, obtenemos que las mejores posiciones donde se halla el

Parámetros del algoritmo PSO para la función χ^2 .

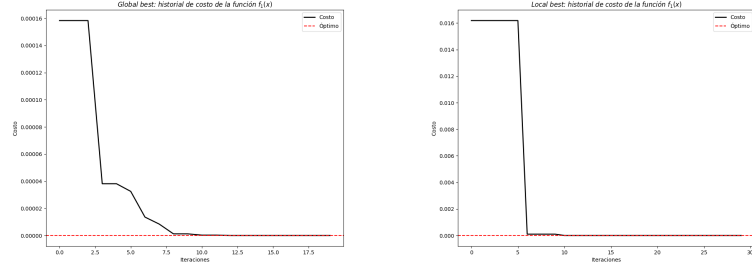
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
χ^2	0.4	0.6	0.9	70	100	$[-10, 10] \times [-10, 10]$

Cuadro 5.7: Parámetros del algoritmo Global best PSO, para optimizar la función χ^2 .

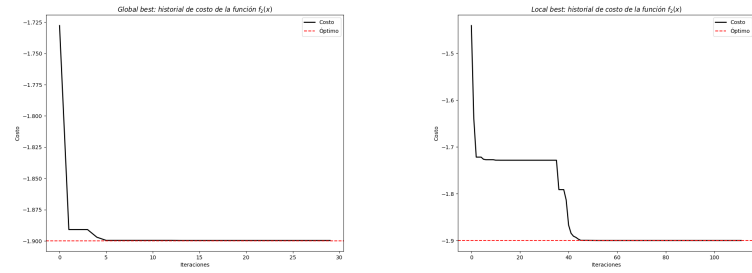
mejor fitness son: $(m, b) = (2.06, 6.06)$. El mejor fitness, es: $\chi^2_{\min} = 28.60$. Con estos resultados podemos mostrar una vez más el historial de costo y también, sustituyendo los resultados hallados de m y b , podemos graficar la recta $y = 2.06x + 6.06$. Estas gráficas se muestran en la figura 5.11. Si bien la función χ^2 es unimodal, requiere de un mayor número de iteraciones y partículas que la función Sphere y Easom, esto puede deberse a que consideramos un rango de búsqueda más amplio. Por otro lado, también vemos que los parámetros de aceleración y el peso de inercia no difieren significativamente a las funciones anteriores.

Este test nos permite tener una idea más clara de cómo será el proceso para optimizar la función χ^2 con los datos cosmológicos mencionados en la sección 2.8. Además, los resultados de las diferentes pruebas realizadas en esta sección nos podrán ayudar bastante en la elección de los parámetros del algoritmo PSO. Hasta el momento, algo con lo que podemos concluir para la siguiente sección es que tanto el algoritmo Global best PSO como el algoritmo Local best PSO son eficaces en minimizar las funciones, por lo que utilizaremos en principio el algoritmo Global best, y especificaremos con base en nuestro análisis, si es necesario el uso del algoritmo Local best.

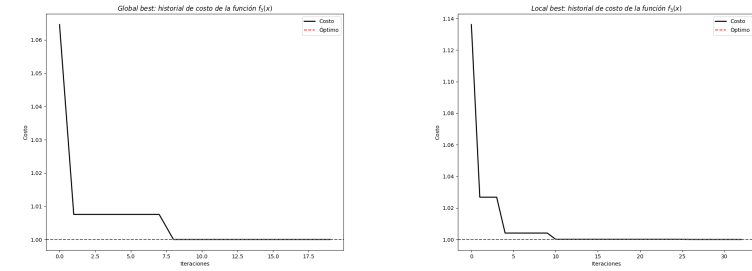
En el ejemplo de los datos sintéticos, al igual que en las funciones test analíticas, nos fiamos de la gráfica de la función χ^2 para la elección de parámetros, pero al tener datos dispersos, el número de partículas implementadas fue mayor que en el caso de las funciones 2-dimensionales. La elección de c_1 y c_2 , se escogió de tal manera que dada la condición $c_2 > c_1$ (para funciones unimodales), no arroje valores que diverjan del óptimo. Los coeficientes de aceleración específicos mostrados en la tabla 5.7 fueron con los que se halló la χ^2 más pequeña, sin embargo, otra elección de estos parámetros, considerando el mismo número de iteraciones, partículas y peso de inercia, dan resultados similares.



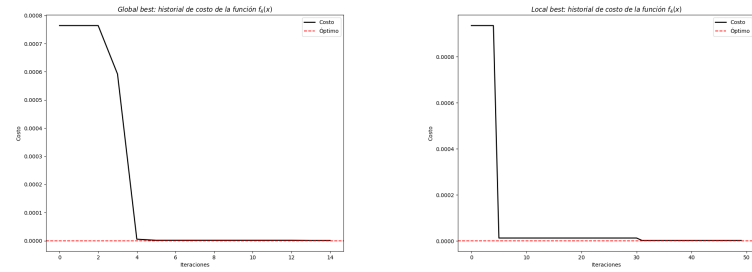
(a) Historial de costo Gbest para f_1 . (b) Historial de costo Lbest para f_1 .



(c) Historial de costo Gbest para f_2 . (d) Historial de costo Lbest para f_2 .

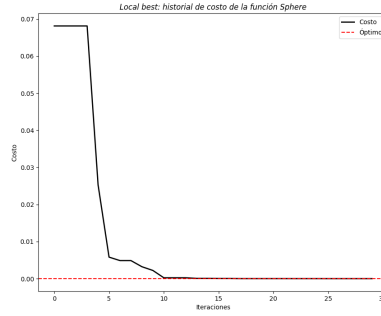
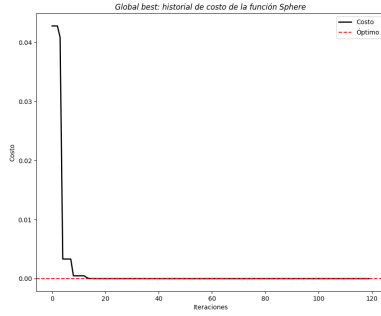


(e) Historial de costo Gbest para f_3 (f) Historial de costo Lbest para f_3 .



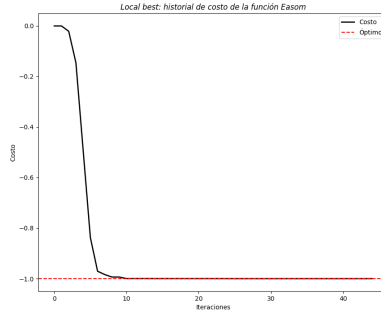
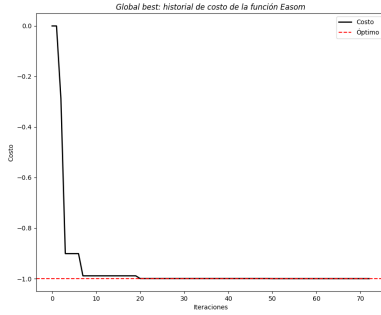
(g) Historial de costo Gbest para f_4 (h) Historial de costo Lbest para f_4 .

Figura 5.5: Comparación de los historiales de costo para funciones test uni-dimensionales.



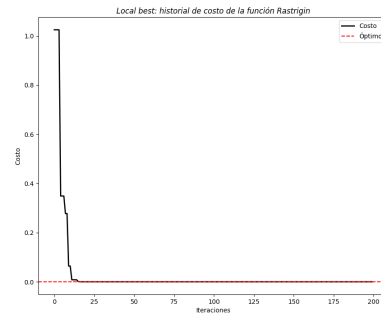
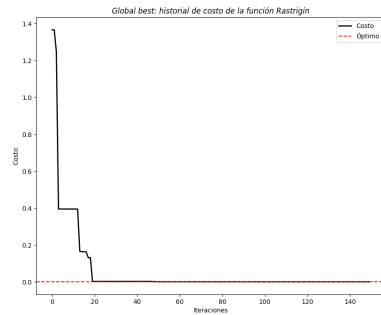
(a) Historial de costo Gbest para Sphere.

(b) Historial de costo Lbest para Sphere.



(c) Historial de costo Gbest para Easom.

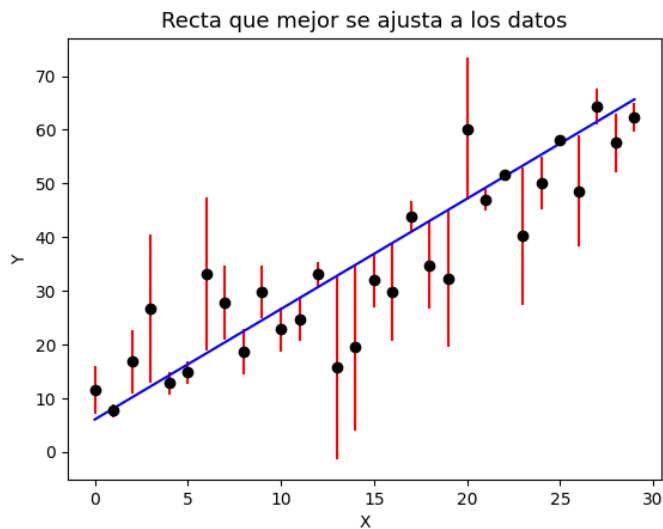
(d) Historial de costo Lbest para Easom.



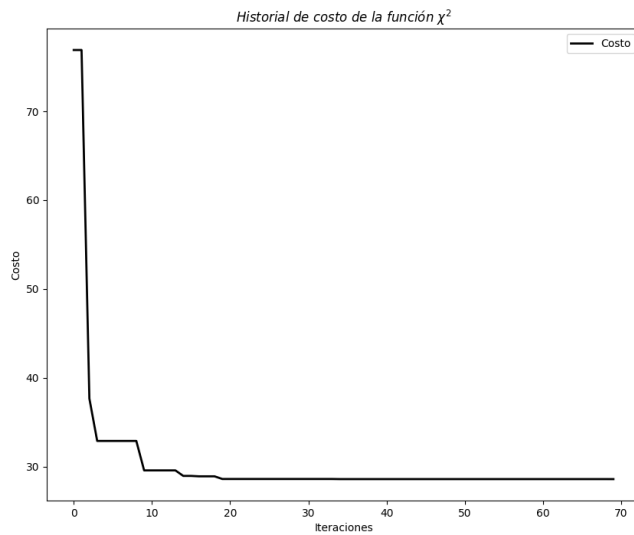
(e) Historial de costo Gbest para Rastrigin.

(f) Historial de costo Lbest para Rastrigin

Figura 5.8: Comparación de los historiales de costo para funciones test 2-dimensionales.



(a) Recta que mejor se ajusta a los datos sintéticos utilizando los resultados de PSO, $m=2.06$ y $b=6.06$.



(b) Historial de costo de la función χ^2 a lo largo de 70 iteraciones.

Figura 5.11: Resultados de la optimización.

6 Resultados

Los resultados de este trabajo se pueden dividir en dos categorías: resultados de la estimación de parámetros para cada modelo y resultados del mejor modelo con base en los criterios AIC y BIC. A su vez, los resultados de la estimación de parámetros se obtienen de cada uno de los conjuntos de datos utilizados y para cada modelo. Iniciaremos este capítulo con los datos de Cronómetros cósmicos debido a su relativa sencillez, ya que estos datos relacionan el parámetro de Hubble $H(z)$ con z de manera directa, por lo que la función χ^2 a optimizar se puede definir fácilmente. Posteriormente, usaremos los datos de Supernovas que requieren de un tratamiento ligeramente más complicado al momento de definir la función χ^2 debido a la integral de la distancia de luminosidad D_L .

6.0.1. Datos de cronómetros Cósmicos

Este conjunto de datos consta de 26 mediciones del parámetro de Hubble, tal como se muestra en la figura 2.5 . Las barras de error asociadas en cada punto representan la desviación estándar en nuestra fórmula para χ^2 . Por lo tanto, para la estimación de parámetros, consideramos directamente como nuestro modelo teórico a las ecuaciones de Friedmann (2.29), (2.30) y (2.34) asociadas a cada modelo.

Modelo Λ CDM

El modelo más sencillo, es decir, el que consta de menos parámetros es el modelo Λ CDM, donde utilizamos la ecuación de Friedmann (2.29) y los datos de Cronómetros Cósmicos para definir nuestra función a optimizar. Los datos teóricos los representamos por:

$$H(z)_{\text{teo}} = H_0 \sqrt{\Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0})}. \quad (6.1)$$

Por lo que, la función χ^2 queda como:

Caso 1. Parámetros del algoritmo PSO para la función $\chi_{\Lambda\text{CDM}}^2$.						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi_{\Lambda\text{CDM}}^2$	0.4	0.8	0.9	30	100	$[0, 1]$

Cuadro 6.1: Parámetros de diseño para optimizar la función $\chi_{\Lambda\text{CDM}}^2$, considerando que $H_0=67.4$.

$$\chi_{\Lambda\text{CDM}}^2 = \sum_{i=1}^{26} \frac{(H(z)_{\text{teo}} - H(z)_{\text{obs}})^2}{\sigma_i^2} \quad (6.2)$$

Dado que este modelo solo consta de dos parámetros a estimar, podemos hacer una sencilla comparación entre hallar $\Omega_{m,0}$ dejando a H_0 fijo, o hallar ambos valores variables. Por ejemplo, si fijamos H_0 de acuerdo a (2.16), el algoritmo PSO estimará un valor para $\Omega_{m,0}$. Posteriormente, si consideramos a H_0 como otro parámetro a estimar, esperaríamos que el valor de $\Omega_{m,0}$ no diverja tanto del primer valor hallado.

Caso 1. Para este primer resultado consideramos que H_0 tiene un valor de 67.4, correspondiente a un análisis reciente de la misión Planck. Como el problema de optimización solo consta de un parámetro, podemos graficar la función (6.2), con valores de 0 a 1 para el parámetro $\Omega_{m,0}$. Cabe resaltar que para la optimización **no** se definió ningún vector de valores para el parámetro de densidad, ya que el algoritmo se encarga precisamente de buscar en el espacio de parámetros el valor donde se halle el mínimo de χ^2 , sin embargo, para el gráfico es necesario hacerlo. La imagen 6.1 muestra la función a optimizar junto con los resultados obtenidos. Vemos que la gráfica es unimodal convexa, por lo que los parámetros elegidos pueden considerarse análogos a los de la función $f_1(x) = x^2$, estos parámetros se muestran en la tabla 6.1.

Como se observa en la figura 6.1, los resultados son:

- El mejor fitness, es decir, el mínimo de $\chi_{\Lambda\text{CDM}}^2$ es: 14.461.
- La mejor posición, es decir, el valor de $\Omega_{m,0}$ que minimiza la función fitness es: 0.332.
- Recordando que $\Omega_{\Lambda,0} = 1 - \Omega_{m,0}$, obtenemos que $\Omega_{\Lambda,0}=0.668$.

Este valor de $\chi_{\Lambda\text{CDM}}^2$ fue el valor más pequeño hallado en una serie de 10 ejecuciones del programa, este método implica que podemos hallar un promedio y una desviación estándar de las cantidad χ^2 y $\Omega_{m,0}$. Estos resultados se pueden observar en el apéndice D.

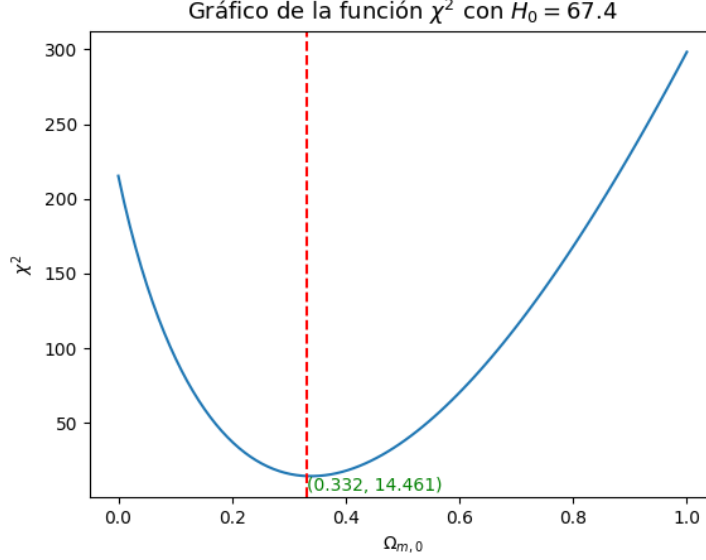


Figura 6.1: Parámetro Ω_m vs χ^2 . Modelo Λ CDM usando los datos de cronómetros cósmicos y una constante de Hubble fija, $H_0=67.4$. Las líneas punteadas en rojo señalan la posición del óptimo y sus coordenadas en color verde.

El resultado de $\Omega_{m,0}$ entra dentro de la incertidumbre de los resultados hallados en los artículos [91] y [92].

Caso 2. Para esta situación, el propósito es estimar los parámetros H_0 y $\Omega_{m,0}$, por lo que la función fitness $\chi^2_{\Lambda\text{CDM}}$ consta de dos variables, es decir que el espacio de parámetros es 2-dimensional, pero como vimos en la sección anterior, el algoritmo PSO puede hallar el óptimo para esta clase de funciones sin mayor problema.

Las consideraciones ahora son diferentes, en primer lugar la búsqueda de las partículas se hará en el rectángulo $[60, 80] \times [0, 1]$, donde el intervalo $[60, 80]$ representa un límite inferior y superior para el parámetro de Hubble. Por otro lado, hemos mencionado anteriormente que una ventaja de los algoritmos bio-inspirados, es que pueden hallar el mínimo de una función sin conocer ciertas características, como su forma o geometría. Es importante tener esto en cuenta, ya que para funciones que dependen de más de dos parámetros, sus gráficas no pueden llevarse a cabo, y la elección de los parámetros de PSO no se siguen tan fácilmente como lo fue con las funciones test. Es por ello, que en este caso no graficaremos la función fitness, en su lugar, seguiremos los pasos mostrados en el apéndice C, que de manera resumida,

Caso 2. Parámetros del algoritmo PSO para la función $\chi^2_{\Lambda\text{CDM}}$.						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi^2_{\Lambda\text{CDM}}$	0.9	0.3	1	950	600	$[60, 80] \times [0, 1]$

Cuadro 6.2: Parámetros de diseño para optimizar la función $\chi^2_{\Lambda\text{CDM}}$, considerando que H_0 no es fija.

consta de fijar primero el número de iteraciones, el número de partículas y el peso de inercia ω para asegurar una buena exploración del enjambre. Una vez hallados y fijados estos valores, se espera que con cualquier combinación de los parámetros c_1 y c_2 , los resultados óptimos de la función fitness no diverjan entre sí.

Los parámetros de diseño que dieron los resultados de $\chi^2_{\Lambda\text{CDM}}$ más óptimos se encuentran en la tabla 6.2. Esto dio como resultado los siguientes valores de los parámetros del modelo ΛCDM y de la función fitness χ^2 :

- El mejor fitness fue: $\chi^2 = 14.395$.
- Las mejores posiciones halladas fueron: $H_0 = 68.187$ y $\Omega_{m,0} = 0.318$.
- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.682.

En esta segunda situación, el valor de $\Omega_{m,0}$ tiene un valor que entra dentro de la incertidumbre del valor obtenido en [27], a diferencia del primer caso. Por otro lado, el valor de H_0 se encuentra entre los valores intermedios hallados en la literatura, tal como se mencionó en la sección 2.4, particularmente, es comparable con lo que se estimó en [29] y [30].

Finalmente, en la imagen 6.2 se muestra el ajuste del modelo ΛCDM a los datos de Cronómetros Cósmicos considerando el caso en que H_0 es fijo y variable. En ella puede observarse que el ajuste entre estos dos casos parecen ser muy similares, ya que cuando el redshift es muy pequeño las rectas se superponen, sin embargo, conforme aumenta, puede verse una pequeña diferencia entre ambos ajustes.

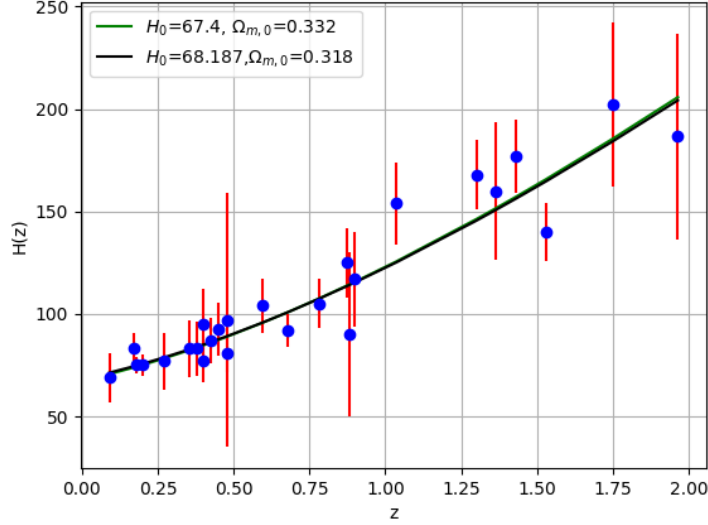


Figura 6.2: Ajuste de los datos utilizando el modelo Λ CDM, utilizando los parámetros hallados por PSO. La línea verde muestra el ajuste considerando el caso 1 y la línea negra muestra el ajuste considerando el caso 2.

Modelo ω CDM

El siguiente modelo a estimar sus parámetros es el modelo ω CDM, que consta de un parámetro extra: ω_0 . Este parámetro es tal que se ajuste a los datos observacionales. Los parámetros a estimar son H_0 , $\Omega_{m,0}$ y ω_0 . Este último parámetro se considera como un parámetro libre, por lo que, siguiendo el artículo [11], consideraremos que el valor está en un intervalo $[-2, 0]$. Es claro que su valor no puede alejarse mucho del propuesto en Λ CDM ($\omega = -1$) debido a que las observaciones aún tienen varias concordancias con el modelo anterior.

La función a optimizar ($\chi^2_{\omega\text{CDM}}$) es totalmente análoga a la mostrada en la ecuación (6.2). La única diferencia es que $H(z)_{\text{teo}}$ está expresada, a partir de la ecuación (2.30), como:

$$H(z)_{\text{teo}} = H_0 \sqrt{\Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0})(1+z)^{3(1+\omega_0)}}. \quad (6.3)$$

Los límites del resto de parámetros H_0 y $\Omega_{m,0}$, permanecerán igual que en el caso anterior, por lo que la optimización se hará en el rectángulo 3-dimensional $[60, 80] \times [0, 1] \times [-2, 0]$. Para este modelo, uno ya no puede

Parámetros del algoritmo PSO para la función $\chi_{\omega\text{CDM}}^2$.						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi_{\omega\text{CDM}}^2$	0.6	0.9	1	1200	1000	$[60, 80] \times [0, 1] \times [-2, 0]$

Cuadro 6.3: Parámetros de diseño seleccionados para optimizar la función $\chi_{\Lambda\text{CDM}}^2$.

ayudarse de la gráfica de la función a optimizar para la elección de los parámetros de PSO. Sin embargo, como la dimensión ha aumentado, es recomendable aumentar el número de partículas para una mejor exploración. La elección de los coeficientes de aceleración, así como del número de partículas e iteraciones se hizo de manera análoga al caso 2 del modelo ΛCDM , siguiendo el apéndice C.

Los parámetros de diseño seleccionados para este modelo se muestran en la tabla 6.3. Se puede observar que el número de partículas e iteraciones tuvieron que aumentar, mientras que el peso de inercia se mantuvo igual. La razón por la que el parámetro ω no cambió, es que si lo consideráramos menor a 1, debíamos aumentar en casi el doble el número de iteraciones (2000), y definir más partículas (1500) que las mostradas en la tabla 6.3. De no hacer esto, los valores de $\chi_{\omega\text{CDM}}^2$ difieren mucho del óptimo, ya que los resultados (para distintas ejecuciones del programa) rondaban entre 14 y 19. Sin embargo, elegir un mayor número de iteraciones implica considerar un mayor tiempo de convergencia. Por otro lado, si elegíamos $\omega > 1$, a costa de disminuir el número de iteraciones o partículas, obteníamos valores de la función fitness que rondaban entre 14 y 16, lo cual es una mejora al caso anterior, pero sigue siendo mayor a los resultados obtenidos con nuestros parámetros de diseño, esto puede verse en el apéndice D.

Los parámetros estimados del modelo ωCDM y de la función fitness son los siguientes:

- El mejor fitness fue: $\chi^2 = 14.226$.
- Las mejores posiciones fueron: $H_0 = 70.895$, $\Omega_{m,0} = 0.326$ y $\omega_0 = -1.285$.
- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.674.

El valor estimado de H_0 se encuentra dentro de la incertidumbre de [31], mientras que el valor del parámetro de densidad de materia $\Omega_{m,0}$ tiene un valor mayor que en la situación pasada, sin embargo entra dentro de la incertidumbre de lo reportado en [93] y [92]. Por otro lado, el parámetro libre ω_0 tiene un valor menor a -1 . Existen restricciones acerca del valor ω_0 , en

Parámetros del algoritmo PSO para la función χ_{CPL}^2 .

Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
χ_{CPL}^2	0.8	0.6	1	1370	1100	$[60, 80] \times [0, 1] \times [-2, 0] \times [-2, 2]$

Cuadro 6.4: Parámetros de diseño seleccionados para optimizar la función χ_{CPL}^2 .

particular que ω_0 debe tomar un valor entre -1 y $-1/3$ para considerar que el universo se expande aceleradamente en la actualidad. Sin embargo hay modelos que permiten un valor de $\omega_0 < -1$, tal como los modelos de energía oscura fantasma [6].

Modelo CPL

El modelo CPL consta de estimar 4 parámetros: $H_0, \Omega_{m,0}, \omega_0$ y ω_a , donde los últimos 2 parámetros se consideran parámetros libres, ω_0 lo volvemos a considerar en un intervalo $[-2, 0]$ y ω_a en el intervalo $[-2, 2]$ siguiendo nuevamente el artículo [11]. En este caso, de acuerdo a la ecuación (6.2), el modelo teórico está descrito por:

$$H(z)_{\text{teo}} = H_0 \sqrt{\Omega_{m,0}(1+z)^3 + (1 - \Omega_{m,0})(1+z)^{3(1+\omega_0+\omega_a)} \cdot \exp\left(\frac{-3\omega_a z}{1+z}\right)}. \quad (6.4)$$

La búsqueda del enjambre debe establecerse en el rectángulo 4-dimensional $[60, 80] \times [0, 1] \times [-2, 0] \times [-2, 2]$, al aumentar en una dimensión el espacio de parámetros, esperamos que al igual que en el caso anterior, el número de iteraciones y/o partículas también aumente. Durante la elección de los parámetros PSO, notamos que al incrementar en 100 el número de partículas y 170 el de iteraciones respecto a la tabla 6.3, los valores óptimos de la función fitness (χ_{CPL}^2) no divergían en gran cantidad al implementar valores aleatorios de c_1 y c_2 . Con este ajuste, los valores óptimos rondaban entre 14 y 15, esto concuerda con lo esperado para este modelo. Finalmente, de acuerdo al criterio de selección de parámetros obtuvimos el siguiente resultados (ver tabla 6.4):

A continuación mostramos los parámetros estimados:

- El mejor fitness fue: $\chi^2 = 14.153$.
- Las mejores posiciones fueron: $H_0 = 71.787$, $\Omega_{m,0} = 0.300$, $\omega_0 = -1.391$ y $\omega_a = 0.923$.

- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.700.

Podemos notar que con este modelo obtuvimos el valor más alto de H_0 , sin embargo, sigue siendo un valor intermedio entre los resultados de [31] y [28]. Por otro lado, $\Omega_{m,0}$ tiene un valor intermedio respecto a los resultados de los modelos anteriores y se corresponde más con lo obtenido en [30].

De lo mencionado en la sección 2.5.3, podríamos esperar que el valor de ω_a sea lo suficientemente pequeño para que a día de hoy $\omega \approx \omega_0$. Sin embargo, en nuestros resultados obtuvimos un valor cercano a uno, esto difiere de la predicción teórica y de lo obtenido en algunos artículos, como en [11] y [94]. Sin embargo, en estos trabajos la estimación de parámetros del modelo CPL considera datos conjuntos y no un conjunto de datos individual como en este caso. Veremos más adelante los valores estimados de ω_0 y ω_a considerando supernovas y datos CC+SN Ia para un análisis más detallado.

Podemos verificar visualmente de la gráfica 6.3, que con los parámetros estimados para cada uno de los diferentes modelos se obtiene un ajuste adecuado a los datos de cronómetros cósmicos.

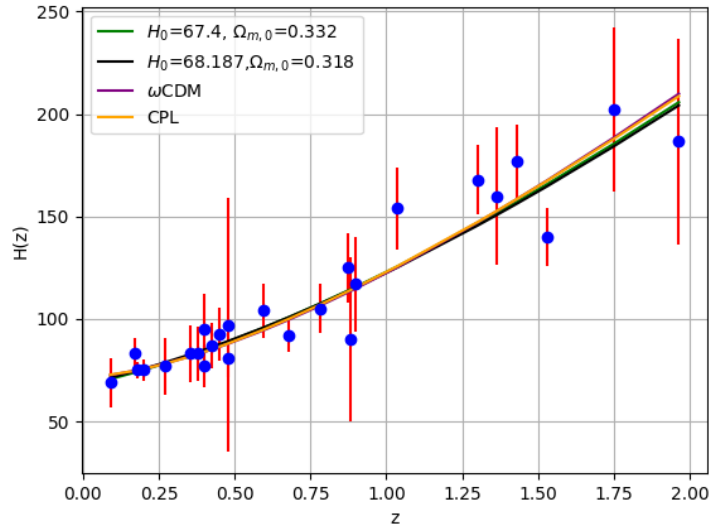


Figura 6.3: Ajuste a los datos de Cronómetros Cósmico utilizando el caso 1 y 2 del modelo Λ CDM, y los modelos ω CDM y CPL.

De la gráfica anterior podemos ver que, aunque los valores estimados de ω_0 y ω_a del modelo CPL no son los previstos teóricamente, éste se ajusta de

manera muy similar a otros modelos, como el Λ CDM, cuyos valores de los parámetros concuerdan con en el de otros trabajos citados anteriormente.

6.0.2. Datos de Supernovas

Los datos de supernovas nos permiten hacer una estimación de parámetros cosmológicos por medio de la distancia de luminosidad D_L (ecuación 2.51), esto nos permite definir el módulo de distancia teórico y compararlo por medio del test χ^2 con el módulo de distancia observado y recopilado.

El ajuste a los modelos cosmológicos con los datos de supernovas será por medio de minimizar la siguiente función:

$$\chi^2 = (\hat{\mu} - \mu)^T \mathbf{C}^{-1} (\hat{\mu} - \mu), \quad (6.5)$$

donde $\hat{\mu}$ representa el módulo de distancia observado, que en este caso es un vector de 31 entradas, μ el módulo de distancia teórico y $\mathbf{C}^{-1}$ la matriz inversa de covarianza, esta expresión es análoga a la ecuación (2.43).

La elección de parámetros PSO para cada modelo se hizo como en el caso de Cronómetros Cósmicos, siguiendo los pasos mostrados en C. Y los límites de cada uno de los parámetros a estimar los mantendremos en los mismos intervalos que en los resultados pasados.

Modelo Λ CDM

De la ecuación de distancia luminosa (2.51), sustituimos en la función $H(z')$ a la ecuación (6.1), por lo que para este y el resto de modelos se considera a la constante de Hubble H_0 como otro parámetro a estimar tal como se hizo en el análisis anterior, la diferencia en esta situación es que no le asignaremos un valor a H_0 .

En la tabla 6.5 se muestran los parámetros de diseño seleccionados para este modelo. En ella se puede notar que consideramos valores más bajos para el peso de inercia, el número de partículas y el número de iteraciones que en el caso de los Cronómetros Cósmicos. La razón de ello fue que el valor óptimo se hallaba rápidamente, es decir, si considerábamos un mayor número de iteraciones, proporcional al escogido para los datos anteriores, obteníamos resultados del fitness muy similares pero a lo largo de un mayor tiempo de ejecución, esto lo vuelve innecesariamente costoso en términos computacionales. Podemos mostrar esto en la figura 6.4, donde se observa que después de 10 iteraciones, el algoritmo encuentra el mínimo de la función fitness. Respecto al peso de inercia, notamos que el valor mostrado en la tabla

Parámetros de diseño para optimizar la función $\chi^2_{\Lambda\text{CDM}}$.						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi^2_{\Lambda\text{CDM}}$	0.5	0.6	0.7	20	80	$[60, 80] \times [0, 1]$

Cuadro 6.5: Parámetros de diseño para optimizar la función $\chi^2_{\Lambda\text{CDM}}$, considerando los datos de supernovas.

era suficiente para evitar que las partículas se estancaran en regiones alejadas del óptimo.

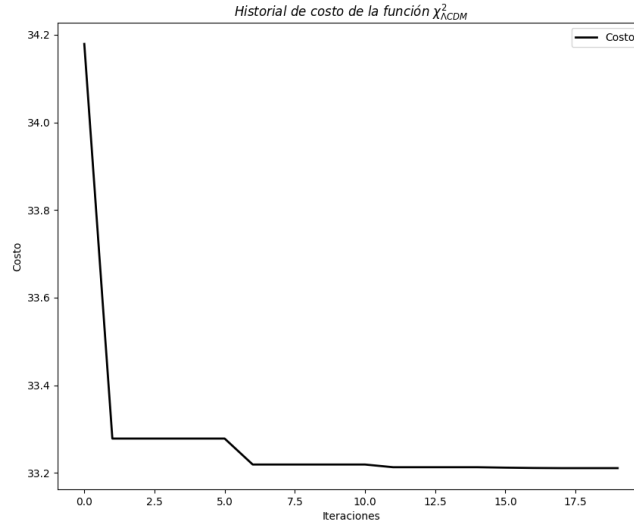


Figura 6.4: Historial de costo para el modelo ΛCDM con datos de supernovas.

Procediendo con los pasos para la estimación de parámetros, obtuvimos los siguientes resultados:

- El mejor fitness fue $\chi^2=33.210$.
- Las mejores posiciones fueron: $H_0=70.008$ y $\Omega_{m,0}=0.297$.
- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.703.

Con este conjunto de datos, vemos que el valor de la función fitness χ^2 es más grande que en el caso previo, lo cuál es de esperarse ya que ambos

Parámetros de diseño para optimizar la función $\chi_{\omega\text{CDM}}^2$.						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi_{\omega\text{CDM}}^2$	0.4	0.6	0.7	40	80	$[60, 0, -2] \times [80, 1, 0]$

Cuadro 6.6: Parámetros de diseño para optimizar la función $\chi_{\omega\text{CDM}}^2$, considerando los datos de supernovas.

conjuntos son de observaciones distintas y el número de datos asociados a cada conjunto es diferente. Por otro lado, respecto a los valores de la constante de Hubble y el parámetro de densidad de materia, vemos que los resultados no difieren en gran medida con los hallados anteriormente, siendo estos resultados muy similares a los que se obtuvieron para el modelo ωCDM , por lo que de igual manera, H_0 entra dentro de la incertidumbre de [31] y $\Omega_{m,0}$ concuerda con lo estimado en [30].

Modelo ωCDM

En esta ocasión consideramos que la función $H(z')$ de la distancia luminosa está dada por la ecuación (6.3). Para este modelo, los parámetros se muestran en la tabla 6.6.

Los parámetros PSO en general no difieren al caso del modelo ΛCDM , a excepción de un cambio en el coeficiente de aceleración c_1 y en el número de iteraciones, el cuál decidimos aumentarlo debido a que la dimensión de la función también aumenta. Sin embargo, en general con cualquier combinación aleatoria entre 0 y 1 de estos coeficientes obtuvimos buenos valores óptimos.

Los resultados de la estimación se muestran a continuación:

- El mejor fitness fue $\chi^2=32.469$.
- Las mejores posiciones fueron: $H_0=69.664$, $\Omega_{m,0}=0.177$ y $\omega_0=-0.763$.
- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.823.

En esta ocasión el resultado mínimo de la función fitness es más bajo que en el caso anterior. Podemos notar de la tabla D.5 que algunos resultados son muy similares a tercer orden, es decir, la tercera cifra decimal. Sin embargo, para obtener el valor más pequeño de estos resultados, junto a sus respectivas posiciones en el espacio de búsqueda, fue necesario utilizar la función $\min()$ de Python que retorna el valor más pequeño de una lista de números.

Respecto a los parámetros cosmológicos de este modelo notamos otra cosa, H_0 tiene un valor intermedio entre 68 y 70, cuyos resultados se obtuvieron anteriormente y entra dentro de la incertidumbre de [30] y [31]. Por otro lado, el valor de ω_0 está ligeramente más próximo a -1 que en el conjunto de datos previo. Una diferencia importante es que al ser próximo a -1 pero menor a este, el término exponencial del modelo es positivo a diferencia del caso con datos CC: $1 + \omega_0 \approx 0.23$. Podemos considerar que esto es una mejora en ω_0 , ya que el término $(1 + z)$ no crecerá tanto conforme z aumente, y por lo tanto, la diferencia entre Λ CDM y ω CDM no será tan significativa.

Finalmente, el parámetro $\Omega_{m,0}$ tiene el valor más bajo obtenido hasta el momento. Pocos trabajos consideran una estimación del parámetro de densidad de materia tan pequeños. En el artículo [95] se estima un valor cercano al 20 %. Con este valor se consideraría que el parámetro de densidad asociado con la energía oscura es mayor, con más de 80 % de prevalencia según nuestros criterios. En la tabla D.5 vemos que los valores para este parámetro permanecen similares, el mayor considera una prevalencia del 18 %. Una manera de comprobar si con estos resultados el ajuste a los datos es bueno, es por medio de una gráfica, la cual mostraremos más adelante con todos los modelos incluidos.

Modelo CPL

Finalmente, para el modelo CPL representado en la ecuación (6.4), los parámetros escogidos se muestran en la tabla 6.7. Siguiendo el *criterio para la elección de parámetros*, decidimos mantener el mismo número de partículas, aumentando en 5 el número de iteraciones en comparación al caso anterior, sin embargo, elegimos un mayor peso de inercia. Con estas especificaciones vimos que para cifras aleatorias de los coeficientes de aceleración, los óptimos se mantenían homogéneos. La razón por la que decidimos no aumentar el número de partículas y en aumentar poco el número de iteraciones, se debe a que para este modelo y con estos datos, el tiempo de búsqueda de las partículas es mayor. Con estas cantidades logramos un buen balance entre tiempo de ejecución y una buena convergencia del enjambre hacia el óptimo.

Podemos comparar el historial de costo de Λ CDM (figura 6.4) con el de este modelo (figura 6.5), donde se aprecia que a lo largo de las iteraciones, las partículas permanecen temporalmente en (lo que podrían ser) óptimos locales hasta converger al global. Cabe mencionar que este historial de costo no corresponde a los resultados que reportamos más adelante, pero sí corresponde a otro valor obtenido considerando los parámetros de la tabla 6.7, es decir al de un valor perteneciente a la vecindad del valor reportado.

Parámetros de diseño para optimizar la función χ^2_{CPL} .						
Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
χ^2_{CPL}	0.8	0.9	0.9	45	80	$[60, 80] \times [0, 1] \times [-2, 0] \times [-2, 2]$

Cuadro 6.7: Parámetros de diseño para optimizar la función χ^2_{CPL} , considerando los datos de supernovas.

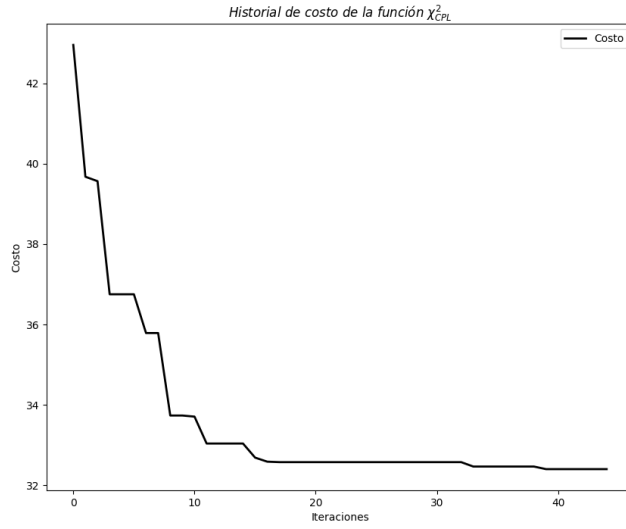


Figura 6.5: Historial de costo para el modelo CPL con datos de supernovas.

Los valores obtenidos de los parámetros del modelo son:

- El mejor fitness fue $\chi^2=32.410$.
- Las mejores posiciones fueron: $H_0=69.481$, $\Omega_{m,0}=0.031$, $\omega_0=-0.653$ y $\omega_a=0.391$.
- Considerando el valor de $\Omega_{m,0}$, obtenemos que el valor de $\Omega_{\Lambda,0}$ es: 0.969.

Nuevamente, para este modelo el valor mínimo de la función fitness es más pequeño que el obtenido para ΛCDM y ligeramente menor que en el modelo anterior. Esto nos llevó a suponer que si aumentamos el número de iteraciones y/o partículas a este modelo, podríamos encontrar valores

más bajos que el reportado, sin embargo, esto no fue así, ya que el óptimo permaneció constante.

Por otro lado, el valor del parámetro de densidad de materia tiene un valor aún más pequeño que en el caso anterior, el valor encontrado correspondería a un 3% del contenido energético del universo, por lo que habría un mayor predominio casi total de la energía oscura. Este valor tan pequeño no ha sido reportado en algún otro trabajo, nuevamente en el artículo [95] se concluye que el universo podría tener una baja densidad, estimando que $\Omega_{m,0} \approx 0.2$, sin embargo, los valores más aceptados se consideran próximos a 30%. Por otro lado, vemos que se obtiene un buen valor de H_0 , aproximado al reportado en [31]. Finalmente, se obtuvo un valor más pequeño de la cantidad ω_a respecto al caso de los datos CC y entra dentro de la incertidumbre del valor hallado en [4].

La tabla D.6 muestra los 10 resultados obtenidos de la optimización, vemos que los valores de χ^2 son próximos entre sí, pero que para cada uno de ellos se obtienen valores muy distintos de $\Omega_{m,0}$. En cambio, todos los estimados de H_0 son buenos valores que rondan entre 69 y 70.

Como consideramos que los valores de los parámetros del modelo CPL mencionados anteriormente no son convincentes, haremos una comparación del ajuste a los datos de supernovas considerando 2 valores distintos de χ^2 , y por lo tanto, otras cantidades de los parámetros asociados al fitness. Es decir, graficaremos el ajuste del modelo para el valor $\chi_{\text{CPL}_1}^2$ reportado anteriormente y para otro valor $\chi_{\text{CPL}_2}^2=32.641$ mostrado en dicha tabla D.6. Tomaremos en cuenta el otro valor fitness, ya que el vector de parámetros correspondiente tiene mejores valores que los reportados. Para esta cantidad, el parámetro $\Omega_{m,0}$ aún es pequeño respecto a lo que encontramos en los datos de Cronómetros Cósmicos, sin embargo, entra en la incertidumbre de lo reportado en [96] y [95], mientras que ω_a tiene un valor más cercano a 0 que lo reportado anteriormente, también ω_0 es más próximo a -1, tal como se espera en la actualidad según el modelo CPL y la concordancia observacional que hay con Λ CDM. La gráfica del ajuste del modelo CPL para estas dos cantidades con sus respectivos parámetros hallados se muestra en la figura 6.6.

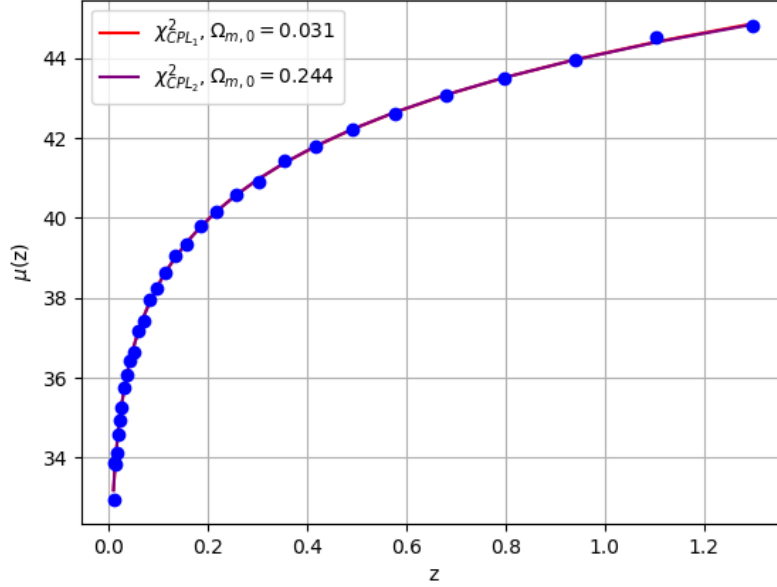


Figura 6.6: Ajuste del modelo CPL a los datos de Supernovas considerando diferentes valores de los parámetros cosmológicos asociados a $\chi^2_{\text{CPL}_1}=32.410$ y $\chi^2_{\text{CPL}_2}=32.641$

Vemos que el ajuste para ambas funciones fitness es prácticamente idéntico, es decir, que con la combinación de los parámetros hallados en ambos casos no hay una diferencia notable en el ajuste de los datos. Sin embargo, para los fines de este trabajo, consideraremos la función fitness $\chi^2_{\text{CPL}_1}$ para la selección de parámetros, ya que de todos los resultados obtenidos, ese es el más pequeño. La comparación anterior solo tenía el propósito de mostrar que también hay una buena aproximación considerando un parámetro de densidad de materia mayor.

La tabla D.6 visibiliza buenas cantidades de H_0 , por lo que podríamos asegurar que con este conjunto de datos podemos hacer buenas estimaciones de la constante de Hubble.

Una vez estimados los parámetros de cada modelo, mostraremos en la imagen 6.7 el ajuste de cada uno de ellos a los datos de supernovas. Se observa que a redshifts pequeños, los 3 modelos parecen superpuestos y conforme el redshift aumenta y es mayor a 1, hay una ligera diferencia del ajuste correspondiente a CPL con los modelos ΛCDM y ωCDM .

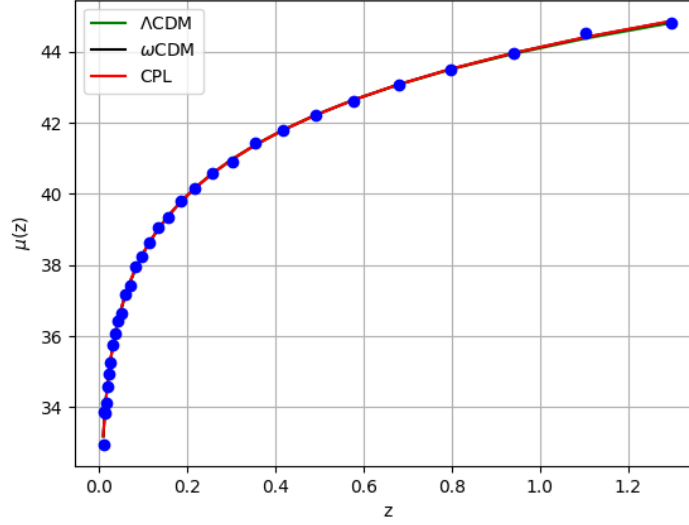


Figura 6.7: Ajuste de los modelos Λ CDM (verde), ω CDM (negro) y CPL (rojo) a los datos de supernovas.

6.0.3. Datos conjuntos CC+SN Ia

Una práctica común en la estimación de parámetros cosmológicos, es hacer un análisis conjunto de los datos. Si asumimos que los datasets son estadísticamente independientes, podemos obtener una función de verosimilitud conjunta. En la práctica, esto se traduce en tomar el producto de los likelihoods obtenidos para cada dataset.

De acuerdo a la relación que dimos entre la función de verosimilitud y la función χ^2 en la sección 2.6, y por lo mostrado en el artículo [97], se tiene que la estimación se hace por medio del cálculo de la siguiente función:

$$\chi_{\text{total}}^2 = \sum_{r=1}^D \chi_r^2, \quad (6.6)$$

donde D es el número de datasets independientes. Por lo que la función fitness a optimizar para cada modelo:

$$\chi_{\text{tot},i}^2 = \chi_{\text{CC},i}^2 + \chi_{\text{SN Ia},i}^2,$$

Parámetros PSO para optimizar la función χ^2 de cada modelo cosmológico.

Función	c_1	c_2	ω	Iteraciones	Partículas	Límites
$\chi_{\text{tot},\Lambda\text{CDM}}^2$	0.6	0.6	0.7	60	25	$[60, 80] \times [0, 1]$
$\chi_{\text{tot},\omega\text{CDM}}^2$	0.9	0.6	0.8	70	45	$[60, 80] \times [0, 1] \times [-2, 0]$
$\chi_{\text{tot},\text{CPL}}^2$	0.5	0.9	0.9	100	60	$[60, 80] \times [0, 1] \times [-2, 0] \times [-2, 2]$

Cuadro 6.8: Parámetros de diseño para optimizar la función fitness de cada modelo cosmológico usando datos conjuntos.

donde el índice i hace referencia al modelo cosmológico. En este análisis no segmentaremos por conjuntos de datos, únicamente por modelos, y juntaremos nuestros resultados en 2 tablas. La primera corresponde a los parámetros del algoritmo PSO y la segunda corresponde a los resultados de la optimización.

Los parámetros del algoritmo PSO se seleccionaron siguiendo los mismos criterios que en los casos pasados, el único parámetro que se mantendrá igual para cada modelo corresponde al de los límites. Los resultados de esta elección se muestran en la tabla 6.8.

Notamos que la elección de iteraciones y partículas es mucho menor que en el caso de Cronómetros Cósmicos, y de hecho estos valores se asemejan más a los reportados para los datos de Supernovas. También podemos observar que le dimos más peso al número de iteraciones que al número de partículas para asegurar una exploración más diversa, esto junto con el incremento del peso de inercia ω , el cual aumentamos conforme el número de parámetros cosmológicos aumentaba.

Finalmente, con estos parámetros obtuvimos los siguientes resultados de la optimización, que se muestran en la tabla 6.9. Marcamos con guión los parámetros que se mantendrán fijos según el modelo, es decir: $\omega = \omega_0 = -1$ y $\omega_a = 0$.

Todos los valores obtenidos de H_0 corresponden a un valor muy próximo a 70, por lo que el resultado se asemeja a lo reportado en [31]. Por otro lado, el parámetro $\Omega_{m,0}$ toma en general, mejores valores que los obtenidos por los datos de Supernovas. El valor más alto de este parámetro, correspondiente al modelo CPL entra en la incertidumbre de lo obtenido en [30], mientras que para el modelo ΛCDM , este valor se aproxima a lo obtenido por [11]. El valor más bajo de este parámetro corresponde al modelo ωCDM , sin embargo, es cercano a una cantidad del 29 %, lo cual sigue siendo aceptado en algunos trabajos, como en [98]. Como consecuencia, al parámetro Ω_Λ le corresponderían valores entre 69.6 % y 71.3 %.

Valores de los parámetros cosmológicos.			
Parámetro	Λ CDM	ω CDM	CPL
H_0	69.946	69.861	69.771
$\Omega_{m,0}$	0.295	0.287	0.304
ω_0	...	-0.969	-0.956
ω_a	...	...	-0.377
χ^2	47.960	47.918	47.900

Cuadro 6.9: Parámetros cosmológicos estimados para diferentes modelos, usando datos conjuntos CC+SN Ia.

Respecto al parámetro libre ω_0 , se hallaron los valores más cercanos a -1 que en el resto de los resultados anteriores con datos individuales. Esto concuerda con lo obtenido en otros trabajos de estimación de parámetros, como en [4], [11] y [13]. Esto indica que el modelo ω CDM puede aproximarse a Λ CDM, ya que el termino $(1+z)^{3(1+\omega_0)}$ es similar a 1 al hacer la sustitución del resultado obtenido.

Finalmente, el modelo CPL es el único que considera a ω_a como un parámetro no fijo, y por la predicción teórica, este valor debería ser cercano a 0, sin embargo vemos que esto no es así, aunque sí es el valor que más se aproxima en comparación de los casos anteriores. Aunque en la tabla D.9 se puede observar que en una vecindad cercana al óptimo reportado, se encuentran valores de ω_a más próximos a cero, manteniendo a su vez, buenos resultados del resto de sus parámetros.

6.0.4. Selección de modelos

Ya que hemos estimado los parámetros de los 3 modelos cosmológicos, estamos interesados en determinar el mejor modelo siguiendo los criterios de información AIC y BIC. Primero lo haremos para cada uno de los datos CC y SN Ia individualmente y luego, para los datos combinados CC+SN Ia.

Datos individuales

Empezaremos por el criterio de información Akaike siguiendo la ecuación (2.46), donde el valor χ^2 expresado en dicha fórmula corresponde a lo que hemos reportado para los diferentes modelos, y está asociado con la función de máxima verosimilitud $\mathcal{L}_{\max}$. Primero utilizaremos los datos de Cronómetros Cósmicos y Supernovas de forma individual, el modelo que tenga el valor AIC más bajo será considerado como el mejor modelo o el menos

Valores óptimos de χ^2 usando datos individuales.

Datos	$\chi^2_{\Lambda\text{CDM}}$	$\chi^2_{\omega\text{CDM}}$	χ^2_{CPL}
CC	14.395	14.226	14.153
SNIa	33.210	32.469	32.140

Cuadro 6.10: Valores obtenidos del test χ^2 para los 3 modelos cosmológicos.

Valores de AIC usando datos individuales.

Datos	ΛCDM	ωCDM	CPL
CC	18.395	20.226	22.153
SNIa	37.210	38.469	40.140

Cuadro 6.11: Valores obtenidos del criterio de información Akaike para los 3 modelos cosmológicos.

penalizado según este criterio. Mostraremos en la tabla 6.10 los resultados óptimos, anteriormente mencionados de la función fitness. Posteriormente, para el criterio Akaike, los resultados se muestran en la tabla 6.11, donde puede verse que el modelo ΛCDM tiene los valores más pequeños.

El otro criterio de información, el Bayesiano tiene la función de penalizar aún más a los modelos con mayor número de parámetros aún considerando que se ajusten bien a los datos. De igual manera, consideramos datos CC y SNIa individualmente. La tabla 6.12 muestra los resultados de este criterio, también puede notarse que el modelo con el valor BIC más bajo es ΛCDM . La diferencia de valores entre los modelos ΛCDM y CPL, para ambos datasets es mayor que la diferencia que hay usando el criterio Akaike, esto confirma la rigurosidad de este criterio con modelos que constan de varios parámetros

Valores de BIC usando datos individuales.

Datos	ΛCDM	ωCDM	CPL
CC	20.911	23.999	27.186
SNIa	40.078	42.772	46.146

Cuadro 6.12: Valores obtenidos del criterio de información Bayesiano para los 3 modelos cosmológicos.

Valores óptimos de χ^2 usando datos conjuntos.

Datos	$\chi^2_{\Lambda\text{CDM}}$	$\chi^2_{\omega\text{CDM}}$	χ^2_{CPL}
CC+SNIa	47.960	47.918	47.900

Cuadro 6.13: Valores obtenidos del test χ^2 para cada modelo cosmológico con datos combinados

Valores AIC y BIC usando datos combinados.

Criterio	ΛCDM	ωCDM	CPL
AIC	51.960	53.918	55.899
BIC	56.046	60.047	64.071

Cuadro 6.14: Valores obtenidos del criterio de información Akaike y Bayesiano para los 3 modelos cosmológicos.

Datos combinados

Utilizamos nuevamente las ecuaciones (2.46) y (2.47) para seleccionar el modelo que más se ajusta a los datos con el menor número de parámetros. Primero recopilamos en la tabla 6.13 los resultados óptimos de la función fitness. Posteriormente, en la tabla 6.14 ponemos el valor del criterio AIC y BIC.

Los valores AIC y BIC son mayores en esta situación, esto se debe a que la función χ^2 también es mayor que en los casos individuales. Sin embargo, también notamos que el modelo con los valores más pequeños de estos criterios es el modelo ΛCDM . Mientras que el más penalizado por ambos criterios es el modelo CPL.

7 Conclusiones

En nuestra tarea de estimar parámetros cosmológicos, propusimos al algoritmo de optimización por enjambre de partículas como un método eficaz para encontrar óptimos globales de funciones que constan de varios óptimos o de alta dimensionalidad, ya que los algoritmos clásicos o deterministas no son eficaces para optimizar esta clase de funciones. Esta característica la comprobamos por medio de las funciones test, tanto de una dimensión o más, y observamos que el algoritmo no tiene problema en hallar los óptimos globales a un bajo costo computacional, tal como se observa en los historiales de costo de la sección 5.2.1.

Considerando que el algoritmo PSO no tuvo problemas con las funciones test, decidimos aplicarlo a los datos cosmológicos utilizando 26 puntos de los datos de Cronómetros Cósmicos [43] y los datos de supernovas de la compilación de 31 puntos JLA. Estimamos los parámetros de cada uno de los modelos cosmológicos, minimizando la función fitness χ^2 con el algoritmo PSO. Comprobamos que para cada modelo, el algoritmo encuentra los parámetros H_0 y $\Omega_{m,0}$ que mejor se ajustan a los datos, comparando nuestros resultados con valores reportados en la literatura. La única discrepancia se verifica con el modelo CPL y los datos de supernovas, ya que en ese caso el valor del parámetro de densidad de materia es muy pequeño y no es comparable con algún valor similar reportado por otros trabajos. Respecto a los parámetros ω_0 y ω_a correspondientes al modelo ω CDM y CPL, no obtuvimos valores comparables con los de la literatura, particularmente ω_a no se aproxima a cero con los datos de supernovas, pero fue un mejor resultado que el obtenido con los datos de Cronómetros Cósmicos. Esto nos lleva a concluir que los datos SNIa sirven para estimar principalmente el parámetro H_0 y ω_a en el caso del modelo CPL. Aún así, comprobamos de las gráficas 6.3 y 6.7, que cada modelo se ajusta bien, mostrando una nula diferencia entre ellos.

Respecto a los datos combinados, los valores obtenidos de los parámetros fueron mejores que en el caso de los datos individuales. Si bien con los Cronómetros Cósmicos se obtuvieron buenas estimaciones de H_0 y $\Omega_{m,0}$, no

se obtuvo una estimación esperada de ω_0 comparable con la literatura. Pero esto sí ocurre con los datos combinados, obteniendo $\omega_0 \approx -1$. Una situación similar ocurre cuando comparamos los datos combinados con los datos de supernovas, ya que además de estimar el parámetro ω_0 cercano a menos -1, también encuentra mejores valores de $\Omega_{m,0}$.

Por medio de los criterios de información Akaike y Bayesiano pudimos comprobar que el modelo más provechoso de los datos fue el modelo Λ CDM, ya que con un menor número de parámetros logra un ajuste igual de bueno que los otros dos modelos, esto aplicó para los datos individuales y combinados. El modelo más penalizado según estos criterios es el modelo CPL, aún cuando con este modelo se obtuvieron los valores de χ^2 más bajos.

Nuestros resultados concuerdan con los obtenidos en trabajos previos, como en [4], donde se hace una comparación de estos mismos modelos y otros más, y se concluye que el modelo más favorecido es el modelo cosmológico estándar. El trabajo [99] también hace una estimación de parámetros y comparación de modelos cosmológicos (Λ CDM, PolyCDM y CPL) utilizando Algoritmos Genéticos, que es otro representante de los algoritmos bio-inspirados, concluyendo también que el modelo que mejor describe a los datos es Λ CDM.

A Ecuaciones de campo de Albert Einstein

En la gravitación clásica de Newton, el potencial gravitatorio Φ cumple la ecuación de Poisson:

$$\nabla^2\Phi = 4\pi G\rho, \quad (\text{A.1})$$

donde ρ es la densidad de materia y G es la constante gravitacional. Sin embargo, esta ecuación de campo no es compatible con la relatividad especial, debido a que no depende del tiempo, por lo que se violaría el principio de que nada puede propagarse más rápido que la velocidad de la luz. Además, dentro de este régimen podemos escribir la ecuación de movimiento de una partícula cuya masa inercial es m_I que está dentro de un campo gravitacional:

$$\frac{d^2x}{dt^2} = -\frac{m_I}{m_G}\nabla\Phi, \quad (\text{A.2})$$

donde m_G es la masa gravitacional. Sin embargo, se ha comprobado experimentalmente que la masa gravitacional y la masa inercial, aunque conceptualmente diferentes, son iguales. Este hecho sirvió como motivación para que Albert Einstein estableciera el principio de equivalencia, el cual asegura que *en un marco de referencia en caída libre que ocupa una pequeña región del espacio-tiempo, las leyes de la física se reducen a las leyes sin gravedad, es decir, obedecen las leyes de la teoría de la relatividad especial* [100]. Esto condujo a que Einstein planteara que la gravedad se considerase como una manifestación geométrica del espacio tiempo, en lugar de una fuerza en el sentido de la gravitación clásica de Newton.

Para formalizar esta idea, partimos del caso en el que un campo gravitacional sea débil. En esta consideración, según la idea de Einstein de asociar gravitación con curvatura, un campo gravitacional débil implica poca curvatura. Estamos en condiciones de hacer algunas suposiciones:

- Las partículas en esta región del espacio-tiempo se mueven lentamente, es decir, sus velocidades son muy pequeñas comparadas con la velocidad de la luz.
- El espacio-tiempo es casi plano, por lo que la métrica puede aproximarse como: $g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$. Con $|h_{\mu\nu}| \ll 1$.
- Las componentes de la métrica no dependen del tiempo, es decir, $g_{\mu\nu,0} = 0$

Para estudiar el movimiento de las partículas (libres) en esta región del espacio-tiempo, debemos considerar sus desviaciones debido a la perturbación $h_{\mu\nu}$. Por lo tanto, tomamos en cuenta la ecuación de la geodésica:

$$\frac{d^2 x^\mu}{d\tau^2} + \Gamma_{\alpha\beta}^\mu \frac{dx^\alpha}{d\tau} \frac{dx^\beta}{d\tau} = 0, \quad (\text{A.3})$$

donde $\Gamma_{\alpha\beta}^\mu$ son los símbolos de Christoffel. Por lo tanto, si desarrollamos el segundo término abriendo las sumas sobre los índices mudos α y β :

$$\begin{aligned} \Gamma_{\alpha\beta}^\mu \frac{dx^\alpha}{d\tau} \frac{dx^\beta}{d\tau} &= \Gamma_{00}^\mu \left(\frac{dx^0}{d\tau} \right)^2 + 2\Gamma_{0i}^\mu \frac{dx^0}{d\tau} \frac{dx^i}{d\tau} + \Gamma_{ij}^\mu \frac{dx^i}{d\tau} \frac{dx^j}{d\tau} \\ &= \left(\Gamma_{00}^\mu + 2\Gamma_{0i}^\mu \frac{\partial x^i}{\partial x^0} + \Gamma_{ij}^\mu \frac{\partial x^i}{\partial x^0} \frac{\partial x^j}{\partial x^0} \right) \left(\frac{dx^0}{d\tau} \right)^2 \\ &= \left(\Gamma_{00}^\mu + 2\Gamma_{0i}^\mu \frac{u^i}{c} + \Gamma_{ij}^\mu \frac{u^i u^j}{c^2} \right) \left(\frac{dx^0}{d\tau} \right)^2, \end{aligned} \quad (\text{A.4})$$

donde hemos usado la regla de la cadena para factorizar las derivadas, y escribimos las derivadas como velocidades u^i de las partículas. Si aplicamos la condición de que las partículas tienen una velocidad lenta comparada con c , obtendremos la siguiente aproximación:

$$\Gamma_{\alpha\beta}^\mu \frac{dx^\alpha}{d\tau} \frac{dx^\beta}{d\tau} \approx \Gamma_{00}^\mu \left(\frac{dx^0}{d\tau} \right)^2. \quad (\text{A.5})$$

Al desarrollar el símbolo de Christoffel en términos de la métrica veremos que:

$$\Gamma_{00}^\mu = -\frac{1}{2} g^{\mu\lambda} g_{00,\lambda}.$$

Con esto obtenemos:

$$\frac{d^2 x^\mu}{d\tau^2} - \frac{1}{2} g^{\mu\lambda} g_{00,\lambda} \left(\frac{dx^0}{d\tau} \right)^2 \approx 0. \quad (\text{A.6})$$

Hace falta conocer la métrica inversa $g^{\mu\lambda}$ tal que se satisfaga $g^{\mu\lambda} g_{\lambda\nu} = \delta_\nu^\mu$. Si tomamos $\eta^{\mu\lambda} h_{\lambda\nu} = \eta_{\lambda\nu} h^{\mu\lambda}$ obtendremos una aproximación de la métrica inversa debido a que las perturbaciones son muy pequeñas:

$$g^{\mu\lambda} \approx \eta^{\mu\lambda} - h^{\mu\lambda},$$

volvemos a la geodésica, y sustituimos la métrica, su inversa y desarrollando se obtiene la ecuación geodésica:

$$\frac{d^2 x^\mu}{d\tau^2} \approx \frac{1}{2} \eta^{\mu\lambda} h_{00,\lambda} \left(\frac{dx^0}{d\tau} \right)^2. \quad (\text{A.7})$$

Podemos sacar algunas conclusiones del resultado anterior, si consideramos $\mu = 0$, es decir, considerando el movimiento en el tiempo veremos que todo se anula debido a nuestra suposición de un espacio-tiempo estático. Esto es:

$$\frac{d^2 x^\mu}{d\tau^2} \approx 0.$$

Esto resulta en que la derivada de la componente temporal del 4-momento se conserva, es decir, la energía se conserva:

$$\frac{d^2 x^\mu}{d\tau^2} = \frac{1}{m} \frac{dp^0}{d\tau}.$$

Por lo tanto la condición de que el espacio-tiempo sea estático implica que la energía se conserva. Esto implica a su vez que $\frac{dx^0}{d\tau} = c \frac{dt}{d\tau} = cte$. Considerando ahora las componentes espaciales hallamos que:

$$\begin{aligned} \frac{d^2 x^i}{d\tau^2} &= -\frac{1}{2} \eta^{ii} h_{00,i} \left(\frac{dx^0}{d\tau} \right)^2 \\ &= -\frac{1}{2} \eta^{ii} h_{00,i} \left(d \frac{dt}{d\tau} \right)^2 \\ &= -\frac{1}{2} h_{00,i} c^2 \left(\frac{dt}{d\tau} \right)^2. \end{aligned} \quad (\text{A.8})$$

Si multiplicamos por $\left(\frac{d\tau}{dt} \right)^2$ obtenemos:

$$\frac{d^2 x^i}{dt^2} \approx -\frac{1}{2} c^2 h_{00,i}. \quad (\text{A.9})$$

La expresión anterior está relacionada con el campo gravitacional dada en la expresión (A.2), ya que si elegimos $h_{00} = (2\Phi)/c^2$ tendremos:

$$\frac{d^2 x^i}{dt^2} = -\Phi_{,i}.$$

Así es como hemos hallado una relación entre el campo gravitacional y la métrica. La interpretación de esto es que una perturbación pequeña está relacionada con el potencial Φ , podemos considerar entonces que el potencial gravitacional es consecuencia de una perturbación en el espacio-tiempo.

Para trabajar con métricas más generales, partiremos de la ecuación de Poisson utilizando h_{00} :

$$\begin{aligned} \nabla^2 h_{00} &= \frac{2}{c^2} \nabla^2 \Phi \\ &= \frac{8\pi G}{c^2} \rho. \end{aligned} \quad (\text{A.10})$$

Por lo visto en la sección 2.7, sabemos que la componente 00 del tensor de energía-momento es la densidad de energía, por lo que reescribiendo la ecuación (A.10):

$$\nabla^2 h_{00} = \frac{8\pi G}{c^4} T_{00}. \quad (\text{A.11})$$

Sin embargo, esto es válido para un marco de referencia en el centro de masa de distribución de masa $M = \int \rho_m d^3x$. Necesitamos hallar esto para cualquier sistema de referencia, es decir, debemos considerar todas las componentes de $T_{\mu\nu}$. Para ello también tendremos que considerar una métrica general $g_{\mu\nu}$ en el lado izquierdo de la ecuación anterior. Para ello emplearemos el tensor de Ricci como una manera de hallar las segundas derivadas de la métrica ya que:

$$\begin{aligned} R_{00} &\approx \Gamma_{00,i}^i \approx -\frac{1}{2} \eta^{i\lambda} h_{00,\lambda i} \\ &= \frac{1}{2} \nabla^2 h_{00}, \end{aligned} \quad (\text{A.12})$$

por lo tanto, obtenemos:

$$R_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}. \quad (\text{A.13})$$

Notemos que, si sacamos la divergencia de la expresión anterior, tendremos por la conservación de la energía que el lado derecho será cero, mientras que para el lado izquierdo esto no es necesario. Por esta razón es necesario hallar otro tensor cuya divergencia sea cero y que esté relacionado con $T_{\mu\nu}$. El tensor de Einstein $G^{\mu\nu}$ soluciona, el cual se define como:

$$G^{\mu\nu} = R^{\mu\nu} - \frac{1}{2} g^{\mu\nu} R,$$

donde R es el escalar de Ricci. Este tensor satisface $\nabla^\mu G_{\mu\nu} = 0$. Finalmente obtenemos la ecuación de campo de Einstein:

$$G^{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}. \quad (\text{A.14})$$

B Distancias

Las ecuaciones de Friedmann describen la evolución del factor de escala, y por lo visto en la sección 2.4 sabemos que esta descripción está en términos de parámetros de densidad cosmológicos y el factor de Hubble. Por lo tanto, considerando la expansión del universo y con ello los constantes cambios en las distancias comóviles que hay entre 2 objetos existen muchas maneras de definir diferentes tipos de distancias. En este apartado nos inspiramos de [7] y [45].

B.0.1. Distancia propia y comóvil

En un universo que no es estático, la noción de distancia no puede corresponderse con las distancias radiales, ya que al ser dinámico, la distancia no depende de la coordenada radial r . En su lugar, uno empieza por definir la distancia propia como la distancia que mide un observador a un objeto en un tiempo fijo t_0 . Es decir, la distancia entre 2 eventos que ocurren al mismo tiempo (simultaneidad). Por otro lado, como la métrica es espacialmente isótropa, la distancia no dependerá de las direcciones θ y ϕ , por lo que las consideraremos fijas, y si en particular consideramos que $dt = 0$ y $d\theta = d\phi = 0$, obtendremos que la distancia propia en un instante t_0 de coordenada comóvil r se define como la integral de la ecuación (2.7):

$$D_p = \int_0^{s'} ds' = a(t_0) \int_0^r \frac{dr}{(1 - kr^2)^{\frac{1}{2}}}. \quad (\text{B.1})$$

La integral representa la distancia comóvil D_c :

$$D_c = \int_0^r \frac{dr}{(1 - kr^2)^{\frac{1}{2}}}, \quad (\text{B.2})$$

donde puede apreciarse que no depende del tiempo, es así que la distancia comóvil se define como la distancia entre 2 objetos que no cambia con el

tiempo debido a la expansión de Hubble. La solución a esta integral para un universo plano ($k = 0$) nos da la siguiente relación:

$$D_p = a(t_0)r, \quad (\text{B.3})$$

es claro que para este caso, la distancia comóvil es igual a la coordenada comóvil radial $D_c = r$. Es posible relacionar la distancia propia con el corrimiento al rojo, para ello hay que considerar que a medida que el universo se expande y el tiempo transcurre, un observador será capaz de ver puntos más distantes, por lo que las regiones que estaban más alejadas del universo, entrarán en contacto causal con el observador. Definimos como distancia de horizonte $s_h(t)$ a la distancia propia del punto observable más lejano en el tiempo t . Si colocamos un observador en el origen ($r = 0$), y emitimos un fotón a $t = 0$ ubicado en el horizonte de partículas a una distancia r_{hor} , entonces el fotón llegará a nuestro observador a un tiempo t . Así, la distancia propia se mide mediante la propagación de la luz, y considerando que la trayectoria geodésica de un rayo de luz es nula, $ds = 0$, se obtiene de la métrica FLRW:

$$\int_0^{r_{\text{hor}}} \frac{dr}{(1 - kr^2)^{\frac{1}{2}}} = \int_0^t \frac{cdt}{a(t)}. \quad (\text{B.4})$$

La ecuación anterior ya nos expresa que la distancia comóvil dependerá del modelo cosmológico debido a la dependencia de $a(t)$. Por lo tanto, juntando las expresiones (B.3) y (B.4), y considerando $c = 1$, obtenemos:

$$D_p = a(t_0) \int_0^t \frac{dt}{a(t)}, \quad (\text{B.5})$$

diferenciando la relación entre z y $a(t)$ (ver ecuación 2.8) se puede obtener una expresión más común en términos de z y $H(z)$:

$$D_p = \int_0^z \frac{dz}{H(z)}, \quad (\text{B.6})$$

donde $H(z)$ está asociado a cada modelo cosmológico.

B.0.2. Distancia de luz

La manera en que medimos distancias de acuerdo a la sección anterior requiere de un procedimiento un poco engorroso, ya que requiere de medir y sumar distancias comóviles diferenciales, y multiplicar cada diferencial por

el factor de escala en el instante de la medición. En la practica, existen 2 métodos para medir distancias.

- El primero se basa en comparar la medición del flujo de una fuente, con la luminosidad predicha por otras fuentes. A esto se le conoce como distancia de luz.
- El otro se basa, siguiendo la definición de [7] en la medición del ángulo subtendido sobre el cielo por una fuente extensa y su comparación con el tamaño físico del objeto, deducido por otros medios. A esto se le conoce como distancia angular.

La distancia de luz cumple la relación escrita en la ecuación (2.49):

$$D_L^2 = \frac{L}{4\pi F}, \quad (\text{B.7})$$

salvo ciertas consideraciones como que en el camino que sigue la luz desde el emisor al receptor no existen procesos de creación o absorción de fotones, ni procesos que hagan variar la energía de los fotones, podemos suponer que la luminosidad se distribuirá uniformemente sobre una superficie esférica σ , en cuya superficie se encuentra el receptor y en el centro el emisor.

Podemos determinar el número de fotones emitidos en un intervalo dt con frecuencias ν y $\nu + d\nu$ y relacionarla con la luminosidad por medio de la siguiente ecuación:

$$dN_\gamma = \frac{dL}{h\nu} dt, \quad (\text{B.8})$$

donde h es la constante de Planck. Diferenciando la ecuación (2.51) y considerando σ como la superficie de área, tendremos al sustituir en la ecuación anterior, que el flujo recibido será:

$$dF_0 = \frac{dL}{\sigma(1+z)^2}. \quad (\text{B.9})$$

para ello consideramos que $\nu_0/\nu = dt_0/dt = \lambda_0/\lambda = (1+z)$. Por otro lado, la superficie de área fija en un tiempo $t = t_0$ se obtiene utilizando:

$$\sigma = a_0 r^2 \int d\Omega,$$

con Ω el ángulo sólido. Esto da como resultado:

$$dF_0 = \frac{dL}{4\pi a_0^2 r^2 (1+z)^2}. \quad (\text{B.10})$$

Con una sencilla comparación con la ecuación (B.7) podemos determinar que:

$$D_L = a_0 r (1 + z). \quad (\text{B.11})$$

La ecuación anterior no depende del modelo cosmológico. Sin embargo, se puede determinar la coordenada comóvil radial r , cuya solución para el caso de un universo plano $k = 0$ es $r = D_p/a_0$, por lo que se obtiene:

$$D_L = D_p (1 + z) \quad (\text{B.12})$$

B.0.3. Distancia transversal y de ángulo

Por ahora hemos definido las distancias en términos de un observador y un objeto, sin embargo, para algunas situaciones es preferente considerar distancias entre 2 objetos situados a un mismo redshift y que subtienden una separación angular respecto al observador [7]. Para este caso, consideramos nuevamente la métrica FLRW a un tiempo fijo $t = t_0$ y con coordenada comóvil r constante. De nuevo, por la isotropía del espacio, la dirección es arbitraria, por lo que la separación angular θ entre 2 objetos está dada por una longitud de arco propio s_0 . La distancia transversal es exactamente la relación que hay entre esta separación angular y el arco, y está dada por:

$$s_0 = \theta D_T. \quad (\text{B.13})$$

Una expresión más general de la distancia transversal está dada a partir de la distancia de luz, ya que para un universo plano la distancia transversal y la distancia propia son iguales:

$$D_T = \frac{D_L}{1 + z}. \quad (\text{B.14})$$

Ahora cabe preguntarse cuál es el ángulo θ que subtiende a un objeto de longitud l , en este caso nos preguntamos por distancias transversales físicas que no cambian con la expansión de Hubble. Así, relacionamos la longitud de un objeto con el ángulo por medio de la distancia transversal de la siguiente manera:

$$D_T \theta = l (1 + z). \quad (\text{B.15})$$

En analogía con la longitud de arco Euclídea, definimos la distancia de ángulo como: $D_A = l/\theta$. Obteniendo:

$$D_A = \frac{D_L}{(1+z)^2}. \quad (\text{B.16})$$

O bien:

$$D_A = \frac{D_T}{(1+z)}. \quad (\text{B.17})$$

Notamos que la ecuación anterior no tiende a infinito cuando el corrimiento al rojo aumenta. Lo importante de esta distancia recae en lo siguiente: las distancias transversales miden la distancia propia entre 2 objetos que están al mismo corrimiento al rojo y que subtienden el mismo ángulo θ , mientras que la distancia de ángulo da la relación entre el tamaño transversal físico de un objeto y su tamaño angular [45]. Como las longitudes físicas no son sometidas por la expansión, y en comparación con la ecuación (B.12) (donde la distancia propia disminuye conforme z aumenta), diremos que estas longitudes aumentarán por un factor $(1+z)$. Por lo tanto, los objetos más distantes podrán verse más grandes en tamaño angular.

C Criterio para la elección de parámetros PSO

La manera en que escogimos los parámetros de diseño del algoritmo PSO para los datos de Cronómetros Cósmicos y Supernovas fue la siguiente:

- Primero fijamos el número de iteraciones y el número de partículas. Estos deben ser considerablemente mayores a comparación de las funciones test analíticas.
- El parámetro de inercia lo escogimos entre 0.9 y 1.1, dependiendo del problema. Una vez fijado estos parámetros, hacemos un programa que considere un par de **números aleatorios** entre 0 y 1. El primer elemento de este par se lo asignamos al coeficiente c_1 y el segundo elemento al coeficiente c_2 .
- Ejecutamos el programa 10 veces, de manera que en cada ejecución, se consideren diferentes coeficientes de aceleración en nuestros parámetros de diseño.
- Revisamos que los resultados obtenidos en cada ejecución no diverjan en más de un dígito.
- De los 10 resultados obtenidos para diferentes coeficientes, escogemos los parámetros c_1 y c_2 con los cuales se halló el valor más pequeño de χ^2 .
- Corremos el programa 10 veces con los respectivos valores escogidos del paso anterior y reportamos el valor más pequeño hallado durante dichas ejecuciones, también reportamos el vector de parámetros que minimizan la función fitness.

- Por último, calculamos el promedio y la desviación estándar de los parámetros obtenidos que conforman cada modelo cosmológico, así como el promedio y la desviación estándar del valor χ^2 .

D Resultados del promedio y desviación estándar

Mostraremos los resultados de las 10 ejecuciones del programa para optimizar la función χ^2 de cada uno de los modelos cosmológicos para los 2 datasets. El resultado más pequeño de la función fitness fue el reportado en la sección 6.

Sin embargo, algunos valores reportados en las tablas, confirman que los hallazgos del mínimo de la función χ^2 se encuentran en una vecindad cercana del valor más bajo reportado y que para cada uno de dichos resultados los parámetros pueden tener otros valores.

No. Ejecuciones	H_0	$\Omega_{m,0}$	χ^2
1	68.4292	0.3168	14.4031
2	68.0241	0.3391	14.7256
3	67.9585	0.3265	14.4171
4	67.8991	0.3149	14.4886
5	68.5095	0.2997	14.6255
6	68.1834	0.3212	14.3999
7	68.1877	0.3181	14.3952
8	68.0666	0.3136	14.4549
9	68.5348	0.3220	14.4909
10	68.7563	0.3090	14.4254
Σ	68.2549	0.3181	14.4826
σ	0.2863	0.0105	0.1099

Cuadro D.1: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo Λ CDM con datos de Cronómetros Cósmicos.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	χ^2
1	72.1896	0.2996	-1.3392	14.6222
2	70.8949	0.3258	-1.2850	14.2257
3	71.1631	0.3080	-1.2422	14.3288
4	73.6054	0.3065	-1.4835	14.3455
5	75.7193	0.3019	-1.5934	14.4051
6	71.5665	0.3149	-1.2579	14.2394
7	77.0975	0.2856	-1.7192	14.6565
8	74.9799	0.3124	-1.5332	14.5638
9	73.3618	0.2899	-1.2699	14.6049
10	71.7366	0.3173	-1.2879	14.2282
Σ	73.2314	0.3061	-1.4011	14.4219
σ	2.1129	0.0123	0.1685	0.1741

Cuadro D.2: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo ω CDM con datos de Cronómetros Cósmicos.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	ω_a	χ^2
1	76.0578	0.2793	-1.6633	1.5027	14.4262
2	74.4175	0.3238	-1.4271	-1.5067	14.5796
3	69.1072	0.3155	-1.1099	0.0769	14.3500
4	71.9855	0.2868	-1.2701	0.6362	14.2380
5	71.7875	0.3004	-1.3915	0.9236	14.1536
6	71.7883	0.3296	-1.2220	-0.9012	14.3359
7	70.5336	0.3224	-1.1079	-0.1385	14.6287
8	71.7545	0.3493	-1.3760	-1.0787	14.5606
9	74.9281	0.3010	-1.6582	0.9326	14.2667
10	75.5812	0.2906	-1.6223	0.8829	14.2754
Σ	72.7941	0.3099	-1.3848	0.1330	14.3814
σ	2.3111	0.0219	0.2115	1.0142	0.1614

Cuadro D.3: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo CPL con datos de Cronómetros Cósmicos.

No. Ejecuciones	H_0	$\Omega_{m,0}$	χ^2
1	70.0331	0.2963	33.2120
2	69.9970	0.2966	33.2110
3	70.0062	0.2972	33.2107
4	69.9926	0.2970	33.2109
5	70.0170	0.2962	33.2109
6	69.9812	0.2977	33.2116
7	70.0089	0.2969	33.2106
8	70.0080	0.2976	33.2115
9	69.9940	0.2973	33.2107
10	70.0161	0.2962	33.2110
Σ	70.0054	0.2968	33.2111
σ	0.0148	0.0006	0.0005

Cuadro D.4: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo Λ CDM con datos de SNIa.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	χ^2
1	69.6630	0.1767	-0.7618	32.4699
2	69.6649	0.1772	-0.7627	32.4699
3	69.6655	0.1776	-0.7632	32.4699
4	69.6661	0.1776	-0.7633	32.4699
5	69.6648	0.1773	-0.7628	32.4699
6	69.6668	0.1779	-0.7637	32.4699
7	69.6644	0.1770	-0.7623	32.4699
8	69.6639	0.1772	-0.7626	32.4699
9	69.6717	0.1802	-0.7671	32.4703
10	69.6558	0.1758	-0.7602	32.4700
Σ	69.6647	0.1774	-0.7629	32.4699
σ	0.0039	0.0011	0.0017	0.0001

Cuadro D.5: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo ω CDM con datos de SNIa.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	ω_a	χ^2
1	69.7146	0.1881	-0.7955	0.1043	32.4629
2	69.4807	0.0308	-0.6529	0.3912	32.4100
3	69.7330	0.0554	-0.6729	0.3995	32.4721
4	69.6567	0.1788	-0.7727	0.1016	32.4674
5	70.0237	0.0923	-0.7331	0.4149	32.4700
6	69.8836	0.2459	-0.8674	-0.1381	32.6687
7	69.7479	0.1495	-0.7367	0.0246	32.5042
8	69.7897	0.1857	-0.7936	0.1153	32.4664
9	69.8279	0.2437	-0.8675	-0.0732	32.6411
10	69.8700	0.0848	-0.7370	0.5769	32.4765
Σ	69.7728	0.1455	-0.7629	0.1917	32.5039
σ	0.1462	0.0760	0.0718	0.2379	0.0831

Cuadro D.6: Resultados de la optimización para 10 ejecuciones del algoritmo PSO para el modelo CPL con datos de SNIa.

No. Ejecuciones	H_0	$\Omega_{m,0}$	χ^2
1	69.9463	0.2954	47.9603
2	69.9463	0.2954	47.9603
3	69.9463	0.2954	47.9603
4	69.9463	0.2954	47.9603
5	69.9463	0.2954	47.9603
6	69.9463	0.2954	47.9603
7	69.9463	0.2954	47.9603
8	69.9463	0.2954	47.9603
9	69.9463	0.2954	47.9603
10	69.9463	0.2954	47.9603
Σ	69.9460	0.2953	47.9604
σ	9.70×10^{-6}	9.34×10^{-7}	3.22×10^{-9}

Cuadro D.7: Resultados de la optimización con 10 ejecuciones del algoritmo PSO para el modelo Λ CDM con datos de CC+SN Ia.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	χ^2
1	69.8610	0.2867	-0.9692	47.9183
2	69.8623	0.2867	-0.9693	47.9183
3	69.8614	0.2867	-0.9693	47.9183
4	69.8607	0.2867	-0.9693	47.9183
5	69.8629	0.2868	-0.9696	47.9183
6	69.8614	0.2867	-0.9693	47.9183
7	69.8610	0.2867	-0.9691	47.9183
8	69.8608	0.2867	-0.9692	47.9183
9	69.8644	0.2869	-0.9698	47.9183
10	69.8573	0.2863	-0.9680	47.9184
Σ	69.8613	0.2867	-0.9692	47.9183
σ	0.0018	0.0001	0.0005	2.46×10^{-5}

Cuadro D.8: Resultados de la optimización con 10 ejecuciones del algoritmo PSO para el modelo ω CDM con datos de CC+SN Ia.

No. Ejecuciones	H_0	$\Omega_{m,0}$	ω_0	ω_a	χ^2
1	69.7576	0.2991	-0.9724	-0.1573	47.9225
2	69.8861	0.2796	-0.9699	0.1282	47.9391
3	69.9158	0.2785	-0.9717	0.1189	47.9378
4	69.7833	0.3141	-0.9544	-0.5914	47.9125
5	69.8172	0.3014	-0.9664	-0.2809	47.9003
6	69.7711	0.3041	-0.9557	-0.3775	47.8997
7	69.8849	0.2950	-0.9762	-0.1329	47.9131
8	69.7619	0.3086	-0.9633	-0.3905	47.9042
9	69.8440	0.3112	-0.9730	-0.4303	47.9132
10	69.7801	0.3011	-0.9634	-0.2941	47.9065
Σ	69.8202	0.2992	-0.9666	-0.2408	47.9148
σ	0.0586	0.0121	0.0074	0.233	0.0142

Cuadro D.9: Resultados de la optimización con 10 ejecuciones del algoritmo PSO para el modelo CPL con datos de CC+SNIa.

Bibliografía

- [1] Adam G Riess et al. “Observational evidence from supernovae for an accelerating universe and a cosmological constant”. En: *The astronomical journal* 116.3 (1998), pág. 1009.
- [2] Saul Perlmutter et al. “Measurements of Ω and Λ from 42 high-redshift supernovae”. En: *The Astrophysical Journal* 517.2 (1999), pág. 565.
- [3] Mario Tafalla. “La expansión acelerada del Universo”. En: *Anuario del Observatorio Astronómico de Madrid,(1)* (2014), págs. 403-442.
- [4] Ke Shi, YF Huang y Tan Lu. “A comprehensive comparison of cosmological models from the latest observational data”. En: *Monthly Notices of the Royal Astronomical Society* 426.3 (2012), págs. 2452-2462.
- [5] Andrew Liddle. *An introduction to modern cosmology*. John Wiley & Sons, 2015.
- [6] Edmund J Copeland, Mohammad Sami y Shinji Tsujikawa. “Dynamics of dark energy”. En: *International Journal of Modern Physics D* 15.11 (2006), págs. 1753-1935.
- [7] Jordi Cepa. *Cosmología física*. Ediciones Akal, 2007.
- [8] Philip Bull et al. “Beyond Λ CDM: Problems, solutions, and the road ahead”. En: *Physics of the Dark Universe* 12 (2016), págs. 56-99.
- [9] Alan Heavens. “Statistical techniques in cosmology”. En: *arXiv preprint arXiv:0906.0664* (2009).
- [10] Jayanti Prasad y Tarun Souradeep. “Cosmological parameter estimation using particle swarm optimization”. En: *Physical Review D* 85.12 (2012), pág. 123008.
- [11] Luis E Padilla et al. “Cosmological parameter inference with Bayesian statistics”. En: *Universe* 7.7 (2021), pág. 213.

- [12] Purba Mukherjee, Jackson Levi Said y Jurgen Mifsud. “Neural network reconstruction of $H'(z)$ and its application in teleparallel gravity”. En: *Journal of Cosmology and Astroparticle Physics* 2022.12 (2022), pág. 029.
- [13] Ricardo Medel-Esquivel et al. “Cosmological parameter estimation with Genetic Algorithms”. En: *Universe* 10.1 (2023), pág. 11.
- [14] IR Medina. “Algoritmos bioinspirados: Una revisión según sus fundamentos biológicos”. En: *University of Manchester* (2014).
- [15] Stephen Hawking y Miguel Ortuño. *Historia del tiempo*. Vol. 21. Editorial Crítica, 1988, págs. 10-11.
- [16] Virgilio Niño. “El tiempo en la mecánica de Newton, la relatividad especial y la mecánica cuántica”. En: *Revista Colombiana de Filosofía de la ciencia* 2.5 (2001), págs. 25-34.
- [17] Jazhiel Chacón Lavanderos. “Modelos de Materia Oscura: Una Perspectiva Numérica”. Instituto Politécnico Nacional, 2018.
- [18] Nelson Leonardo Falcón Veloz. “La constante cosmológica: ¿un error de Einstein?” En: (2001).
- [19] Vesto Melvin Slipher. “The radial velocity of the Andromeda Nebula”. En: *Lowell Observatory Bulletin*, vol. 2, no. 8, pp. 56-57 2 (1913), págs. 56-57.
- [20] Edwin Hubble. “A relation between distance and radial velocity among extra-galactic nebulae”. En: *Proceedings of the national academy of sciences* 15.3 (1929), págs. 168-173.
- [21] Viatcheslav F Mukhanov. *Physical foundations of cosmology*. Cambridge university press, 2005.
- [22] Scott Dodelson y Fabian Schmidt. *Modern cosmology*. Academic press, 2020.
- [23] Delia Perlov y Alex Vilenkin. *Cosmology for the Curious*. Springer, 2017.
- [24] Alejandro Guarnizo Trilleras. “A Model-Independent Approach to Dark Energy Cosmologies: Current and Future Constraints”. Tesis doct. Universitätsbibliothek Heidelberg, 2016.
- [25] Joan Arnau Romeu. “Derivation of Friedman equations”. En: (2014).
- [26] Andrea De Simone. “submitter: Introduction to cosmology and dark matter”. En: *CERN Yellow Rep. School Proc.* 6 (2019), págs. 145-180.

- [27] Nabila Aghanim et al. “Planck 2018 results-VI. Cosmological parameters”. En: *Astronomy & Astrophysics* 641 (2020), A6.
- [28] Wendy L Freedman et al. “Final results from the Hubble Space Telescope key project to measure the Hubble constant”. En: *The Astrophysical Journal* 553.1 (2001), pág. 47.
- [29] MJ Reid et al. “The megamaser cosmology project. IV. A direct measurement of the Hubble constant from UGC 3789”. En: *The Astrophysical Journal* 767.2 (2013), pág. 154.
- [30] Peter AR Ade et al. “Planck 2015 results-xiii. cosmological parameters”. En: *Astronomy & Astrophysics* 594 (2016), A13.
- [31] Gary Hinshaw et al. “Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) observations: cosmological parameter results”. En: *The Astrophysical Journal Supplement Series* 208.2 (2013), pág. 19.
- [32] John C Mather et al. “Calibrator design for the COBE* far infrared absolute spectrophotometer (FIRAS)”. En: *The Astrophysical Journal* 512.2 (1999), pág. 511.
- [33] A Arbey y F Mahmoudi. “Dark matter and the early Universe: a review”. En: *Progress in Particle and Nuclear Physics* 119 (2021), pág. 103865.
- [34] Edvige Corbelli y Paolo Salucci. “The extended rotation curve and the dark matter halo of M33”. En: *Monthly Notices of the Royal Astronomical Society* 311.2 (2000), págs. 441-447.
- [35] N Falcón. “La Cosmología del Siglo XXI”. En: *Revisión critica del Big Bang y la Dinámica del Universo Observable. Caracas: Cosmografica CA ISBN* (2010), págs. 978-980.
- [36] A Vázquez-González y T Matos. “La materia oscura del universo: retos y perspectivas”. En: *Revista mexicana de física E* 54.2 (2008), págs. 193-202.
- [37] David Tamayo y J Alberto Vazquez. “Fourier-series expansion of the dark-energy equation of state”. En: *Monthly Notices of the Royal Astronomical Society* 487.1 (2019), págs. 729-736.
- [38] Robert J Scherrer. “Mapping the Chevallier-Polarski-Linder parametrization onto physical dark energy models”. En: *Physical Review D* 92.4 (2015), pág. 043001.
- [39] Roberto Trotta. “Bayesian methods in cosmology”. En: *arXiv preprint arXiv:1701.01467* (2017).

- [40] José M Bernardo. “Bayesian statistics”. En: *Probability and Statistics, R. Viertl, Ed 2* (1994), págs. 345-407.
- [41] Raul Jimenez y Abraham Loeb. “Constraining cosmological parameters based on relative galaxy ages”. En: *The Astrophysical Journal* 573.1 (2002), pág. 37.
- [42] Nicola Borghi, Michele Moresco y Andrea Cimatti. “Toward a Better Understanding of Cosmic Chronometers: A New Measurement of $H(z)$ at $z \approx 0.7$ ”. En: *The Astrophysical Journal Letters* 928.1 (2022), pág. L4.
- [43] Omer Farooq et al. “Hubble parameter measurement constraints on the redshift of the deceleration–acceleration transition, dynamical dark energy, and space curvature”. En: *The Astrophysical Journal* 835.1 (2017), pág. 26.
- [44] Marc Betoule et al. “Improved cosmological constraints from a joint analysis of the SDSS-II and SNLS supernova samples”. En: *Astronomy & Astrophysics* 568 (2014), A22.
- [45] David W Hogg. “Distance measures in cosmology”. En: *arXiv preprint astro-ph/9905116* (1999).
- [46] N Suzuki et al. “The Hubble Space Telescope cluster supernova survey. V. Improving the dark-energy constraints above $z > 1$ and building an early-type-hosted supernova sample”. En: *The Astrophysical Journal* 746.1 (2012), pág. 85.
- [47] Mikel Martin Barandiaran et al. “Detección de las Oscilaciones Acústicas de los Bariones en la Estructura a Gran Escala del Universo”. En: (2022).
- [48] Daniel J Eisenstein et al. “Detection of the baryon acoustic peak in the large-scale correlation function of SDSS luminous red galaxies”. En: *The Astrophysical Journal* 633.2 (2005), pág. 560.
- [49] Will J Percival et al. “Measuring the baryon acoustic oscillation scale using the sloan digital sky survey and 2dF galaxy redshift survey”. En: *Monthly Notices of the Royal Astronomical Society* 381.3 (2007), págs. 1053-1066.
- [50] Éric Aubourg et al. “Cosmological implications of baryon acoustic oscillation measurements”. En: *Physical Review D* 92.12 (2015), pág. 123516.
- [51] Stephen P Boyd y Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

- [52] Jorge Nocedal y Stephen J Wright. *Numerical optimization*. Springer, 1999.
- [53] Panos M Pardalos y H Edwin Romeijn. *Handbook of global optimization: Volume 2*. Vol. 62. Springer Science & Business Media, 2013.
- [54] Yu M Ermoliev y RJ-B Wets. *Numerical techniques for stochastic optimization*. Springer-Verlag, 1988.
- [55] Orlando de Antonio Suárez. “Una aproximación a la heurística y metaheurísticas”. En: *INGE@ UAN-Tendencias en la Ingeniería 1.2* (2011).
- [56] Sachin Desale et al. “Heuristic and meta-heuristic algorithms and their relevance to the real world: a survey”. En: *Int. J. Comput. Eng. Res. Trends* 351.5 (2015), págs. 2349-7084.
- [57] Amir Hossein Gandomi et al. “Metaheuristic algorithms in modeling and optimization”. En: *Metaheuristic applications in structures and infrastructures* 1 (2013), págs. 1-24.
- [58] R Tyrrell Rockafellar. “Lagrange multipliers and optimality”. En: *SIAM review* 35.2 (1993), págs. 183-238.
- [59] Erik Cuevas et al. *MATLAB: Computación metaheurística y bioinspirada*. Rc libros, 2021.
- [60] Mauricio Granada, Eliana M Toro y Antonio Escobar. “Optimización matemática en superficies de desempeño de redes neuronales artificiales”. En: *Scientia et technica* 3.32 (2006).
- [61] Mark A Schulze. “Linear programming for optimization”. En: *Perspective Scientific Instruments Inc., USA* (1998).
- [62] Catherine Lewis. “Linear programming: theory and applications”. En: *Whitman College Mathematics Department* (2008).
- [63] Sixto Ríos Insua. *Investigacion Operativa, Programacion Lineal y Aplicaciones*. EDITORIAL RAMÓN ARECES, 1996.
- [64] Amir Safdarian, MAHMOOD FOTUHI-FIRUZABAD y Farrokh Aminifar. “Composite power system adequacy assessment based on postoptimal analysis”. En: *Turkish Journal of Electrical Engineering and Computer Sciences* 21.1 (2013), págs. 90-106.
- [65] Dimitris Bertsimas y John N Tsitsiklis. *Introduction to linear optimization*. Vol. 6. Athena scientific Belmont, MA, 1997.
- [66] Ron Shamir. “The efficiency of the simplex method: a survey”. En: *Management science* 33.3 (1987), págs. 301-334.

- [67] S Binitha, S Siva Sathya et al. “A survey of bio inspired optimization algorithms”. En: *International journal of soft computing and engineering* 2.2 (2012), págs. 137-151.
- [68] Pravin S Game, Dr Vinod Vaze et al. “Bio-inspired Optimization: metaheuristic algorithms for optimization”. En: *arXiv preprint arXiv:2003.11637* (2020).
- [69] Ashraf Darwish. “Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications”. En: *Future Computing and Informatics Journal* 3.2 (2018), págs. 231-246.
- [70] Thomas Bäck y Hans-Paul Schwefel. “An overview of evolutionary algorithms for parameter optimization”. En: *Evolutionary computation* 1.1 (1993), págs. 1-23.
- [71] Pradnya A Vikhar. “Evolutionary algorithms: A critical review and its future prospects”. En: *2016 International conference on global trends in signal processing, information computing and communication (ICGTSPICC)*. IEEE. 2016, págs. 261-265.
- [72] Thomas Bartz-Beielstein et al. “Evolutionary algorithms”. En: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4.3 (2014), págs. 178-195.
- [73] John H Holland. “Genetic algorithms and the optimal allocation of trials”. En: *SIAM journal on computing* 2.2 (1973), págs. 88-105.
- [74] O Deepa y A Senthilkumar. “Swarm intelligence from natural to artificial systems: Ant colony optimization”. En: *Networks (Graph-Hoc)* 8.1 (2016), págs. 9-17.
- [75] Aleksandar Mratinković. *Illustrated Handbook of Particle Swarm Optimisation*. 2019.
- [76] S Aicevarya Devi y C Vijayalakshmi. “Bio inspired optimization algorithms in disaster”. En: *Procedia Computer Science* 172 (2020), págs. 176-180.
- [77] Andries P Engelbrecht. *Computational intelligence: an introduction*. John Wiley & Sons, 2007.
- [78] Olympia Roeva, Stefka Fidanova y Maria Ganzha. “InterCriteria Analysis of the Evaporation Parameter Influence on Ant Colony Optimization Algorithm: A Workforce Planning Problem”. En: *Recent Advances in Computational Optimization: Results of the Workshop on Computational Optimization WCO 2019*. Springer. 2021, págs. 89-109.

- [79] James Kennedy y Russell Eberhart. “Particle swarm optimization”. En: *Proceedings of ICNN’95-international conference on neural networks*. Vol. 4. IEEE. 1995, págs. 1942-1948.
- [80] Konstantinos E Parsopoulos y Michael N Vrahatis. “Particle swarm optimization and intelligence: advances and applications: advances and applications”. En: (2010).
- [81] Yuhui Shi et al. “Particle swarm optimization: developments, applications and resources”. En: *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*. Vol. 1. IEEE. 2001, págs. 81-86.
- [82] Soumya D Mohanty. “Particle Swarm Optimization and regression analysis-I”. En: *Astronomical Review* 7.2 (2012), págs. 29-35.
- [83] Ch Skokos et al. “Particle swarm optimization: an efficient method for tracing periodic orbits in three-dimensional galactic potentials”. En: *Monthly Notices of the Royal Astronomical Society* 359.1 (2005), págs. 251-260.
- [84] Stephen R Taylor, Jonathan R Gair y L Lentati. “Using Swarm Intelligence To Accelerate Pulsar Timing Analysis”. En: *arXiv preprint arXiv:1210.3489* (2012).
- [85] Maurice Clerc. *Particle swarm optimization*. Vol. 93. John Wiley & Sons, 2010.
- [86] Maurice Clerc y James Kennedy. “The particle swarm-explosion, stability, and convergence in a multidimensional complex space”. En: *IEEE transactions on Evolutionary Computation* 6.1 (2002), págs. 58-73.
- [87] Lester James V Miranda. *Pyswarms documentation*. 2020.
- [88] Ponnuthurai N Suganthan. “Particle swarm optimiser with neighbourhood operator”. En: *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*. Vol. 3. IEEE. 1999, págs. 1958-1962.
- [89] Andries Petrus Engelbrecht. “Particle swarm optimization: Global best or local best?” En: *2013 BRICS congress on computational intelligence and 11th Brazilian congress on computational intelligence*. IEEE. 2013, págs. 124-135.
- [90] Joaquín Izquierdo Sebastián y Silvia Carpitella. “Funciones test para optimización mono-objetivo”. En: (2018).

- [91] TMC Abbott et al. “First cosmology results using type Ia supernovae from the dark energy survey: constraints on cosmological parameters”. En: *The Astrophysical Journal Letters* 872.2 (2019), pág. L30.
- [92] G Steigman, TP Walker y A Zentner. “Global constraints on key cosmological parameters”. En: *arXiv preprint astro-ph/0012149* (2000).
- [93] Adam G Riess et al. “Type Ia supernova discoveries at $z > 1$ from the Hubble Space Telescope: Evidence for past deceleration and constraints on dark energy evolution”. En: *The Astrophysical Journal* 607.2 (2004), pág. 665.
- [94] Ricardo Chávez et al. “Constraining the dark energy equation of state with H II galaxies”. En: *Monthly Notices of the Royal Astronomical Society* 462.3 (2016), págs. 2431-2439.
- [95] Neta A Bahcall, Lori M Lubin y Victoria Dorman. “Where is the dark matter?” En: *The Astrophysical Journal* 447.2 (1995), pág. L81.
- [96] Licia Verde et al. “The 2dF Galaxy Redshift Survey: the bias of galaxies and the density of the Universe”. En: *Monthly Notices of the Royal Astronomical Society* 335.2 (2002), págs. 432-440.
- [97] M Maltoni y T Schwetz. “Testing the statistical compatibility of independent data sets”. En: *Physical Review D* 68.3 (2003), pág. 033020.
- [98] Daniel Moshe Scolnic et al. “The complete light-curve sample of spectroscopically confirmed SNe Ia from Pan-STARRS1 and cosmological constraints from the combined pantheon sample”. En: *The Astrophysical Journal* 859.2 (2018), pág. 101.
- [99] Alejandro A. Morales Sánchez. “Algoritmos genéticos y sus aplicaciones a la cosmología”. Tesis doct. Universidad Nacional Autónoma de México, 2022.
- [100] Saúl Ramos-Sánchez. *Relatividad para futuros físicos*. Vol. 1. CopIt ArXives, 2018.