



Universidad Nacional de La Plata
Facultad de Ciencias Astronómicas y Geofísicas

Tesis para obtener el grado académico de
Licenciatura en Astronomía

ESTUDIO Y COMPARACIÓN DE CÓDIGOS PÚBLICOS
PARA EL ANÁLISIS DE DATOS COSMOLÓGICOS

AUTOR: Martín Miguel Gamboa Lerena

DIRECTORA: Claudia Graciela Scóccola

LA PLATA, ARGENTINA
- FEBRERO DE 2016 -

Agradecimientos.

Sentir la necesidad de agradecer en tan importante trabajo demuestra sólo una cosa: que esto no es mérito de quién escribe. Por el contrario, orgullosos o no es responsabilidad de a quienes agradezco. De ninguna manera puede hablarse de méritos personales cuando la institución y el sistema en cuál uno vive posibilita y lo favorece. Por otro lado, no puedo dejar de nombrar, como dice un amigo, que también es consecuencia de pertenecer a una clase absolutamente privilegiada. Una clase que, ante tantas injusticias que hoy existen, tiene la posibilidad de centrar su horizonte y proyectar en torno a educarse y formarse (por más que después salga cualquier cosa).

En esa línea, en primer lugar quiero agradecerle al sistema público de enseñanza. Desde la primaria, pasando por la secundaria y finalmente por la educación universitaria. Soy un producto de ello. Al pueblo argentino que a través de sus impuestos ha costeado mis estudios, me ha dado becas de ayuda económica, etc.

Por otro lado, agradecerle a Claudia. Desde que le mandé el primer mail de contacto hasta estos días me enseñó muchísimo y fundamentalmente me tuvo una paciencia y dedicación infinita.

A la Agrupación Estudiantil INTI, ese grupo de personas comprometida con causas súper dignas, por su trabajo transparente, por sus preocupaciones, por enseñarme a luchar con convicción contra las desigualdades del sistema en el que vivimos, por la enorme enseñanza política que me dejó y me seguirá dejando.

A toda la gente que le puso el cuerpo y cabeza a levantar y ver nacer al ENEA. Me demostraron que con esfuerzo, compromiso, seriedad, diversión, se puede construir en el ámbito que elegí para vivir.

A las personas de la Facultad que me enseñaron que la Astronomía no sólo es la que está en las curriculas.

A la familia Fushimi/Spinelli: Alejandro y Marita fundamentalmente por su infinita ayuda. Cuando digo que sin ellos no estaría donde estoy, no miento, y por supuesto a toda esa gran familia.

A mi familia platense. Este grupo de personas que tuvieron la valentía de bancarme, la paciencia de enseñarme, la calidez de escucharme, la atención para responderme. A ellos les debo todo.

Un párrafo aparte le voy a dedicar a esa persona que estuvo siempre a mi lado y que me aceptó con todos mis errores y locuras. Gracias por todos esos momentos vividos, por tu paciencia, por tu cariño, por todas las charlas y por elegirme para crecer juntos... Gracias por todo Kei.

Por último quiero agradecer a mi familia. En particular a 5 personas que le pusieron más que el pecho a las situaciones difíciles a lo largo de estos 8 años, pero que no dejaron de alentarme, preocuparse y darme todo el apoyo necesario para que pueda terminar la carrera. Estoy hablando de mi papá y mi mamá que me apoyaron desde el principio en todo, todo; y de Javi, Guille y Marce por su ayuda y apoyo incondicional.

Prefacio

Este trabajo de Tesis es presentada como parte de los requisitos necesarios para obtener el grado académico de Licenciado en Astronomía de la Universidad Nacional de La Plata. Contiene resultados de estudios desarrollados bajo la dirección de la Dra. Claudia G. Scóccola.

Martín Miguel Gamboa Lerena.
e-mail: mgamboa@carina.fcaglp.unlp.edu.ar
La Plata, Febrero de 2016.

Resumen

El estudio del *cosmos* podría considerarse una de las primeras actividades que realizó el ser humano. Hoy en día, estudiarlo implica tener un conocimiento vasto sobre las ecuaciones perturbadas de la Relatividad General. Conocer la evolución de esas perturbaciones en el plasma primordial compuesto por una sopa de diferentes partículas es una condición necesaria para responder la pregunta ¿De dónde venimos?. Para ello, es necesario desarrollar un modelo que pueda ser falsable. El Fondo Cósmico de Radiación juega un papel fundamental en este aspecto. A partir de la medición precisa de los fotones que nos llegan desde este fondo se puede inferir la historia térmica del Universo. Desde su descubrimiento en 1965, las técnicas observacionales han mejorado notablemente, permitiendo la determinación de los observables con gran precisión. Para poder contrastar los datos observacionales con los modelos teóricos se utilizan diferentes códigos numéricos. **CLASS** y **CAMB** son códigos que resuelven las ecuaciones de Boltzmann acopladas para las diferentes especies de partículas de un modelo cosmológico. Por otro lado, los valores de los parámetros cosmológicos inciden en la forma del espectro de potencias, por lo tanto se requiere de herramientas estadísticas para poder estimar los valores teóricos a partir de las observaciones realizadas. Actualmente hay instrumentos terrestres y satelitales que realizan las mediciones. En particular, el satélite Planck, de la Agencia Espacial Europea, es la tercera generación de satélites que realiza las observaciones y cuyos conjuntos de datos se liberan periódicamente. En el año 2015 se liberó el paquete de datos que será utilizado en esta Tesis. Para el análisis estadístico existen dos códigos: **CosmoMC** y **MontePython**. **CAMB** y **CosmoMC** son códigos que se usan en el ámbito científico desde hace más de una década, **CLASS** y **MontePython** han utilizado técnicas numéricas que resuelven las ecuaciones con un grado de precisión comparable, con la ventaja de usar menos tiempo de cómputo y la flexibilidad que presenta y facilita la edición de los mismos para adaptarlos a modelos particulares. En esta Tesis nos propondremos mostrar la comparación de los diferentes códigos en el modelo estándar Λ CDM.

Índice general

Prefacio	v
Resumen	vii
1. Introducción	1
1.1. Breve reseña histórica	1
1.2. Breve historia térmica del Universo	2
2. Fondo cósmico de radiación	5
2.1. Perturbaciones primordiales	5
2.2. Descripción de CAMB	12
2.2.1. Estructura de CAMB y flujo de información	12
2.2.2. Uso de CAMB	14
2.3. Descripción de CLASS	14
2.3.1. Estructura de CLASS y flujo de información	14
2.3.2. Uso de CLASS	17
2.4. Comparación de eficiencia y resultados de C_l 's	18
3. Análisis estadístico	27
3.1. Introducción a métodos bayesianos y cadenas de Markov	27
3.2. Descripción de MontePython	31
3.2.1. Lógica del programa	31
3.3. Descripción de CosmoMC	32
3.4. Comparación de resultados y rendimiento	33
4. Conclusiones	45
A. Relaciones matemáticas	47
B. Archivo de precisión de CLASS	49
Bibliografía	51

Capítulo 1

Introducción

Sonaba como los mosquitos en verano, aunque no era verano. Aquella noche de 1964, Arno Penzias y Robert Wilson no podían trabajar en paz. Desde una cresta de los montes Apalaches, los dos astrónomos estaban tratando de captar las ondas emitidas por quién sabe qué lejanísima galaxia, pero la antena les devolvía un zumbido que les atormentaba los oídos. Después, se supo. El zumbido era el eco de la explosión que había dado origen al universo. Aquella vibración de la antena no venía de los mosquitos, sino del estallido que había fundado el tiempo y el espacio y los astros y todo lo demás. Y quizá, quién sabe, digo yo, el eco estaba todavía ahí, resonando en el aire, porque quería ser escuchado por nosotros, terrestres personitas, que también somos ecos de aquel remoto llanto del universo recién nacido.

Eduardo Galeano, Bocas del Tiempo.

1.1. Breve reseña histórica

El Fondo Cósmico de Radiación (FCR) fue descubierto en el año 1965 por Penzias, A. y Wilson, R. [1] en los laboratorios Bell mientras realizaban observaciones de la Vía Láctea[2]. Los dos radioastrónomos detectaron un exceso constante de 3.5 K en la temperatura de antena, en todas direcciones del cielo.

Después del Big Bang (BB) el Universo estaba compuesto de un plasma caliente de diferentes partículas. A medida que se fue expandiendo, la temperatura del mismo disminuía. Aproximadamente 380000 años después del BB, la temperatura desciende por debajo de los ~ 3000 K y los electrones comienzan a combinarse con los protones formando H neutro, en este momento el Universo se hace transparente; esto es, los fotones del plasma primordial que estaban fuertemente acoplados con los electrones (a través de la dispersión Thomson) comienzan a viajar libremente por el espacio en todas las direcciones y se detecta hoy como el FCR. Esta etapa se la conoce como *recombinación*. Este desacople deja que los fotones viajen libremente y con muy pocas interacciones con los bariones. Por lo tanto, la forma y distribución que poseían al momento de desacople puede observarse en la radiación que nos llega de ese momento. También queda plasmada esta información en la distribución de materia a gran escala. La radiación se corresponde con gran precisión a la de un cuerpo negro con una temperatura de $T \sim 2.7255 \pm 0.0006$ K[3], debido al corrimiento al rojo que sufren por la expansión del Universo.

El FCR nos brinda información de los primeros momentos del Universo. Hacia fines de 1980 se lanzó la primera de las, hasta ahora, tres misiones espaciales. El satélite COBE (*COsmic Background Explorer*) fue el encargado de medir la temperatura del FCR a distintas escalas angulares, encontrando fluctuaciones de temperatura del orden de 10^{-5} K, confirmando lo que predecía la

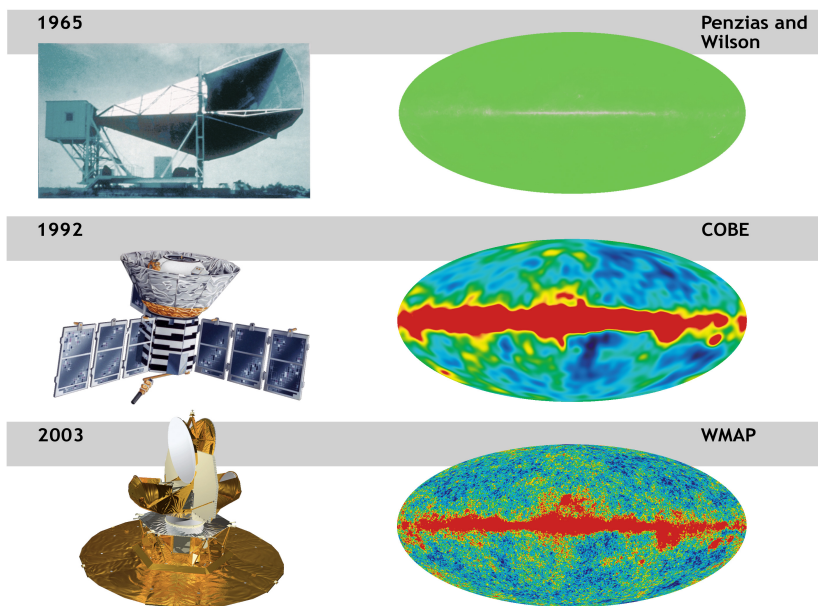


Figura 1.1. Cómo fue evolucionando la precisión y la resolución con la que se midieron las anisotropías del fondo cósmico de radiación desde su descubrimiento hasta el satélite WMAP. Fuente <http://cosmictimes.gsfc.nasa.gov>.

teoría. Además, gracias a su resolución angular, se pudo determinar el primer pico acústico. Estos picos son producto de las oscilaciones acústicas que se daban en el plasma primordial cuando fotones y bariones permanecían acoplados. Estas oscilaciones son producto de la expansión del plasma (como consecuencia de la presión de los fotones) en las regiones donde había grandes densidades del plasma y la atracción gravitatoria entre los bariones. Estas oscilaciones se congelaron al momento del desacople y hoy pueden medirse. La siguiente misión fue la WMAP (*Wilkinson Microwave Anisotropy Probe*) que midió con mayor detalle las anisotropías en la temperatura (ver Figura 1.1) y por último el satélite Planck. Este último actualmente está en funcionamiento y posee una gran precisión y resolución angular (ver Figura 1.2).

Los fotones liberados al momento del desacople atraviesan diferentes regiones y son afectados por múltiples efectos hasta que llegan a nosotros. Esos efectos se consideran al momento de hacer los mapas de las anisotropías en la temperatura y polarización. Para ello se observa en diferentes frecuencias (Figura 1.3), se modelan estos distintos efectos que son luego restados a los datos dando como resultado el mapa observado 1.2. Algunos de estos efectos son *Sunyaev-Zel'dovich*, *Sachs-Wolfe* y *Sachs-Wolfe integrado*.

1.2. Breve historia térmica del Universo

Se pueden desarrollar diferentes modelos teóricos sobre lo que había antes del desacople entre fotones y bariones pero casi no disponemos de datos observacionales directos para contrastarlos.

Se cree que 10^{-36} segundos luego del BB la gravedad comienza a ser una teoría clásica en un plasma con pequeñas inhomogeneidades. 10^{-32} segundos después del BB se produce un proceso inflacionario a energías de $\sim 10^{15} - 10^{16} \text{ GeV}^1$, este podría ser el momento donde se rompe la teoría de gran unificación. Esta etapa soluciona grandes conflictos que se presentan en el modelo del BB. A estas energías todas las partículas son tratadas como ultrarelativistas.

¹ $1 \text{ GeV} = 1.16 \times 10^{13} \text{ K}$

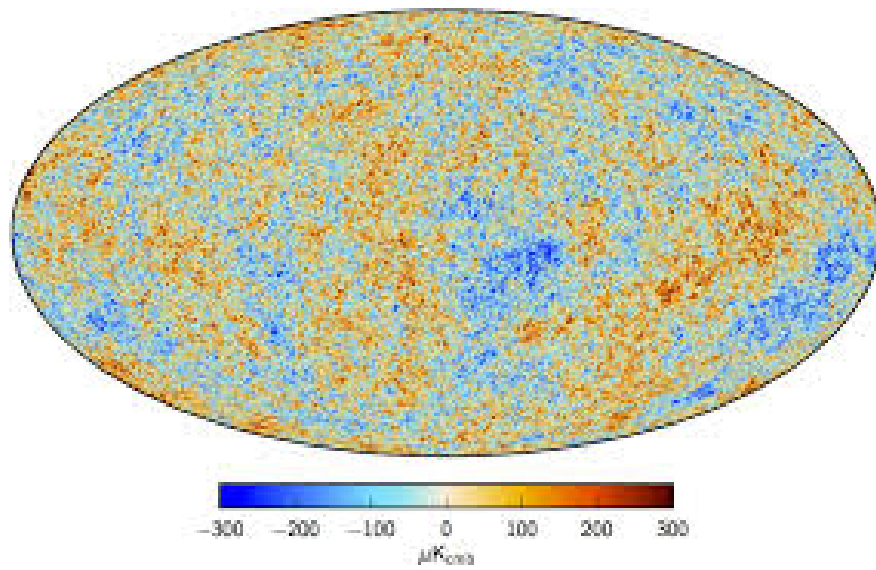


Figura 1.2. Anisotropías del fondo cósmico de radiación medida con la última misión satelital Planck. Fuente: arXiv:1502.01582.

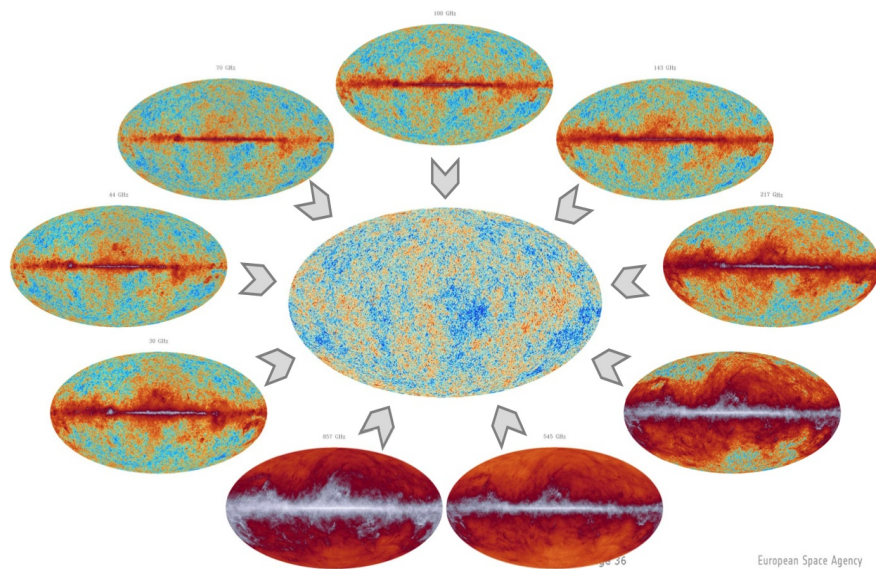


Figura 1.3. Diferentes mapas, en diferentes frecuencias del fondo cósmico. Se pueden combinar para hacer un único mapa de las anisotropías. Fuente: <http://perso.telecom-paristech.fr>.

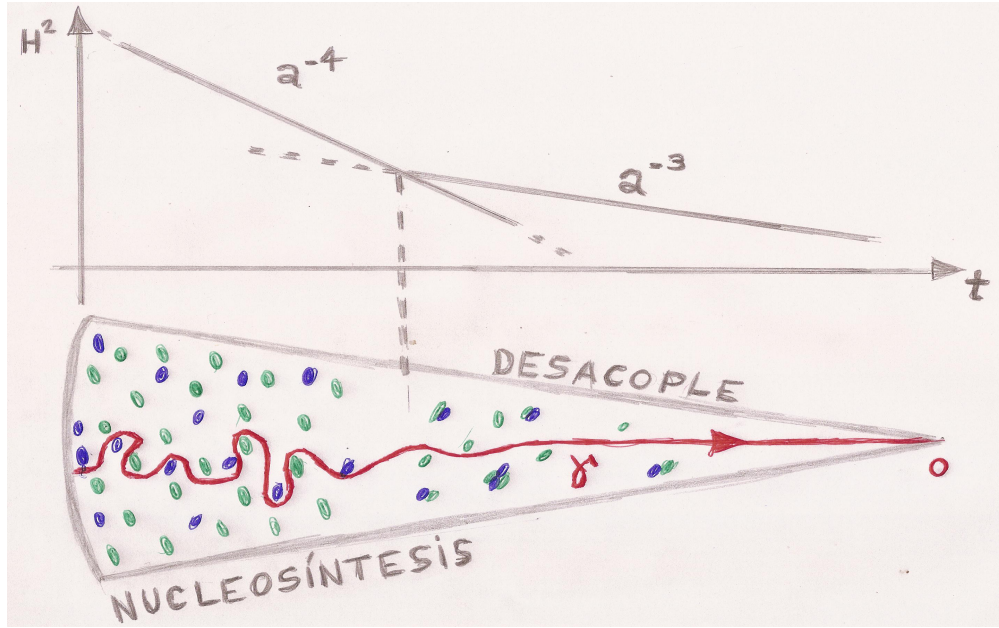


Figura 1.4. En verde y azul se representa a los bariones y la línea roja el camino de un fotón. Se esquematiza el acople que había previo a la recombinación y luego el camino libre del fotón después del desacople (Basado en esquema de Julien Lesgourgues).

Luego viene una etapa dominada por la radiación donde la temperatura y densidad evolucionan como $T \propto a^{-1}$ y $\rho \propto a^{-4}$, respectivamente. Aproximadamente a los 10^{-4} segundos se forman los primeros hadrones². Cuando el universo desciende a temperaturas de $T \sim 10$ MeV estaba compuesto de p, n, e^+, e^- , seis especies de neutrinos, γ y materia oscura.

Cuando la temperatura del plasma cae por debajo del MeV, los neutrinos dejan el equilibrio y se desacoplan del plasma. Su función de distribución se congela. Para cada especie individual, desacoplada de las otras, su momento, por efecto del corrimiento al rojo, evoluciona de la forma $p(t) = p(t_D)a(t_D)/a(t)$, donde t_d es el tiempo de desacople de la especie. Considerando las abundancias relativas, cuando la temperatura del plasma cae por debajo de $T \sim 0.07$ MeV comienza la etapa de la Nucleosíntesis Primordial. En esta etapa se forman los primeros núcleos atómicos como el deuterio, ^3He y ^7Li .

Luego de la Nucleosíntesis Primordial el universo está compuesto por fotones relativistas, bariones no relativistas y neutrinos. A $T \ll \text{MeV}$ las interacciones débiles son ineficientes y comienza a tomar relevancia la interacción electromagnética. Los fotones permanecen fuertemente acoplados a los electrones por efecto de la dispersión de Compton, y los electrones a los núcleos por efecto de la dispersión de Coulomb. Estas interacciones son eficientes mientras el H permanezca ionizado.

De la misma forma que en la Nucleosíntesis Primordial, considerando las abundancias relativas de los elementos, la recombinación se da a la temperatura de 0.254eV. Comienzan a formarse los primeros átomos de H neutro. Esta etapa se conoce como recombinación y ocurre a $z \sim 1080$. La recombinación genera el desacople entre los fotones y los bariones (ver imagen 1.4), y es importante estudiarla pues controla los fotones del FCR que observamos. Por este acoplamiento que había entre materia y radiación, se puede estudiar la polarización generada en estos fotones. Los cálculos muestran que cuando ocurre el desacople, $z_{dec} = z_{rec} = 1080$, la edad del Universo es de 380000 años.

²Partícula sub-atómica formada por la unión de quarks. Los quarks son una de las especies de partículas que forman la materia.

Capítulo 2

Fondo cósmico de radiación

A partir del descubrimiento del FCR comenzaron a utilizarse técnicas más precisas para la medición de las anisotropías. El desarrollo tecnológico e instrumental requirió nuevas técnicas numéricas para realizar cálculos precisos y sistemáticos y así poder comparar los modelos teóricos con las observaciones.

En 1995 Bertschinger escribe el código COSMICS en `fortran 77`[4]. El mismo resolvía las ecuaciones de Boltzmann (para neutrinos y fotones) que Ma&Bertschinger [5] demuestran para el *gauge* sincrónico y el modelo de recombinación de Peebles hasta un $l_{\max} \sim 2500$. Luego, en 1996, Seljak&Zaldarriaga desarrollan una extensión al anterior, llamado CMBFAST, agregando nuevas funciones de transferencia y las convoluciones con las funciones de Bessel[6].

En 1999 se desarrolla el código RECFAST[7], un modelo para la recombinación mucho más rápido pero con una estructura complicada para leer. Ese mismo año, Lewis, Challinor y Lasenby[8] reorganizan CMBFAST, implementan herramientas computacionales para mejorar su rendimiento y agregan efectos nuevos a la cosmología creando un nuevo código más moderno, flexible a cambios, y que actualmente es uno de los dos que se mantiene actualizado: CAMB.

Más adelante, en 2011, Julien Lesgourgues desarrolla un nuevo código más transparente, fácil de modificar y estructurado en forma modular: CLASS[9].

Comenzaremos este capítulo haciendo un breve resumen de las ecuaciones y la cosmología que resuelven estos códigos. Luego contaremos brevemente el flujo que poseen ambos códigos, CAMB y CLASS, y finalmente mostraremos los resultados a los que llegamos en la comparación entre los códigos.

2.1. Perturbaciones primordiales

Las ecuaciones que se encuentran en los códigos que veremos en las secciones 2.2 y 2.3 se basan en el trabajo de Ma&Bertschinger [5] por lo que contaremos brevemente las mismas.

En promedio la materia en el Universo tiene carga neutra, lo que hace que la única interacción entre las partículas se dé a través de la gravedad. En ese sentido una teoría de gravedad, la Relatividad General, es la apropiada para estudiar la cosmología. Las ecuaciones de campo de Einstein relacionan la geometría del espacio-tiempo, determinada por la métrica, con la energía que se encuentra en esa porción del espacio-tiempo.

A gran escala se considera que el Universo es homogéneo e isótropo. Esta aproximación es válida a partir de definir una escala en la cuál se observa que la materia podría describirse como fluido. Una prueba de esto es el FCR que observamos. Este modelo de universo se denomina universo de Friedmann-Lemaître-Robertson-Walker (FLRW). La teoría de perturbaciones lineales del universo FLRW fue desarrollada por Lifshitz (1946), posteriormente revisada y luego publicada en libros de texto de Weinberg (1972) y Peebles (1980). Particularmente, una de las aplicaciones de la teoría de

2. Fondo cósmico de radiación

perturbaciones lineales de la Relatividad General es el FCR. En teoría de perturbaciones lineales se trabaja a primer orden perturbativo en la métrica, o sea

$$g_{\mu\nu} = \bar{\eta}_{\mu\nu} + a^2 h_{\mu\nu}, \quad (2.1)$$

donde $\bar{\eta}$ representa la métrica sin perturbar (Minkowsky) y $h_{\mu\nu}$ es la perturbación. La no especificación del sistema de coordenadas en la cuál esa perturbación es *chica* deja cierta libertad para elegir ese sistema de coordenadas en el espacio-tiempo. Puede haber más de un sistema de coordenadas donde la métrica se escriba como Minkowsky más un término perturbativo. Dicho término puede no ser igual de un sistema de coordenadas a otro, o sea, la descomposición de la métrica en esos términos no es única. Esta libertad es la que conocemos como libertad de *gauge*. La 4-velocidad de un observador define un marco de referencia, entonces podemos decir que la elección del *gauge* es equivalente a hablar de la elección del marco de referencia.

Al marco de referencia en el cuál la materia oscura está en reposo se lo conoce como *gauge* sincrónico. Si utilizamos el referido al marco comóvil, será el *gauge* comóvil. Por otro lado, existe el *gauge* Newtoniano que surge como solución al problema de singularidad de coordenadas que presenta el *gauge* sincrónico. El *gauge* Newtoniano se utiliza para estudiar las perturbaciones escalares.

El elemento de línea es, para el *gauge* Newtoniano:

$$ds^2 = a^2(\tau) \left[- (1 + 2\psi)d\tau^2 + (1 - 2\phi)dx^i dx_i \right],$$

donde ψ y ϕ son potenciales escalares que caracterizan la perturbación. El *gauge* Newtoniano puede aplicarse sólo a los modos escalares de la perturbación en la métrica.

Por otro lado, para un universo homogéneo de FLRW con densidad de energía $\bar{\rho}(\tau)$ y presión $\bar{P}(\tau)$, las ecuaciones de Einstein dan la evolución del factor de escala $a(\tau)$:

$$\begin{aligned} \left(\frac{\dot{a}}{a}\right)^2 &= \frac{8\pi}{3} G a^2 \bar{\rho} - \kappa, \\ \frac{d}{d\tau} \left(\frac{\dot{a}}{a}\right) &= -\frac{4\pi}{3} G a^2 (\bar{\rho} + 3\bar{P}), \end{aligned}$$

donde los puntos son derivadas respecto a τ , y κ es positivo, cero o negativo para universos cerrado, plano o abierto, respectivamente. Puede definirse el corrimiento al rojo z de un objeto como $z = a(t_0)/a(t_e) - 1$, donde $a(t_0)$ es el factor de escala hoy y $a(t_e)$ es el factor de escala al momento de emisión de la luz del objeto que medimos. Por otro lado, se define el parámetro de Hubble como $H(t) = \dot{a}(t)/a(t)$. Este parámetro suele expresarse en unidades adimensionales: $h(t) = H(t)/100 \text{ km seg}^{-1} \text{ Mpc}^{-1}$.

A partir de una métrica pueden medirse distancias utilizando distintos métodos:

- a partir del corrimiento al rojo;
- a partir del diámetro angular. Hay objetos en el Universo que poseen un tamaño físico que puede conocerse. A partir de allí se calcula la distancia de diámetro angular d_A como el cociente entre el diámetro físico dl y diámetro angular del objeto $d\theta$: $d_A = dl/d\theta$. Esto es válido en un espacio Euclideo, por eso se adopta como una medida de distancia;
- a partir de la luminosidad de un objeto. Se conoce como distancia de luminosidad.

Las ecuaciones de Einstein se pueden escribir a primer orden en teoría de perturbaciones igualando el desarrollo a primer orden en la métrica, con el desarrollo a primer orden en el tensor energía-impulso $T_{\mu\nu}$.

El tensor energía-impulso tiene en cuenta las diferentes especies que entran en un modelo: fotones, bariones, materia oscura (fría y caliente), neutrinos, y otras. Para conocer la función de distribución para cada especie de partículas, se resuelven las ecuaciones de Boltzmann con encuentros entre las distintas especies. Los neutrinos interactúan débilmente con las otras partículas (a primer orden se puede considerar que no hay interacción), mientras que los electrones y los fotones interactúan por dispersión de Thomson (se encuentran acoplados) antes de que se produzca la Recombinación.

En términos de las variables (x^i, q, n_j, τ) donde x^i representa la posición, q es la magnitud del momento, n_j la dirección del momento y τ el tiempo, se escribe la perturbación a la función de distribución f en el espacio de las fases como

$$f(x^i, q, n_j, \tau) = f_0(q)[1 + \Psi(x^i, q, n_j, \tau)]. \quad (2.2)$$

Por otro lado, la evolución de f viene determinada por la solución de la ecuación de Boltzmann

$$\frac{Df}{d\tau} = \frac{\partial f}{\partial \tau} + \frac{dx^i}{d\tau} \frac{\partial f}{\partial x^i} + \frac{dq}{d\tau} \frac{\partial f}{\partial q} + \frac{dn_i}{d\tau} \frac{\partial f}{\partial n_i} = \left(\frac{\partial f}{\partial \tau} \right)_C, \quad (2.3)$$

donde el término de la derecha representa el término colisional, que depende del tipo de interacción entre las especies. Una forma conveniente de trabajar con esta ecuación es pasando al espacio de Fourier.

A partir de la conservación de la energía se puede encontrar que la densidad del universo evoluciona proporcional al factor de escala $a(t)$, que da idea del tamaño del Universo

$$\rho \propto a(t)^{-3(1+w)} \begin{cases} a(t)^{-3}, & w = 0 \text{ materia sin presión} \\ a(t)^{-4}, & w = 1/3 \text{ radiación} \\ \text{constante}, & w = -1 \text{ energía de vacío} \end{cases} \quad (2.4)$$

por lo que se ve que al principio dominaba la radiación y, a medida que crece el factor de escala, domina la materia y por último el vacío. El hecho de que la radiación caiga como $a(t)^{-4}$ tiene que ver, por un lado por la expansión del volumen (aporta un factor de $a(t)^{-3}$) y por otro lado el corrimiento al rojo (aporta un factor $a(t)^{-1}$).

El término de la densidad puede descomponerse como la suma de las densidades de las diferentes especies que lo componen. Si consideramos un universo compuesto por radiación, materia oscura, materia bariónica y con densidad de vacío, podremos describir la densidad total $\Omega = 1$ (normalizada) como la suma

$$\Omega_R + \Omega_M + \Omega_k + \Omega_\Lambda = 1, \quad (2.5)$$

donde Ω_R es la densidad de radiación, Ω_M la densidad de materia (bariónica + oscura), Ω_k la densidad de curvatura (= 0 en universo plano) y Ω_Λ densidad de vacío. Estas densidades están normalizadas a la densidad crítica y todas dependen del parámetro de Hubble. Quedándonos con $\Omega_x, x \in (R, M, \Lambda)$, se pueden parametrizar quedando adimensional como $\omega_x \equiv \Omega_x h^2$. Por lo tanto la evolución en un Universo de FLRW está determinada por estos cuatro parámetros $(\Omega_R, \Omega_M, \Omega_\Lambda, H_0)$.

Los fotones del FCR tienen un espectro de cuerpo negro con una temperatura de brillo que depende sólo de la dirección de donde proviene el fotón. Sin embargo, se observan anisotropías

2. Fondo cósmico de radiación

del orden de 10^{-5} K. Ahora bien, ¿Cómo se relacionan las perturbaciones de las ecuaciones de Einstein con las anisotropías en la radiación observada? La anisotropía podemos pensarla como un campo perturbado de la forma $\Delta(x^i, \hat{n}, \tau) \equiv \Delta T/T$; a partir de la distribución de partículas para los fotones (Bose-Einstein) f , la perturbación puede escribirse, utilizando la notación de Ma&Bertschinger(1995)[5], como

$$f(x^i, q, n_j, \tau) = f_0 \left(\frac{q}{1 + \Delta} \right) \quad (2.6)$$

y, a partir de la ecuación 2.2, podemos relacionar la perturbación en la temperatura con la perturbación Ψ como

$$\Delta = - \left(\frac{d \ln f_0}{d \ln q} \right)^{-1} \Psi. \quad (2.7)$$

Para calcular la anisotropía a un dado punto del espacio-tiempo (x_i, τ) superponemos la contribución de todas las ondas planas y pasamos la perturbación Δ al espacio de Fourier:

$$\begin{aligned} \Delta(\vec{x}, \hat{n}, \tau) &= \int d^3 k \exp(i\vec{k} \cdot \vec{x}) \Delta(\vec{k}, \hat{n}, \tau), \\ &= \int d^3 k \exp(i\vec{k} \cdot \vec{x}) \sum_{l=0}^{\infty} (-i)^l (2l+1) \Delta_l(\vec{k}, \tau) P_l(\hat{k} \cdot \hat{n}). \end{aligned} \quad (2.8)$$

Los coeficientes en la anisotropía $\Delta_l(\vec{x}, \tau)$ son variables aleatorias con amplitudes y fases que dependen de las perturbaciones iniciales. Esas condiciones iniciales se podrían establecer, fijando el *gauge* Newtoniano, a través del potencial ψ . Como la evolución de la perturbación es independiente para cada modo k , se puede escribir a la perturbación como producto de su condición inicial $\psi_i(\vec{k})$ y su evolución en el tiempo $\Delta_l(k, \tau)$

$$\Delta_l(\vec{k}, \tau) = \psi_i(\vec{k}) \Delta_l(k, \tau). \quad (2.9)$$

Estas fluctuaciones ψ_i poseen una dada distribución de probabilidad, es decir, posee una probabilidad de ocurrencia A . Un tiempo después, en la misma posición, la evolución lineal transformará ese valor A en uno proporcional, αA , donde α crece linealmente. En particular, si la probabilidad inicial es Gaussiana, será Gaussiana un tiempo posterior. Por lo tanto, si asumimos condiciones iniciales Gaussianas, podremos resolver la evolución de las fluctuaciones como sus valores medios cuadráticos. Usando la función de correlación de dos puntos de ψ_i en el espacio de Fourier se obtiene

$$\langle \psi_i(\vec{k}_1) \psi_i(\vec{k}_2) \rangle = P_\psi(k) \delta_D(\vec{k}_1 + \vec{k}_2), \quad (2.10)$$

donde δ_D es la delta de Dirac y P_ψ el espectro de potencia de ψ . Los paréntesis angulares representan un promedio sobre muchas realizaciones de un dado modelo de universo. Cada realización la entendemos como la resolución de las ecuaciones para un conjunto de condiciones iniciales diferentes. Lo que nosotros observamos en nuestro Universo podemos llamarlo una realización del mismo.

Espectro angular de potencias. ¹ La anisotropía en la temperatura depende de la dirección del cielo \hat{n} en la que observamos. Por lo tanto, puede escribirse en función de los armónicos esféricos

$$\Delta(\vec{x}, \hat{n}, \tau) = \sum_{lm} a_{lm}(\vec{x}, \tau) Y_{lm}(\hat{n}), \quad (2.11)$$

cuyos coeficientes se calculan

$$a_{lm}(\vec{x}, \tau) = \int Y_{lm}^*(\hat{n}) \Delta(\vec{x}, \hat{n}, \tau) d\Omega. \quad (2.12)$$

Para el desarrollo que sigue utilizaremos las relaciones matemáticas de los armónicos esféricos que se encuentran en el Apéndice A.

En las expresiones anteriores, l se relaciona con las escalas angulares mientras que m con la orientación. Al considerar un universo isótropo, la correlación entre dos puntos separados un ángulo θ no puede depender de la orientación, por lo tanto no puede depender del número m . Matemáticamente

$$C_l \equiv \langle |a_{lm}|^2 \rangle = \frac{1}{2l+1} \sum_m \langle |a_{lm}|^2 \rangle, \quad (2.13)$$

luego como los valores a_{lm} son independientes y aleatorios, no están correlacionados, o sea,

$$\langle a_{lm} a_{l'm'}^* \rangle = C_l \delta_{ll'} \delta_{mm'}. \quad (2.14)$$

Los C_l se denominan espectro de potencia angular (teóricos). Poseen la información estadística sobre la anisotropía del FCR. Esta cantidad, que se obtiene de la teoría, se compara con la misma cantidad obtenida de las observaciones.

Teóricamente, los C_l se relacionan con la varianza de la anisotropía en la temperatura $\Delta(\hat{n})$:

$$\langle \Delta(\hat{n})^2 \rangle = \sum_l \frac{2l+1}{4\pi} C_l. \quad (2.15)$$

Por lo tanto la teoría predice valores de expectación para $\langle |a_{lm}|^2 \rangle$ a través de los C_l . Esta expresión nos dá la varianza de la anisotropía sobre diferentes realizaciones de universo con condiciones iniciales diferentes.

Observacionalmente podemos definir, para un conjunto de $\{a_{lm}\}$, el espectro de potencias \hat{C}_l como

$$\hat{C}_l = \frac{1}{2l+1} \sum_m |a_{lm}|^2.$$

La varianza en la anisotropía observada en todo el cielo es

$$\frac{1}{4\pi} \int (\Delta(\hat{n}))^2 d\Omega = \sum_l \frac{2l+1}{4\pi} \hat{C}_l \quad (2.16)$$

¹En este punto se siguieron los lineamientos en los apuntes de clase de Hannu Kurki-Suonio[10].

2. Fondo cósmico de radiación

Esta relación nos dá la varianza de la anisotropía en la temperatura sobre todo el cielo.

Para los detalles de las ecuaciones 2.15 y 2.16 ver Apéndice A.

Las perturbaciones en la temperatura se encuentran a través de los coeficientes a_{lm} de la expansión en armónicos esféricos. La perturbación total observada es composición de los diferentes efectos físicos que sufren los fotones desde que se desacoplan hasta que los observamos. Cada efecto físico puede modelarse y para hacer la predicción en la esfera celeste se toma la transformada de Fourier de la función y se proyecta en armónicos esféricos. Más precisamente, una función de \vec{x} puede escribirse como

$$f(\vec{x}) = f(x\hat{x}) = \sum_{lm} a_{lm}(x) Y_{lm}(\hat{x}), \quad \text{con } x = |\vec{x}|.$$

Por lo tanto, la proyección en la esfera será:

$$\begin{aligned} a_{lm} &= \int f(x\hat{x}) Y_{lm}^*(\hat{x}) d\Omega \\ &= \sum_k \int Y_{lm}^*(\hat{x}) f_k e^{i\vec{k}\cdot\vec{x}} d\Omega \\ &= 4\pi \sum_k \sum_{l'm'} \int f_k Y_{lm}^*(\hat{x}) i^{l'} j_{l'}(kx) Y_{l'm'}(\hat{x}) Y_{l'm'}^*(\hat{k}) d\Omega, \quad \text{por A.3} \\ &= 4\pi i^l \sum_k f_k j_l(kx) Y_{lm}^*(\hat{k}) \end{aligned} \tag{2.17}$$

donde j_l son las funciones esféricas de Bessel. A partir de allí se puede calcular el espectro angular de potencia (C_l).

Los C_l poseen codificada toda la información del modelo de Universo, que depende de los parámetros cosmológicos. Cualitativamente, los C_l reflejan lo siguiente: uno determina un dado ángulo θ entre dos direcciones diferentes y compara cuánto se parece un punto a otro. Esto último es lo que se conoce como función de correlación entre dos puntos. Ese ángulo se asocia con el momento multipolar como $\theta = \pi/l$. El ángulo θ está relacionado a una escala física de la superficie de la última dispersión. En el espacio angular, la escala correspondiente al horizonte de sonido en la época del desacople corresponde al primer pico. Esto muestra esa correlación entre los puntos. Además se observan picos en el resto de los armónicos a este primero.

La función de transferencia depende de los parámetros cosmológicos. Los datos que se obtienen a partir de la medición de las fluctuaciones del FCR se utilizan para determinar los parámetros cosmológicos y ellos se utilizan para estimar las perturbaciones iniciales. Hay distintos efectos que pueden influir en la forma de los C_l 's. Para ello podemos descomponerlo en varios términos: un término producto del efecto Sachs-Wolfe (SW), un término debido al efecto Doppler, un término del Sachs-Wolfe integrado (ISW), y finalmente tres términos correspondientes a la relación cruzada entre pares de términos. Los primeros tres están representados en la imagen 2.1.

Un último efecto que mencionaremos es la reionización. El mismo se produce a $z \sim 10$ y se corresponde al momento en el cuál empiezan a formarse las estrellas. Se produce una dispersión de los fotones del CMB con los electrones liberados en el proceso de formación estelar. La reionización reduce la amplitud global de los C_l 's. El efecto de suavizado no afecta las grandes escalas[2].

Los planos de vibración de los campos eléctricos y magnéticos resultantes de la dispersión de Thomson no producen polarización alguna. La polarización lineal observada en el FCR es generada a partir del término cuadrupolar en la anisotropía de la distribución de fotones (temperatura) en el momento del desacople de materia y radiación. Las perturbaciones pueden descomponerse en términos escalares, vectoriales y tensoriales. Las perturbaciones escalares generan los modos E de

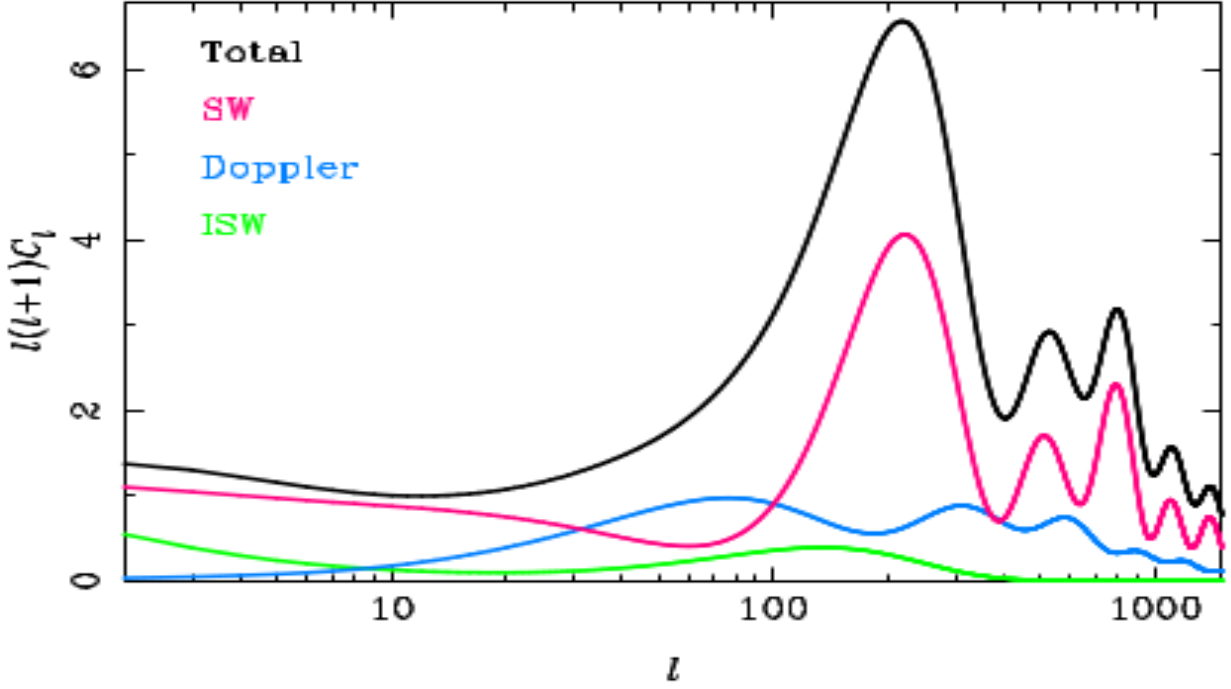


Figura 2.1. Efectos que modelan la forma de los Cl 's. Referencia [11].

polarización mientras que las perturbaciones tensoriales (ondas gravitatorias) generan los modos E y B de polarización, por lo tanto detectar los modos B implica la existencia de ondas gravitatorias primordiales siguiendo el modelo estándar[12]. Cada uno de estos modos modifica la distribución de temperaturas observada en el FCR. Los modos E y B de polarización generan patrones particulares en el FCR como se muestra en la Figura 2.2. El campo de las velocidades y las anisotropías en la T son creadas por las fluctuaciones en la densidad primordial. Por lo tanto, la componente E de la polarización debe mostrar correlación con la anisotropía en la T.

Para estudiar la polarización de un campo de radiación es necesario utilizar los parámetros de Stokes I, Q, U y V. Cada parámetro se relaciona con las componentes del campo de radiación. Ahora bien, trabajar con los parámetros de Stokes implica necesariamente definir un sistema de ejes en el cielo. El problema radica en que el valor de los parámetros varía según la elección arbitraria que se haga de dichos ejes. Por ese motivo se introducen los modos E y B de polarización que se relacionan con los parámetros de Stokes Q y U de la forma

$$E(\vec{l}) = \tilde{Q}(\vec{l})\cos(2\phi) + \tilde{U}(\vec{l})\sen(2\phi), \quad (2.18)$$

$$B(\vec{l}) = -\tilde{Q}(\vec{l})\sen(2\phi) + \tilde{U}(\vec{l})\cos(2\phi). \quad (2.19)$$

donde \tilde{Q} y \tilde{U} son las transformadas de Fourier de los parámetros de Stokes Q y U normalizados respectivamente y ϕ es el ángulo azimutal entre el vector de onda \vec{l} y el eje x . Estos parámetros son invariantes ante rotaciones.

Resolviendo la parte geométrica de las ecuaciones de Einstein perturbadas a primer orden, junto con la ecuación de Boltzmann para los fotones se encuentra la relación

$$\frac{l(l+1)C_l}{2\pi} = \int \frac{dk}{k} \Delta_l^2(k, \tau) P_\psi(k) \quad (2.20)$$

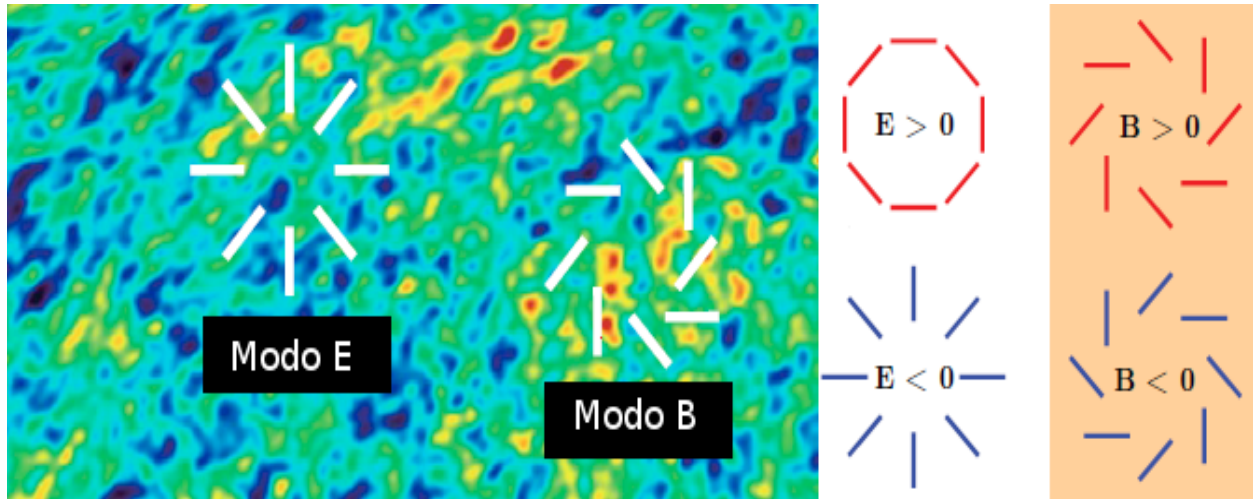


Figura 2.2. Patrones que dejan los modos de polarización alrededor de la dirección del fotón -centro de cada patrón- y que debería observarse en el FCR. Se observa que los modos E de polarización pueden ser radiales o tangenciales, mientras que los modos B forman un bucle alrededor del fotón.

En general, el espectro de potencias $P_\psi(k)$ para el modelo Λ CDM se modela como

$$P_\psi(k) = A_s \left(\frac{k}{k_0} \right)^{n_s}, \quad (2.21)$$

depende de sólo dos parámetros: A_s y n_s . Una elección del conjunto de parámetros básicos en el modelo cosmológico Λ CDM es: $A_s, n_s, \omega_b, \omega_{CDM}, \theta_s, \tau_{reio}$. Este último, τ_{reio} , es la profundidad óptica a la época de reionización. Se utiliza este modelo como base ya que la variación de estos parámetros permite controlar muchos efectos físicos que impactan en la forma de los C_l s.

Los códigos CAMB y CLASS se encargan de resolver las ecuaciones linealizadas de la Relatividad General, las ecuaciones de Boltzmann y las ecuaciones de fluido de un universo de FLRW. A partir de los modelos teóricos desarrollados más arriba, para un dado modelo cosmológico y de fluctuaciones iniciales, se pueden calcular las anisotropías del FCR en temperatura y polarización.

Ambos códigos, CAMB y CLASS, utilizan las ecuaciones del trabajo de Ma&Bertschinger (1995) [5]. CAMB es un código vastamente probado y se utiliza actualmente para realizar el ajuste de las observaciones de la misión Planck. Por lo tanto, en esta Tesis nos centraremos en estudiar más detalladamente la estructura, eficiencia y resultados de CLASS. Pese a ello hacemos un breve resumen de la estructura y funcionamiento de CAMB.

2.2. Descripción de CAMB

El Código para las Anisotropías del Fondo de Microondas o CAMB por sus siglas en inglés (Code for Anisotropies in the Microwave Background), es un código numérico creado por Anthony Lewis y Anthony Challinor [8]. Es uno de los dos códigos que se mantiene actualizado permanentemente y puede descargarse de su sitio web². Se encuentra programado en lenguaje `fortran90`.

2.2.1. Estructura de CAMB y flujo de información

CAMB puede compilarse con `fortran`, `F90`, `g95`, `f95`, `ifc` e `ifort`, compilador no libre para el cual CAMB se encuentra optimizado. El código está paralelizado. Esto no significa que puede

²<http://camb.info/>

correrse más de un modelo cosmológico al mismo tiempo, sino que realiza distintas tareas simultáneamente para obtener los resultados de un modelo cosmológico dado. Está basado en el CMBFAST de Seljak y Zaldarriaga [8].

El código está escrito en 22 (veintidos) archivos. Los mismos pueden dividirse en subgrupos según las tareas que desempeñan dentro del código. De los 22, hay:

- 3 (tres) que sirven para correr diferentes pruebas: `inidriver.f90`, `sigma8.f90` y `tester.f90`.
- 6 (seis) archivos de programa con subrutinas útiles para cualquier módulo y no poseen cosmología. No deben modificarse:
 - `bessels.f90` calcula las funciones de Bessel;
 - `inifile.f90` módulo para leer de un archivo los pares nombres/valores de los parámetros;
 - `Matrix_utils.F90` módulo con herramientas matriciales;
 - `subroutines.f90` posee rutinas numéricas generales y de precisión;
 - `utils.F90` módulo con definiciones generales y rutinas;
 - `writefits.f90` exporta los C_l 's a formato FITS para usar con HEALPix
- 12 (doce) archivos de programa con la cosmología:
 - Dos módulos para la dirección: `camb.f90` y `cmbmain.f90`;
 - Tres módulos para la recombinación: `cosmorec.F90`, `hyrec.F90` y `recfast.F90`;
 - Un módulo para la reionización: `reionization.f90`
 - Un módulo para el cálculo del espectro de potencia: `power_tilt.f90`;
 - Un módulo para el cálculo de espectro de potencia no-lineal: `halofit.f90`;
 - Un módulo para el cálculo del bi-espectro: `SeparableBispectrum.F90`;
 - Un módulo para el cálculo de lentes gravitacionales: `lensing.f90`;
 - Un módulo para calcular las ecuaciones del fondo y las perturbaciones: `equations.f90`;
 - Un módulo para calcular las ecuaciones del fondo y las perturbaciones agregándole ecuación de estado para la energía oscura variable: `equations_ppf.f90`.

El módulo `cmbmain.f90` es la rutina principal que hace las integraciones. Engloba a CMBFAST, CMBflat, CMBopen. La subrutina `cmbmain` se conecta con el módulo `camb.f90`, encargado de correr el CAMB.

Para la recombinación se utiliza por defecto RECFAST (`recfast.f90`). Si bien las tres rutinas tienen varias funciones y subrutinas en común, RECFAST viene listo para usar mientras que para usar HyRec o `cosmorec` hay que introducir nuevas librerías.

El archivo `reionization.f90` define un módulo llamado `reionization` que parametriza la historia de reionización y provee una función que da la fracción de ionización del H (X_e) como función del corrimiento al rojo. La profundidad óptica está mapeada en función del corrimiento al rojo en la era de la reionización (Z_{re} -definido cuando la X_e está a mitad de su máximo-).

Por último, las ecuaciones utilizadas en `equations.f90` y `equations_ppf.f90` están derivadas a partir de la aproximación covariante, fijas al marco de referencia de la materia oscura fría, lo que es igual a decir que estamos en el *gauge* sincrónico, y por ende se utilizan las ecuaciones en ese *gauge*.

2.2.2. Uso de CAMB

Para correr el código es necesario compilarlo usando el `Makefile` y construir las librerías, eso se hace mediante el comando `make`. Una vez realizado esto se procede a ejecutar el archivo ejecutable dándole como argumento un archivo con los parámetros cosmológicos iniciales, esto es,

```
./camb parametros.ini.
```

El archivo `params.ini` posee una detallada explicación sobre los parámetros cosmológicos y cómo pasárselos a la corrida.

Los resultados se muestran en un archivo con tantas columnas como autocorrelaciones y correlaciones cruzadas se hayan pedido. Esto implica que si nosotros le pedimos un análisis con lentes gravitacionales y considerando perturbaciones tensoriales, el resultado será

$$1 \quad C_{TT} \quad C_{EE} \quad C_{BB} \quad C_{TE} \quad C_{dd} \quad C_{dT} \quad C_{dE}, \quad (2.22)$$

donde cada C_{XX} está convenientemente normalizado. Por otro lado, genera un archivo con las funciones de transferencia, si éstas son solicitadas.

2.3. Descripción de CLASS

El Sistema de Solución para las Anisotropías Lineales del Cosmos o CLASS por sus siglas en inglés (Cosmic Linear Anisotropy Solving System), es un código numérico creado por Julien Lesgourgues [9] y actualmente se perfecciona también por los propios usuarios. El mismo está escrito en el lenguaje libre C y estructurado en forma modular.

2.3.1. Estructura de CLASS y flujo de información

Al descargar el código pueden visualizarse los siguientes directorios: `bbn/`, `cpp/`, `doc/`, `external_Pk/`, `hyrec/`, `include/`, `main/`, `output/`, `python/`, `source/`, `test/` y `tools/`. Por otro lado tendremos varios archivos de inicio `.ini`, archivos de precisión `.pre`, archivos para graficar y el `Makefile`.

A continuación detallaremos el contenido y función de los principales directorios dentro del código:

- `bbn/` posee los datos de la abundancia primordial del He (YHe) para el modelo de BBN estándar;
- `cpp/` posee el *wrapper* de C++ para CLASS;
- `doc/` documentación del código escrita por su autor;
- `external_Pk/` es un módulo que permite calcular el espectro primordial a partir del pedido de un programa externo, como por ejemplo `MontePython`;
- `hyrec/` módulo de recombinación para *HyRec*;
- `include/` contiene todos los archivos que se incluyen en los módulos principales;
- `main/` posee el módulo principal llamado `class.c` con la rutina principal que inicializa la ejecución del código;
- `output/` posee información de los archivos de salida;
- `python/` construye un entorno para usar en CLASS librerías de Python;

- `source/` contiene los archivos C para cada módulo de CLASS. Más adelante entraremos en detalle sobre a qué llamamos módulo y cuál es la función de cada uno de ellos;
- `test/` contiene archivos de prueba de cada uno de las funcionalidades del código;
- `tools/` Contiene las herramientas computacionales: algoritmos numéricos, integradores, manipulación de matrices (entre las cuales se encuentra cálculo de las funciones de Bessel), etc. Todo lo que puede ser usado en cualquier otro módulo.

Dentro del directorio `source/` encontraremos los archivos principales. Módulo es un archivo `C source/archivo_i.c` y un archivo `include/archivo_i.h`. A este conjunto de archivos lo denominaremos estructura, en este caso `estructura_i`.

La estructura modular de CLASS para calcular los C_l 's y los P_k se basa en la siguiente lógica:

1. Se le da un conjunto de parámetros iniciales;
2. Calcula las ecuaciones del fondo cosmológico;
3. Calcula la evolución en el tiempo de las cantidades termodinámicas. Para eso hay dos rutinas: HyRec o RECFAST;
4. Calcula las perturbaciones y funciones fuente $S(k, \tau)$;
5. Calcula las funciones de Bessel;
6. Calcula las funciones de transferencia, $\Delta_l(k)$ para saber cómo evolucionan las cantidades conforme pasa el tiempo;
7. Calcula el espectro de potencias primordial (para cada modo y condición inicial) escalar, vectorial y tensorial;
8. Calcula los espectros de potencia observables de las funciones fuente, funciones de transferencia y espectro primordial;
9. (Opcional para el usuario) Calcula una estimación no-lineal de los espectros;
10. (Opcional para el usuario) Calcula los espectros en la temperatura y polarización producidos por las lentes gravitacionales;
11. Calcula el espectro de potencias C_l 's y/o los $P(k)$'s.

Cada uno de los pasos enumerados anteriormente los realizan distintas estructuras y archivos dentro del código. El primero, lectura de los parámetros iniciales, no está asociado a lo que llamamos estructura. Por otro lado el cálculo de las funciones de Bessel no posee módulos. Estas son necesarias al momento de utilizar las funciones de transferencia. En las últimas versiones del código, se pasan en forma de tablas cuando son requeridas. El resto de las tareas tienen una estructura asociada.

Antes de entrar en detalle sobre cómo está pensado cada módulo asociado con cada una de esas tareas, detallaremos la lógica que atraviesa transversalmente a cada uno de ellos.

Los principales módulos se encuentran dentro del directorio `source/`. Los mismos contienen, como mínimo, dos funciones `modulo_i_init()` y `modulo_i_free()`, en ese orden. Antes de que se ejecute `_init` de la estructura correspondiente (la `estructura_i` en este caso) se carga con los parámetros de entrada (realizado por el módulo `input.c`). Luego actúan las funciones cuya funcionalidad es:

- `modulo_i_init()` función que ordena y completa la información necesaria para la ejecución del módulo. Luego de que es ejecutada, la `estructura_i` contiene toda la información relevante que los siguientes módulos van a necesitar. Hay información interna específica del módulo que no es almacenada, pero la información relevante para las otras partes del código sí es guardada. Por ejemplo, luego de la ejecución del módulo `perturbation.c`, la estructura `perturbation` posee una tabla con las funciones fuente $S(k, \tau)$.
- `modulo_i_free()` función que libera la memoria utilizada por la `estructura_i`.

Hay una tercer función, `modulo_i_variable_at_algunvalor`. Esta función la poseen los módulos: `background.c`, `thermodynamics.c`, `perturbations.c`, `transfer.c`, `primordial.c`, `spectra.c`. Sirve para interpolar cantidades precalculadas y guardadas en estructuras anteriores a la que lo necesita.

Los argumentos de la función `modulo_i_init()` serán los datos de precisión pasados a través de un archivo con sufijo `.pre`, las estructuras de los módulos previos (de 1 a $i - 1$) y los parámetros de entrada correspondientes a la `estructura_i`. Dicho esto, resumimos el archivo `main/class.c` de la siguiente forma:

```
modulo_1_init(precision, estructura_1)
modulo_2_init(precision, estructura_1, estructura_2)
...
modulo_10_init(precision, estructura_1, estructura_2, ..., estructura_10)
/***** all calculations done, now free the structures *****/
modulo_10_free(estructura_10)
...
modulo_1_free(estructura_1)
```

Anteriormente hablamos de la estructura modular de CLASS, de cómo está programado para ir calculando los parámetros en las distintas etapas de evolución del universo. Ahora detallaremos el nombre del módulo asociado a cada una de ellas:

1. El archivo `input.c` es el primer módulo que llama el código. El mismo está compuesto de dos funciones: `input_init_form_arguments()` e `input_init()`. La primera lee un máximo de dos argumentos: un archivo con los valores iniciales de los parámetros `.ini` y un archivo con los valores de precisión para cada valor `.pre` (más adelante detallaremos qué hay en uno y otro archivo). Los parámetros que no se pasen en esos archivos se completarán con los valores por defecto a través de esta función. La segunda función, es la primer función que se llama cuando se utiliza el código dentro de un código de extracción de parámetros.
2. `background.c` es el encargado de resolver la ecuación de Friedmann y de inicializar las cantidades del fondo, integrar sus ecuaciones y centralizarlas en tablas de interpolación para que los módulos siguientes puedan utilizarlas. Las cantidades del fondo están en función del tiempo (o corrimiento al rojo). Toda esa información la ordena dentro de la estructura de `background`. También provee las rutinas que permiten calcular las cantidades del fondo para los otros módulos.
3. `thermodynamics.c` al principio integra las ecuaciones de la termodinámica y ordena las cantidades termodinámicas como función del corrimiento al rojo en una tabla de interpolación (dentro de una estructura llamada `thermo`). El cálculo de la recombinación está basado en RECFAST v1.5, mientras que para la reionización utiliza la misma función del CAMB.

Este módulo también es el que provee la rutina necesaria para, cuando los otros módulos lo requieren, evaluar una cantidad termodinámica en un valor determinado de tiempo (o corrimiento al rojo).

4. `perturbations.c` resuelve la evolución de todas las perturbaciones y guarda temporalmente los términos que contribuyen a las funciones fuente en función del tiempo conforme. Esto queda guardado en la estructura `perturbs`.
Este módulo posee la rutina para calcular las funciones fuente $S(k, \tau)$ en cualquier tiempo τ exigido en alguna otra parte del código.
5. `transfer.c` calcula las funciones de transferencia $\Delta_l(k)$ y las guarda en tablas de interpolación para que usen los otros módulos. La estructura en la cuál se almacena esa información se denomina `transfers`.
6. `primordial.c` calcula los espectros de potencia primordiales y los almacena en la estructura `primordial`.
7. `spectra.c` dadas las funciones de Bessel y de transferencia (para las anisotropías), las funciones fuente (para el espectro de Fourier) y los espectros primordiales, calcula las anisotropías y el espectro de potencias de Fourier ($C_l, P(k)$). Guarda la información en la estructura `spectra`.
8. `nonlinear.c` da una estimación del espectro por efectos de evolución no lineal. Corrige los valores de los C_l y de los $P(k)$.
9. `lensing.c` calcula el espectro en la temperatura y polarización del FCR por efectos de lentes gravitacionales.
10. `output.c` escribe los archivos de salida. Este módulo no se utiliza cuando CLASS es llamado desde otro código.

2.3.2. Uso de CLASS

Para compilar el código no es necesario tener una versión específica de un compilador, no requiere un paquete ni librería especial [9]. La opción de compilador que posee el `Makefile` -archivo que construye las dependencias necesarias para la compilación y la compilación en sí misma- es en C (`gcc`), con nivel de optimización 4. Para una corrida paralelizada puede hacerse uso de OpenMP, por ejemplo `gcc -O4 -fopenmp`.

Una vez compilado el código, y creado el ejecutable `class`, se le puede pasar como argumento dos archivos como máximo. Los dos archivos que se le pueden pasar son: un archivo con los parámetros cosmológicos iniciales (con extensión `.ini`) y uno con la precisión (con extensión `.pre`). Si no se le pasa como argumento ningún archivo, el código le asigna los valores por defecto a los parámetros cosmológicos y errores.

En la distribución de CLASS hay muchos ejemplos de prueba con diferentes modelos cosmológicos. En los mismos (tanto los `.ini` como los `.pre`) podemos encontrar en detalle las diferentes características y las formas de modificarlos a fin de adaptarlos a las necesidades del usuario.

CLASS acepta distintas convenciones para los parámetros. Por ejemplo, se puede pasar el valor de la constante de Hubble como `H` o `h`, la densidad de fotones como `T_cmb`, `Omega_g` u `omega_g`, etc.

Una vez realizada la corrida tenemos, al menos, un archivo con los C_l , otro con los parámetros usados y otro con los parámetros sin usar. Luego, dependiendo de qué requerimientos le hagamos, nos agregará nuevos archivos de datos.

En resumen, el código CLASS posee una detallada explicación en cada uno de sus archivos sobre lo que hace, las funciones que utiliza, dónde debemos editarlo si queremos agregarle nuevas especies a nuestro modelo de universo, o si queremos agregar términos, cambiar la cosmología, etc. Posee una gran claridad en cuanto refiere a la edición y adaptación a nuevos escenarios.

$\Omega_b h^2 = 0.02260$	$\Omega_{CDM} h^2 = 0.11200$
$100\theta = 1.037962$	$\tau_{reio} = 0.0900$
$A_s = 2.1 \times 10^{-9}$	$n_s = 0.96$
$Y_{He} = 0.24$	$N_{eff} = 3.046$
$T_{cmb} = 2.7255$	$H_0 = 70$
<code>get_transfer = F</code>	<code>do_lensing = F</code>
<code>l_max_scalar = 4000</code>	<code>k_eta_max_scalar = 12000</code>

Tabla 2.1. Parámetros cosmológicos para la realización de corridas de comparación. Además se corrió con dos especies de neutrinos sin masa y un neutrino masivo.

2.4. Comparación de eficiencia y resultados de C_l 's

Buscamos realizar una comparación de la salida entre los códigos teniendo en cuenta tiempos de cómputo y diferencias relativas en los espectros de potencia. Por otro lado, ejemplificamos con corridas propias las variaciones que se producen en los espectros de potencia del FCR ante la variación de los parámetros cosmológicos. Tomamos el conjunto de parámetros cosmológicos que se puede ver en la tabla 2.1.

Para conocer la precisión relativa entre los códigos realizaremos una corrida con cada uno utilizando el mismo conjunto de parámetros y la misma computadora. Las versiones de los códigos utilizadas fueron: Enero2015 para CAMB y la versión 2.4.2 de CLASS. Los mismos fueron corridos en las mismas condiciones, en la supercomputadora de México en dos CPU. Se han corrido ambos códigos utilizando compilador `gfortran` para CAMB y `gcc` para CLASS y con valores para los *flags* de `-O4`.

La comparación fue realizada, para una dada historia de recombinación, entre la autocorrelación y correlación cruzada de la temperatura y el modo E de polarización sin considerar lentes gravitacionales. Se han usado los espectros de referencia utilizados en [13] y que se corresponden con la tabla 2.2. Utilizaremos el modelo básico de Λ CDM, y condiciones iniciales adiabáticas.

Las ecuaciones físicas integradas por ambos códigos son las mismas. Lo que varía de uno a otro son las herramientas computacionales y los esquemas de aproximación. Ambos códigos fueron desarrollados íntegramente en forma independiente a excepción del módulo de recombinación RECFAST.

Para CAMB se utilizaron los valores que se muestran en la Tabla 2.2 mientras que para CLASS se utilizó el archivo `lcdm.ini` que provee el mismo código y que es utilizado por los autores de [13] como espectro de referencia. Los resultados que se observan son con los parámetros de precisión de CAMB: `accuracy_boost = 12`, `l_accuracy_boost = 4`, `l_sample_boost = 3`, el resto de los parámetros de precisión está dentro del mismo código, y con los parámetros de precisión por defecto del CLASS, y que se muestran en el Apéndice B, corresponden al archivo `cl_ref.pre` que provee el código. Se observa que se tiene un mayor control sobre los errores en el código CLASS.

El primer ejercicio que realizamos fue ver cómo varía la autocorrelación de la temperatura, (C_l^{TT}), a medida que varían los parámetros cosmológicos en forma independiente. Los resultados pueden visualizarse en las figuras 2.3, 2.4, 2.5.

Se observa que la variación de cada parámetro se condice con una variación en la forma y en la posición de los picos acústicos. Las anisotropías del espectro de temperatura en el FCR son sensibles a las variaciones de los parámetros cosmológicos. Vemos que a medida que:

- crece H_0 , a partir de $l \sim 20$ el espectro se desplaza hacia la derecha. Este efecto está asociado a que la posición de los picos acústicos l_{pic} se relaciona con el horizonte de sonido en la época de desacople $l_{pic} = \pi/\theta_s$. Por otro lado, $\theta_s = r_s(t_{dec})/d_A(t_{dec})$ y $d_A \propto H_0^{-1}$. Por lo tanto,

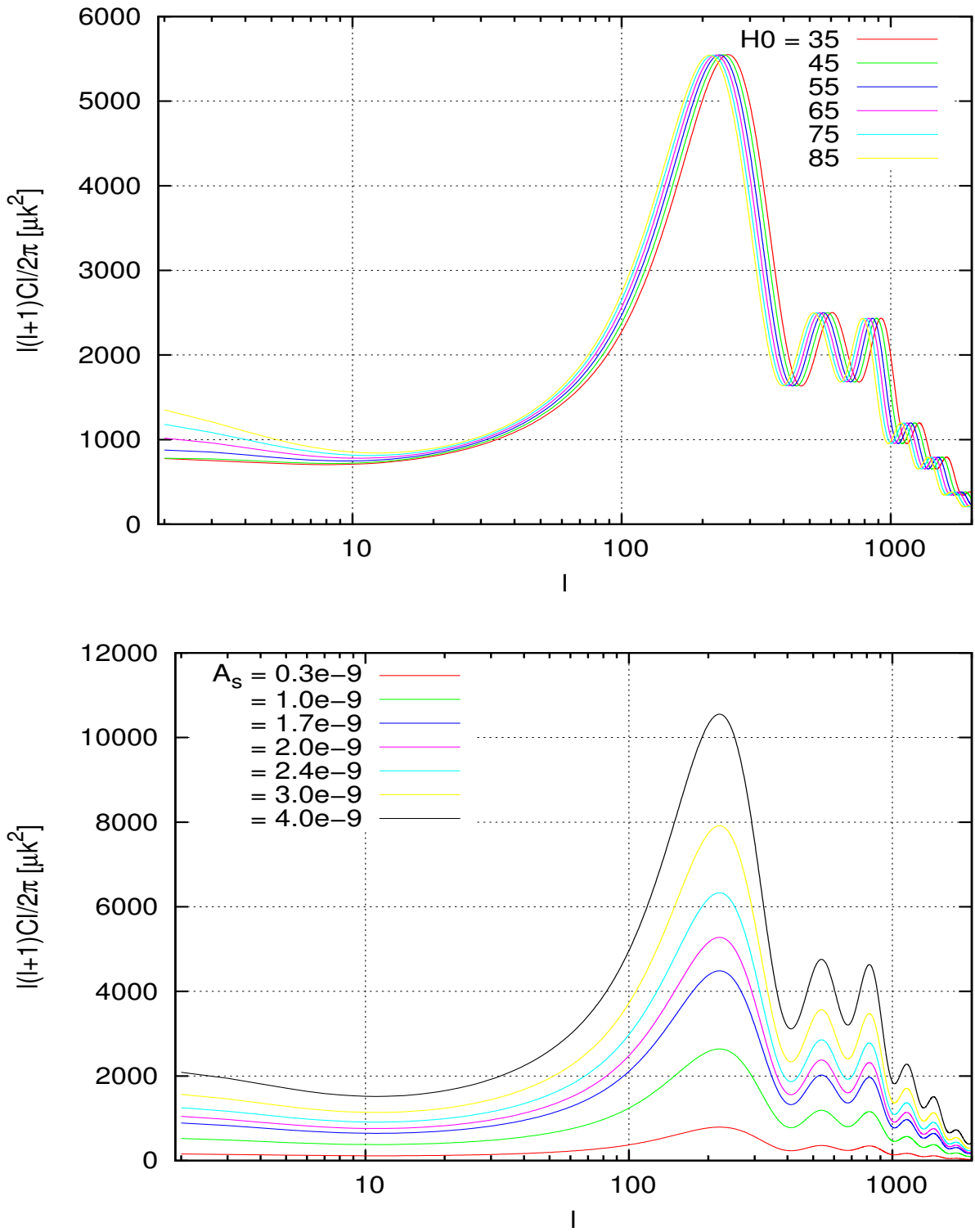


Figura 2.3. Variación de H_0 y A_s

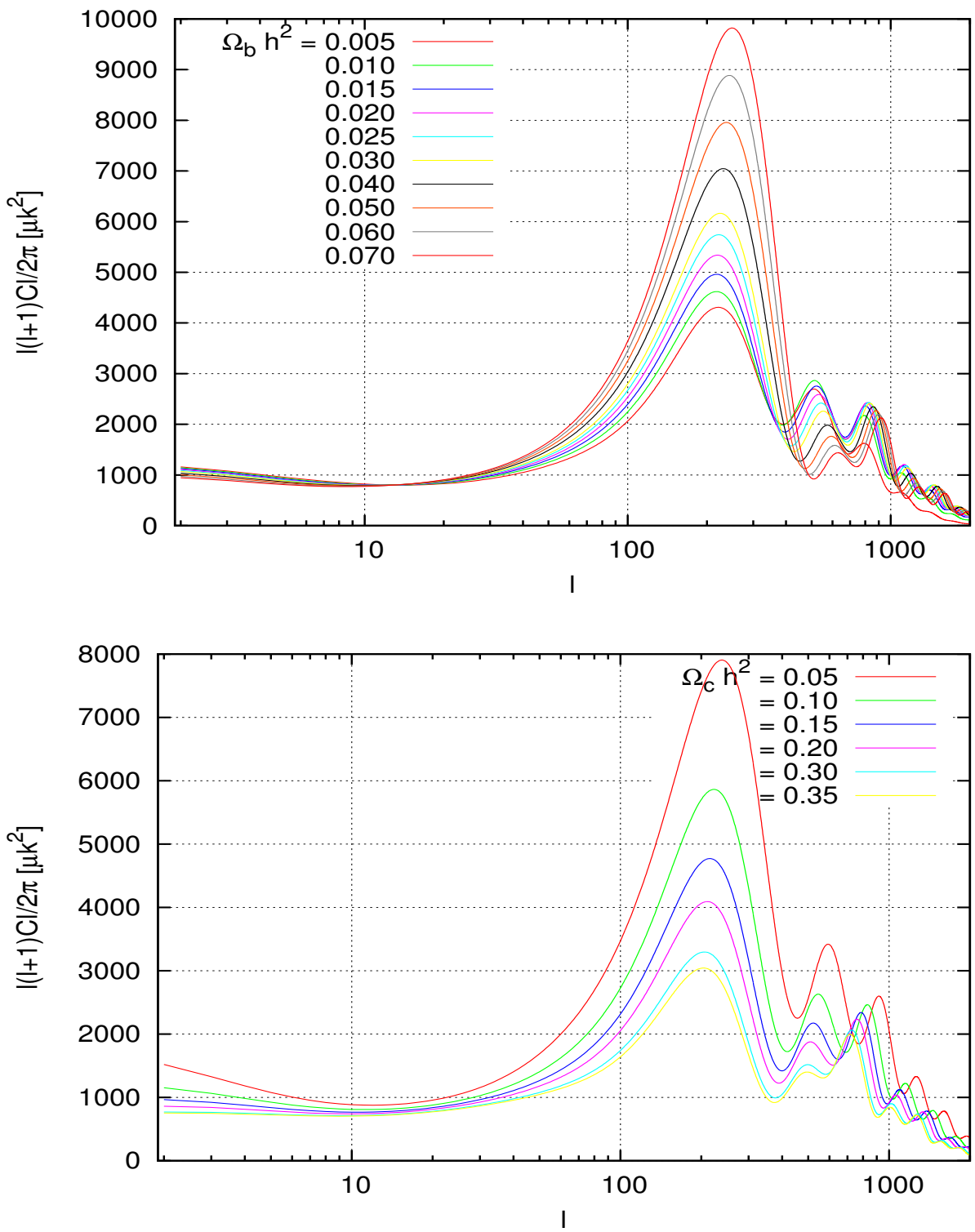


Figura 2.4. Variación de Ω_b y Ω_{CDM}

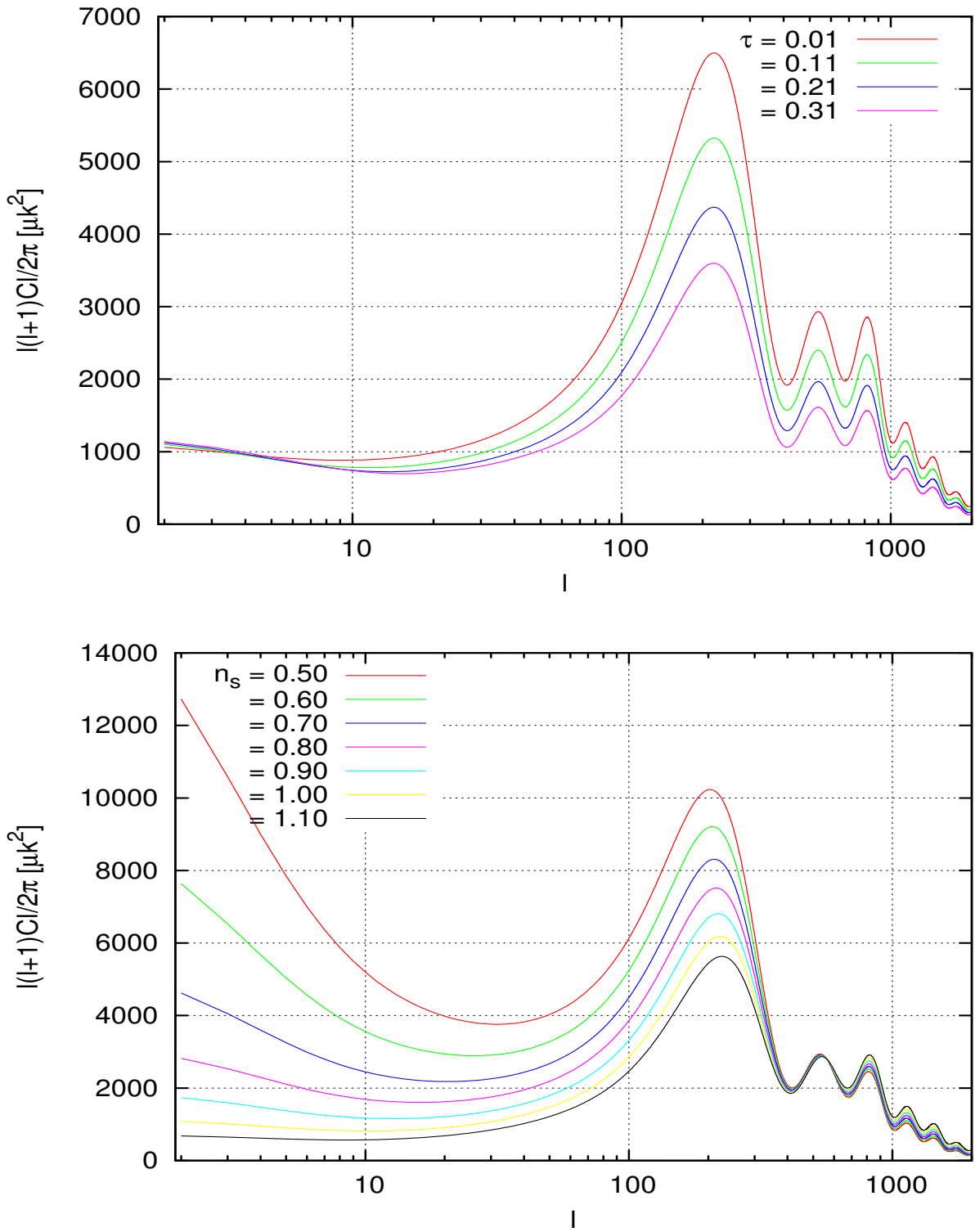


Figura 2.5. Variación de τ_{reio} y n_s

<code>l_max_scalar</code>	4000
<code>k_eta_max_scalar</code>	12000
<code>accurate_BB</code>	T
<code>accurate_polarization</code>	T
<code>accurate_reionization</code>	T
<code>do_late_rad_truncation</code>	F
<code>accuracy_boost</code>	12
<code>l_accuracy_boost</code>	4
<code>l_sample_boost</code>	3
<code>nthermo</code> en <code>modules.f90</code>	100000
Δx para $x > 25$ en <code>bessels.f90</code>	0.2

Tabla 2.2. Parámetros de referencia para CAMB.

cuando H_0 crece $\rightarrow d_a$ decrece $\rightarrow \theta_s$ crece $\rightarrow l_{pic}$ decrece. Mientras tanto los picos acústicos no son afectados en lo que refiere a la amplitud;

- crece A_s varía la amplitud de todo el espectro en forma proporcional. Este efecto es consecuencia que A_s es la amplitud del espectro primordial (2.21);
- crece Ω_b se observan variaciones en el tamaño relativo entre los picos impares-pares y la posición de los mismos. Esta mezcla de efectos están relacionadas con los procesos físicos que actúan en las distintas escalas y en los diferentes momentos y cómo ellos se relacionan con la densidad bariónica. Principalmente los efectos que lo producen son el valor de la velocidad del sonido en el medio ($c_s^2 \propto \omega_b^{-1}$) y la influencia gravitacional en los valles y en las barreras de potencial generados por la materia;
- crece Ω_{CDM} se observan comportamientos a la inversa de lo que ocurre en Ω_b . Esto se debe a que los efectos vistos para Ω_b pueden describirse en términos generales por la variación (crecimiento o decrecimiento) en la densidad total de materia Ω_M . Luego, al aumentar Ω_b , a Ω_M fijo, decrece Ω_{CDM} ;
- crece τ_{reio} a muy bajos l el espectro de potencias no se modifica, pero a mayores valores de l el espectro baja. La profundidad óptica τ nos da idea de la dispersión de los fotones generado por su interacción con los electrones liberados en la época en la cuál se da la reionización. Se cree que no debieron dispersarse gran cantidad de fotones del FCR. La profundidad óptica aparece como un factor a la cantidad de fotones medidos, por lo que su variación hará que varíe en forma proporcional el espectro;
- crece el índice espectral n_s crece la pendiente.

El espectro de temperatura del FCR es muy dependiente de los valores de los parámetros cosmológicos. Para poder contrastar los modelos teóricos utilizados por CAMB y CLASS con los datos observacionales de las diferentes misiones (WMAP, COBE, Planck, BICEP, BICEP2, BAO, etc) se utilizan códigos que recorren el espacio de los parámetros buscando el mejor ajuste de teoría con observación.

La autocorrelación en la temperatura, la correlación cruzada entre la temperatura y el modo E de la polarización y la autocorrelación en el modo E de la polarización para el modelo estándar (Λ CDM) calculada con los códigos CAMB y CLASS pueden observarse en las figuras 2.6, 2.7 y 2.8 respectivamente. Se obtuvieron para el conjunto de parámetros cosmológicos que se detallan en la Tabla 2.1. Se observa que el error relativo puede considerarse despreciable en comparación

al orden de magnitud involucrado. Se realizaron diferentes pruebas variando individualmente pares de parámetros cosmológicos de la Tabla mencionada.

Los tiempos de cómputo fueron de $45m43.880s$ y $26m15.785s$ para CAMB y CLASS, respectivamente. Notamos que el tiempo de cómputo de CLASS es significativamente menor.

CAMB vs. CLASS: autocorr. de la T

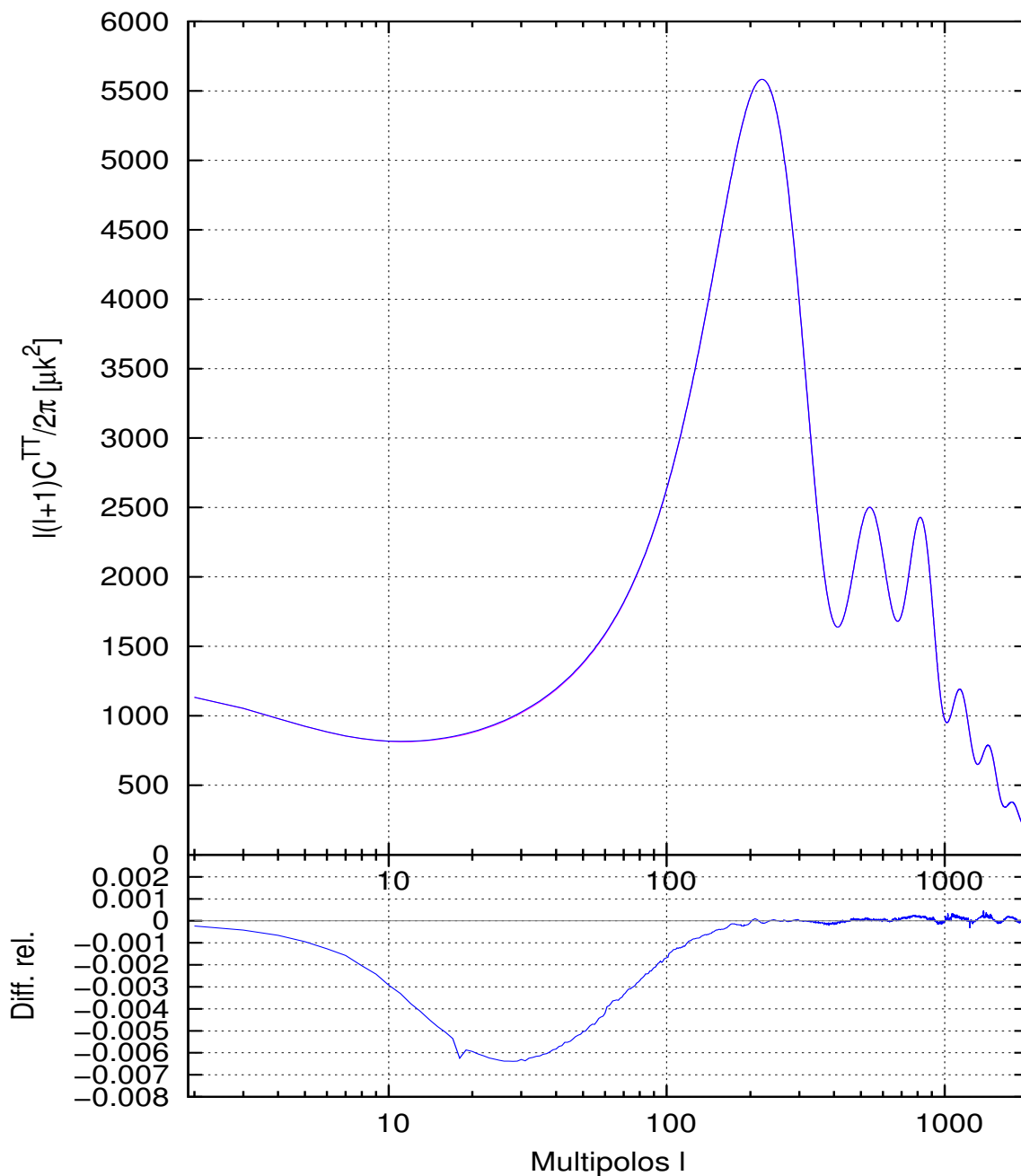


Figura 2.6. Panel superior: Autocorrelación en la T. Panel inferior: Error relativo entre los valores de CLASS y CAMB.

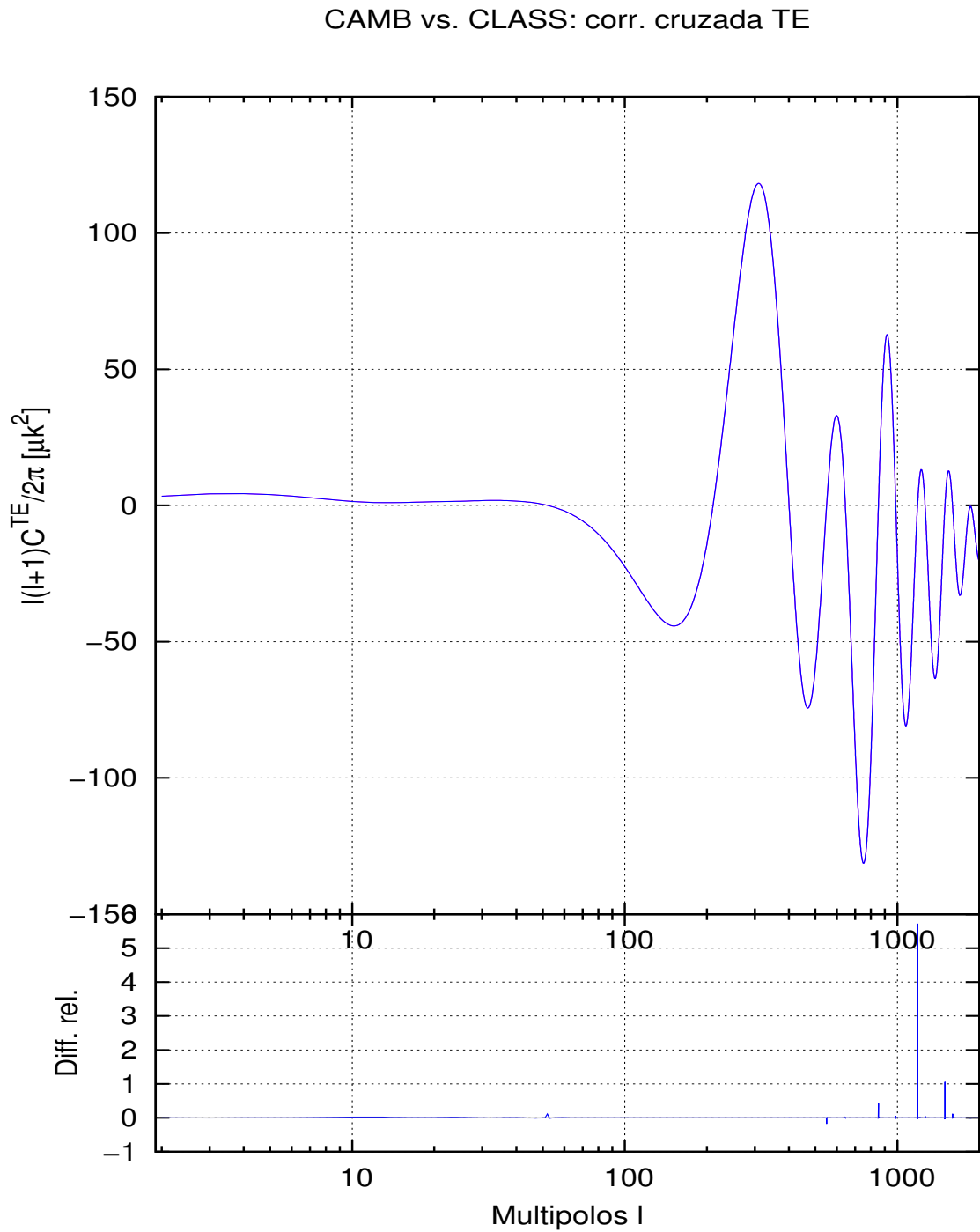


Figura 2.7. Panel superior: Correlación cruzada entre T y E. Panel inferior: Error relativo entre los valores de CLASS y CAMB.

CAMB vs. CLASS: autocorr. de la E

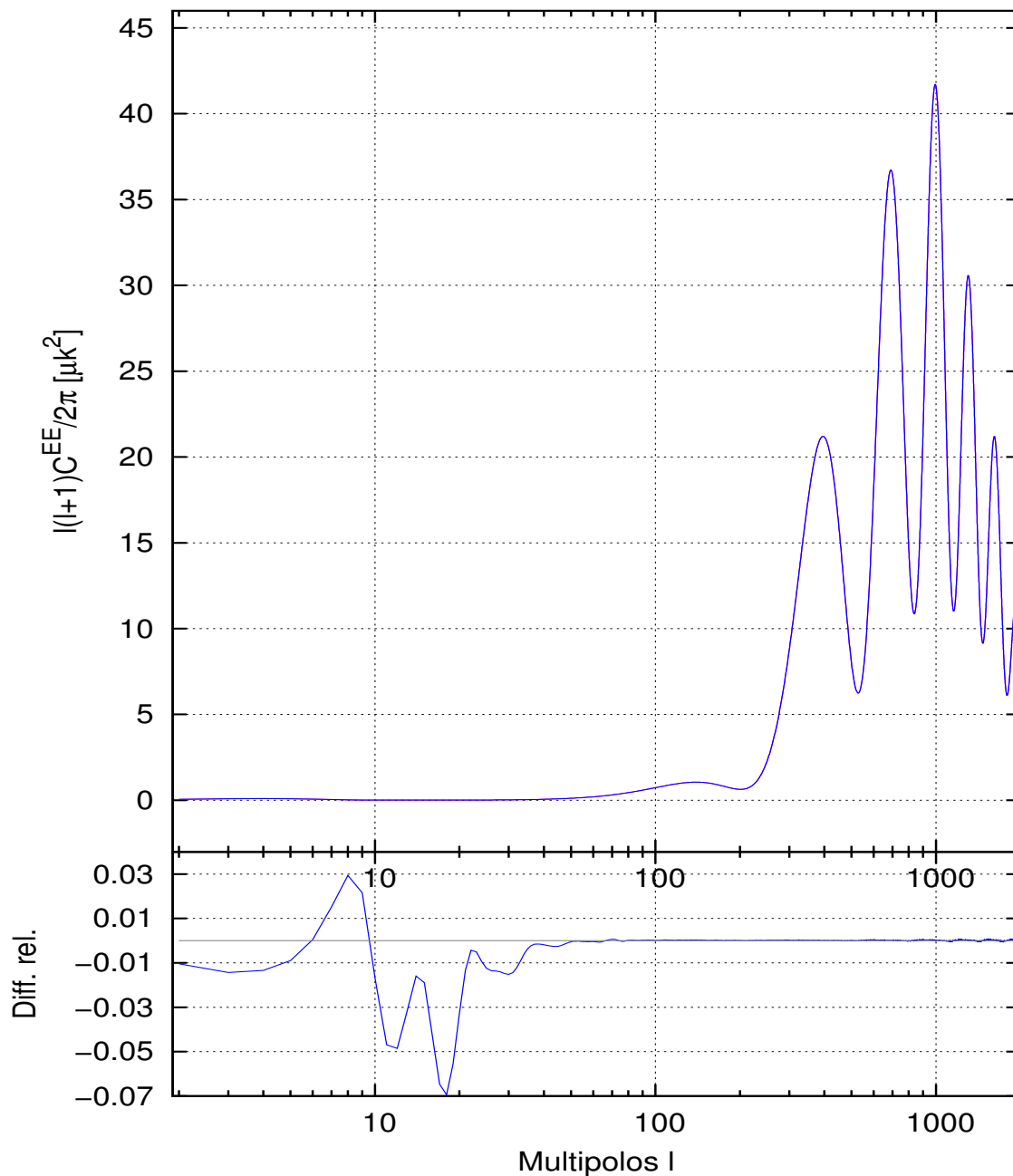


Figura 2.8. Panel superior: Autocorrelación en la E. Panel inferior: Error relativo entre los valores de CLASS y CAMB.

Capítulo 3

Análisis estadístico

La variación de los parámetros cosmológicos trae como consecuencia una variación en la forma y la ubicación de los picos acústicos del espectro de potencias. Podemos estimar el mejor ajuste de nuestros parámetros cosmológicos para un dado modelo teórico (calculado con CAMB y CLASS) a partir de un estudio estadístico.

La gran cantidad de información proveniente de las diferentes misiones satelitales y terrestres hace que la cantidad de datos para procesar sea muy grande. Es necesario tener herramientas estadísticas y computacionales para poder procesar esa información. Esto nos permite estimar mejor el conjunto de parámetros cosmológicos que mejor describen las observaciones y de esta forma intentar responder: ¿observamos más especies de partículas elementales? ¿observamos ondas gravitacionales? ¿qué curvatura posee el Universo? ¿qué tan viable es una teoría alternativa a la Relatividad General? y muchas otras.

3.1. Introducción a métodos bayesianos y cadenas de Markov

Para hacer ese análisis estadístico se utiliza el enfoque *bayesiano*. Utilizamos como base teórica para esta sección las explicaciones que se encuentran en [15]. Este enfoque considera el grado de creencia en una hipótesis, y se utiliza éste en Cosmología debido a que queremos saber qué probabilidad hay de que el modelo correcto sea el que representa nuestro Universo. La herramienta estadística que nos permite obtener ese tipo de predicciones es el teorema de Bayes. Consideramos dos eventos A y B y su intersección $A \cap B$. La probabilidad de que salga un evento B habiendo ocurrido el evento A es

$$P[B|A] \equiv \frac{P[A \cap B]}{P[A]}, \quad (3.1)$$

donde $P[B|A]$ es la probabilidad condicional de que salga B dado A. Cambiando A con B se obtiene la probabilidad de que dado B salga A. Luego despejando

$$P[A \cap B] = P[A] \cdot P[B|A] = P[B] \cdot P[A|B]. \quad (3.2)$$

Las dos últimas igualdades permiten despejar a $P[B|A]$ o $P[A|B]$. Ese despeje se conoce como teorema de Bayes y nos permite conocer el grado de verdad de una hipótesis dado un conjunto de datos. Si tenemos un conjunto de datos D y queremos verificar una hipótesis H, la probabilidad condicional de nuestra hipótesis es

$$P[H|D] = \frac{P[H]P[D|H]}{P[D]} \quad (3.3)$$

donde:

$P[H]$ es la probabilidad *a priori*;

$P[D|H]$ es la probabilidad de obtener D dada nuestra hipótesis, o función de verosimilitud o *likelihood*;

$P[D]$ es la probabilidad de observar nuestros datos, también llamada probabilidad marginal;

$P[H|D]$ es la probabilidad *a posteriori*.

Consideremos un dado modelo cosmológico determinado por un conjunto m de parámetros. Por otro lado, consideremos un conjunto de datos d_1, \dots, d_N y sus respectivas dispersiones $(\sigma_1, \dots, \sigma_N)$. Por ejemplo si d_i son los coeficientes a_{lm} de la expansión en armónicos esféricos de las fluctuaciones en temperatura, sus varianzas son los C_l 's.

Dado nuestro modelo, podemos estimar teóricamente los valores de los d_i , que llamaremos $d_i(m)$, con una incerteza $\sigma_i(m)$. Considerando que son variables aleatorias, su función de distribución estará representada por una Gaussiana. Por lo tanto, la probabilidad de medir d_i con una incerteza de σ_i para un dado modelo m es

$$P[\{d_i, \sigma_i\}|m] = \frac{1}{\sqrt{2\pi(\sigma_i(m)^2 + \sigma_i^2)}} \exp\left(-\frac{(d_i - d_i(m))^2}{2(\sigma_i(m)^2 + \sigma_i^2)}\right) [15]. \quad (3.4)$$

Ahora bien, dado un conjunto de datos (d_1, \dots, d_N) independientes, la probabilidad conjunta será el producto de las probabilidades individuales,

$$\begin{aligned} P[\{d_i, \sigma_i\}|m] \equiv \mathcal{L}(\{d_i, \sigma_i\}, m) &= \prod_{i=1}^N \left[\frac{\exp\left(-\frac{(d_i - d_i(m))^2}{2(\sigma_i(m)^2 + \sigma_i^2)}\right)}{\sqrt{2\pi(\sigma_i(m)^2 + \sigma_i^2)}} \right] \\ &= \frac{1}{\sqrt{\det C (2\pi)^N}} \exp\left(-\frac{d_i C_{ij}^{-1} d_j}{2}\right), \quad \text{usando } \langle d_i d_j \rangle = C_{ij} \end{aligned}$$

esta última expresión no es otra cosa que la función de verosimilitud o *likelihood* y C la matriz de covarianza total, suma de la matriz de covarianza del ruido y la de la señal. Nos dice la probabilidad de que obtengamos los datos $\{d_i, \sigma_i\}$ a partir de un modelo m . La función de verosimilitud posee información referente al experimento que estamos utilizando. Esto es, cuando utilizamos observaciones de Planck, BAO, SN, WMAP, etc, se tienen diferentes funciones. Para obtener la imagen del FCR es necesario quitar la contribución de las diferentes estructuras que se encuentran entre la superficie de última dispersión y el observador. Para ello se utilizan modelos de efectos interestelares e intergalácticos. Los mismos dependen de parámetros libres denominados *nuisance*.

Ahora bien, nosotros estamos interesados en los valores de los parámetros teóricos. Por lo tanto, no queremos obtener la *likelihood* $P[\{d_i, \sigma_i\}|m]$. En lugar de ello, se desea obtener la $P[m|\{d_i, \sigma_i\}]$. La calculamos utilizando el teorema de Bayes

$$P[m|\{d_i, \sigma_i\}] = P[\{d_i, \sigma_i\}|m] \frac{P[m]}{P[\{d_i, \sigma_i\}]}. \quad (3.5)$$

Aquí $P[m]$ se conoce como *prior* y el denominador es la probabilidad marginal. El lado izquierdo se conoce como la distribución *a posteriori*. El *prior* depende de nuestro conocimiento previo del modelo. Para obtener el mejor ajuste para nuestro modelo m se maximiza la función *likelihood* respecto de los parámetros del modelo.

Una vez conocida la distribución *a posteriori* puede calcularse el valor medio de una función f (que depende de los parámetros del modelo) como

$$E[f(m)|\{d_i, \sigma_i\}] = \frac{\int dm f(m) P[m] P[\{d_i, \sigma_i\}|m]}{\int dm P[m] P[\{d_i, \sigma_i\}|m]}. \quad (3.6)$$

Cadenas de Markov Monte Carlo También llamadas MCMC por sus siglas en inglés, *Monte Carlo Markov Chains*. Consiste en utilizar cadenas de Markov¹ para realizar integraciones Monte Carlo². El objetivo de estos métodos es muestrear el espacio de parámetros de la manera más eficiente y obtener la distribución *a posteriori* $P[m|\{d_i, \sigma_i\}]$.

Los métodos Monte Carlo se pueden utilizar para estimar el valor medio de una función arbitraria $f(x)$. Para hacerlo, se toma una muestra aleatoria de n números x_i con distribución $P[m|\{d_i, \sigma_i\}]$ (*a posteriori*) y se calcula el valor medio haciendo

$$\bar{f}(x) = \frac{1}{n} \sum_{i=1}^n f(x_i). \quad (3.7)$$

Ahora bien, para obtener $P[m|\{d_i, \sigma_i\}]$ es necesario recurrir a cadenas de Markov. La idea general es considerar a $P[m|\{d_i, \sigma_i\}]$ como distribución estacionaria³. Para hacer eso se toma una secuencia aleatoria de puntos $\{x_1, x_2, \dots\}$ tal que el número x_{i+1} depende solamente del estado actual de la cadena x_i y lo calculo haciendo $P(x_{i+1}|x_i)$. De esta forma la cadena irá perdiendo información de la condición inicial x_1 .

Luego de un tiempo, la distribución de puntos comenzará a muestrearse con la forma de la distribución estacionaria. Para hacer un estimador del valor medio, no se toman en cuenta los puntos iniciales donde se puede considerar que no se muestrean con la distribución estacionaria. Digamos que lo hace luego de m iteraciones, entonces el estimador de $f(x)$ será

$$\bar{f}(x) = \frac{1}{n-m} \sum_{i=m+1}^n f(x_i). \quad (3.8)$$

Para asegurarnos que la distribución estacionaria coincide con la distribución *a posteriori* hay diferentes algoritmos. En particular describiremos el algoritmo más utilizado en los códigos que utilizaremos, el algoritmo de Metropolis-Hastings.

Algoritmo de Metropolis-Hastings Es un método que se utiliza para obtener una distribución de probabilidades desconocida a partir de una conocida. Supone que una distribución arbitraria, digamos $q(\cdot, m_t)$, es proporcional a la que deseamos obtener. La función de distribución q nos permite tomar muestras en el espacio de los parámetros. Seguimos los lineamientos de [18].

Para un paso t , la elección del siguiente estado X_{t+1} se realiza muestreando un valor Y a partir de una función de distribución conocida, q . Por ejemplo, si la distribución es Gaussiana, la misma será función del estado actual de la muestra, X_t , y de su matriz de covarianza X (fija). Para el punto candidato Y se calcula una probabilidad $\alpha(X_t, Y)$ tal que

$$\alpha(X, Y) = \min\left(1, \frac{P(Y)q(X|Y)}{P(X)q(Y|X)}\right) \quad (3.9)$$

Por otro lado, se elige un número aleatorio entre 0 y 1. Si este número es menor que α , se toma $X_{t+1} = Y$, en otro caso $X_{t+1} = X_t$. Se puede demostrar que cualquiera sea la distribución propuesta q , la distribución estacionaria de la cadena será P .

¹Son procesos estocásticos en donde el siguiente punto de la cadena depende sólo del punto anterior, o sea, tiene pérdida de memoria.

²Métodos de integración que utilizan número pseudo-aleatorios.

³Esto significa que es la distribución a la cuál vamos a hacer tender nuestra muestra de puntos.

Convergencia Gelman y Rubin desarrollaron un estadístico capaz de establecer la convergencia entre muestreos independientes. Para ello es preciso realizar múltiples cadenas independientes unas de otras que recorran el espacio de los parámetros. Si tenemos M cadenas MCMC, ya desechados los primeros puntos (que introducen mucho ruido), cada parámetro del modelo tendrá un valor medio y una varianza ya establecida. A partir de las varianzas de estas cadenas se establece el criterio: cuando la varianza entre cadenas diferentes es menor que la varianza individual de una cadena se dice que la cadena de Markov ha convergido.

Sean θ_{ij} los n distintos valores que toma un parámetro de la cadena j y sea $\bar{\theta}$ la media del mismo. Su varianza s_j de la cadena j estará dada por

$$s_j^2 = \frac{1}{n-1} \sum_{i=1}^n (\theta_{ij} - \bar{\theta}_j)^2, \quad (3.10)$$

y la media de las varianzas de las m cadenas se define como

$$W = \frac{1}{m} \sum_{j=1}^m s_j^2. \quad (3.11)$$

Sin embargo, W subestima los valores verdaderos de la varianza de la distribución estacionaria ya que las cadenas probablemente no hayan recorrido totalmente el espacio de los parámetros, por ser finita. Luego, entre las cadenas se define la varianza entre las cadenas como

$$B = \frac{n}{m-1} \sum_{j=1}^m \left(\bar{\theta}_j - \bar{\bar{\theta}} \right)^2, \quad \text{donde } \bar{\bar{\theta}} = \frac{1}{m} \sum_{j=1}^m \bar{\theta}_j. \quad (3.12)$$

Luego, el estimador de la varianza de la distribución estacionaria (recordemos que es la que queremos reproducir a partir de nuestra muestra) tiene un peso promediado por W y B como

$$\hat{V}ar(\theta) = \left(1 - \frac{1}{n}\right)W + \frac{1}{n}B, \quad (3.13)$$

de esta forma se sobreestima el valor verdadero de la varianza si los puntos fueron apropiadamente dispersados y no posee sesgo si la distribución inicial es la distribución estacionaria. En el límite cuando la población tiende a infinito, o sea $n \rightarrow \infty$, el estimador de la varianza $\hat{V}ar(\theta)$ tiende a la varianza de la distribución estacionaria $Var(\theta)$.

Se define el estadístico R

$$R = \sqrt{\frac{\hat{V}ar(\theta)}{W}} \quad (3.14)$$

Este valor, para cadenas convergidas, se encuentra entre 1.1 o 1.2. Generalmente, en la literatura y en los códigos que usaremos se utiliza $R - 1$.

Un conjunto de parámetros cosmológicos nos dá una realización del universo. Pero al modificarlas independientemente unas de otras, se modifica la realización del universo. Cada conjunto de parámetros produce una predicción para las distintas cantidades estadísticas. ¿Cómo saber qué conjunto de parámetros es el más apropiado para describir lo que se observa? Para realizar el muestreo según lo visto en esta introducción se utilizan los dos códigos que veremos en las siguientes secciones.

3.2. Descripción de MontePython

MontePython es un código de estimación de parámetros cosmológicos en lenguaje Python. Está programado para trabajar acopladamente con CLASS (a través de la librería `cython`). Se encarga de recorrer el espacio de los parámetros para un dado modelo cosmológico.

Está programado en forma modular, esto es, podemos encontrar las diferentes tareas que ejecuta el código en *islas* autoconsistentes. Posee una vasta librería con las que trabaja desde la interacción e interpretación de datos iniciales y variables, pasando por la consultas a CLASS, aplicando algoritmos estadísticos y llegando a la manipulación de los datos y procesamiento de los mismos.

Utiliza las librerías Cython, para comunicar las partes de las corridas programadas en C con las de Python. Classy, se encarga de la interacción con CLASS para extraer los parámetros cosmológicos de CLASS y utilizarlos en MontePython.

MontePython, junto con CosmoMC que veremos en la próxima sección, se encarga de realizar todo el análisis estadístico visto al principio de esta sección.

Para realizar las muestras de cadenas fue utilizado el cúmulo de supercomputadoras Miztli ⁴.

3.2.1. Lógica del programa

Se le dá como argumento un archivo inicial en el cuál se especifica:

- los experimentos a utilizar. Puede ser WMAP, BAO, SN, Planck, cada uno con sus opciones, por citar los más utilizados;
- los parámetros cosmológicos del modelo y los que queremos derivar a partir del modelo. Se le pasa el valor medio, un valor mínimo, uno máximo (entre los cuales se hará el muestreo), un valor para la precisión, la escala del mismo y el rol del parámetro dentro de la cadena.
- la cantidad de pasos de las cadenas.

Todos los parámetros que posee están determinados por defecto y se utilizan para complementar la información que se le pasa en el archivo. Dentro del rol se pueden especificar tres tipos de parámetros:

- *nuisance*;
- *cosmo* son parámetros que utiliza el código Boltzmann;
- *derived* no son variados por las MCMC pero se mantienen en la memoria de las cadenas y se derivan a partir de otros parámetros.

Todos los elementos que tengan la etiqueta `cosmo` o `derived` son interpretados por el código cosmológico (CLASS en nuestro caso), y no están en ningún lugar del MontePython.

El esquema de trabajo de código puede resumirse como muestra la Figura 3.1. El archivo `MontePython.py` inicializa los tres bloques: Data (a través de `data.py`), Cosmo (a través de `classy.py`) y Likelihood (a través de `likelihood_class.py`), y le da inicio al *muestreador genérico*. Éste fija un primer conjunto de valores de los parámetros que se lo pasa a Data. Este módulo completa los valores de los datos y se los pasa a Cosmo (CLASS) que es el encargado de hacer el cálculo de los C_l 's y los P_k . Estos valores son comunicados al módulo Likelihood que finalmente hace los cálculos estadísticos. Una vez realizado esto, Likelihood le pasa al *muestreador genérico* los valores de la Likelihood y éste varía los parámetros para dar un nuevo paso de la cadena.

⁴<http://www.super.unam.mx/>

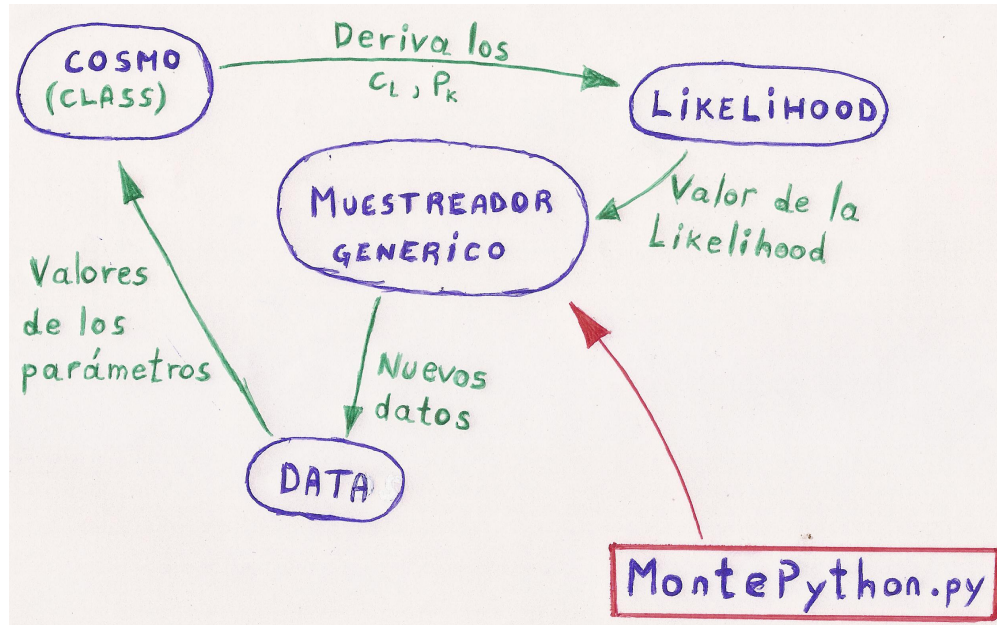


Figura 3.1. Flujo del código MontePython. MontePython.py inicializa la corrida exigiéndole al muestreador que tome una muestra de valores de parámetros. Esta muestra le es pasada a Data. Se completan todos los parámetros y ese conjunto de parámetros se le pasa a CLASS para que calcule los C_l y P_k . Una vez calculados se realiza el análisis estadístico y vuelve a empezar el ciclo. Basado en esquema de Julien Lesgourgues https://lesgourg.web.cern.ch/lesgourg/class-tour/MP_lecture3.pdf.

Los módulos principales de MontePython son MontePython, run, parser_mp, initialize, data, sampler y likelihoods_class. Por otro lado tenemos el módulo analyze que es el encargado de calcular la convergencia de las cadenas, de calcular los histogramas, calcular la matriz de covarianza, y realizar los gráficos.

El código genera una carpeta de salida con la cantidad de cadenas solicitadas junto con un archivo, log.param. Este archivo posee toda la información de la corrida y, para no generar conflictos, si en una corrida posterior por error se selecciona el mismo directorio de salida donde están esas cadenas, este archivo es el que se pasa como argumento, evitando así la pérdida de información.

El análisis de las cadenas y los gráficos son realizados con el comando info. El mismo calcula los estimadores de convergencia, los valores del mejor ajuste, valores medios para cada parámetro, los intervalos de mínima credibilidad (a 68 %, 95.4 % y 99.7 %), el valor de las likelihoods, las matrices de covarianza y los gráficos: marginalizados (1D, 2D) junto con un gráfico triangular donde se muestra la correlación entre todos los parámetros que el usuario desee.

3.3. Descripción de CosmoMC

CosmoMC (*Cosmological Monte Carlo*) es un código en lenguaje Fortran 2008 que calcula las Cadenas de Markov con métodos de Monte Carlo (MCMC por sus siglas en inglés *Markov Chain Monte Carlo*) que exploran el espacio de los parámetros cosmológicos. El mismo posee una gran variedad de librerías en Python que permiten construir gráficos o salidas de tablas en Latex.

CosmoMC utiliza como código a CAMB para realizar los cálculos de los espectros de potencia.

Para realizar las cadenas es necesario poseer varios procesadores ya que la cantidad de cálculos necesarios para recorrer el espacio de los parámetros, calcular las *likelihoods*, y otros estadísticos es mucho. El algoritmo que utiliza para optimizar el tiempo de cómputo es el de Metropolis. El

mismo funciona muy bien aún con muchos parámetros *nuisance*, como es el caso de Planck.

Se puede compilar con la opción de MPI (para correr a través de diferentes nodos en un cúmulo de computadoras), permite finalizar el cálculo una vez alcanzado el criterio de convergencia que se haya elegido.

La salida de la corrida nos da tres conjuntos de archivos:

- `.data` son los archivos binarios con los C_l 's calculados
- `.log` posee información de las cadenas que está calculando (por ejemplo el estadístico $R - 1$)
- `.txt` posee las cadenas calculadas.

Por otro lado, el programa `getdist` es una herramienta muy poderosa que puede implementarse por fuera de los límites de `CosmoMC`, aunque de ello no nos ocuparemos. Es una herramienta para realizar el pos-procesamiento de las cadenas y poder calcular toda la estadística y realizar los gráficos de salida. Tiene librerías Python y una vasta cantidad de manuales y ayudas para poder construir los propios *scripts* para realizar los gráficos. Pueden realizarse gráficos 1D, 2D y 3D.

Para utilizar `CosmoMC` es necesario pasarle como argumento un archivo. El mismo contiene los nombres de las funciones *likelihoods* utilizadas, las matrices de covarianza utilizadas (puede actualizarla a medida que calcula los nuevos parámetros), y el criterio de convergencia, entre otros datos técnicos.

MPI realiza comunicaciones entre las cadenas varias veces durante una larga corrida. El motivo es para informar a las cadenas el espacio de parámetros que están recorriendo las otras cadenas.

3.4. Comparación de resultados y rendimiento

Se utilizaron las funciones *likelihood* y datos observacionales de la última misión Planck, Planck 2015. Los mismos fueron instalados y utilizados en el súper cúmulo de computadoras Mitzli.

Los resultados para uno y otro código han sido comparados con el artículo de resultados de los parámetros cosmológicos de la colaboración Planck 2015 [3]. Las corridas que consideramos fueron las realizadas para calcular el espectro de potencias de temperatura y polarización para un modelo cosmológico (Λ CDM). Los resultados que exponen son compatibles con las misiones anteriores[3] (Planck 2013[19] y WMAP[20]) pero con mayor precisión. Sus resultados fueron obtenidos a partir de modelos corridos con las versiones `camb` y `CosmoMC`, ambos de Enero 2015 con algunas modificaciones.

Nos restringiremos al cálculo de C_l 's de la temperatura y polarización para valores altos y bajos de l introduciendo el efecto de lente gravitacional. Los parámetros cosmológicos que hemos variado son 6, los correspondientes al modelo de 6 parámetros Λ CDM, es decir: ω_b , ω_{CDM} , θ_s , A_s , n_s y τ_{reio} . Los valores obtenidos por la colaboración Planck 2015[3] considerando polarización y lente gravitacional ($TT + lowP + lensing$), son los que se muestran en la tabla 3.1. En la misma se puede observar que los valores de los parámetros cosmológicos en nuestras corridas son compatibles con los publicados en la literatura.

Podemos observar a partir de los valores de $R - 1$ para cada parámetro que las cadenas han convergido. Por otro lado podemos observar la evolución de los parámetros durante las corridas de las ocho cadenas en cada código en las figuras: 3.2, 3.3, 3.4, 3.5, 3.6, 3.7. En las mismas observamos que las cadenas recorrieron el espacio de los parámetros entre valores acotados e iguales.

En las Figuras 3.10 a 3.24 podemos observar la diferencia en el procesamiento de los códigos. A izquierda se encuentran los correspondientes a `MontePython` y a derecha a `CosmoMC`. Comparándolos con los análogos que se encuentran en la literatura vemos una diferencia en la suavidad de las curvas, como consecuencia del suavizado que elegimos utilizar y la densidad de puntos que hay en el interior.

3. Análisis estadístico

Parámetro	Planck 2015	CosmoMC	MontePython
$\Omega_b h^2$	0.02226 ± 0.00023	0.02224 ± 0.00023	0.02226 ± 0.00024
$\Omega_{CDM} h^2$	0.1186 ± 0.0020	0.1186 ± 0.0020	0.1187 ± 0.0020
$100\theta_s$	1.04103 ± 0.00046	1.04101 ± 0.00046	1.04200 ± 0.00045
τ_{reio}	0.066 ± 0.016	0.066 ± 0.016	0.070 ± 0.016
$\ln 10^{10} A_s$	3.062 ± 0.029	3.064 ± 0.029	3.071 ± 0.028
n_s	0.9677 ± 0.0060	0.9674 ± 0.0059	0.9669 ± 0.0061

Tabla 3.1. Estos valores corresponden a la misión Planck 2015 considerando $TT+lowP+lensing$ con 68% de confianza, la segunda columna representa los valores de nuestras corridas en CosmoMC y la tercera correspondiente a MontePython.

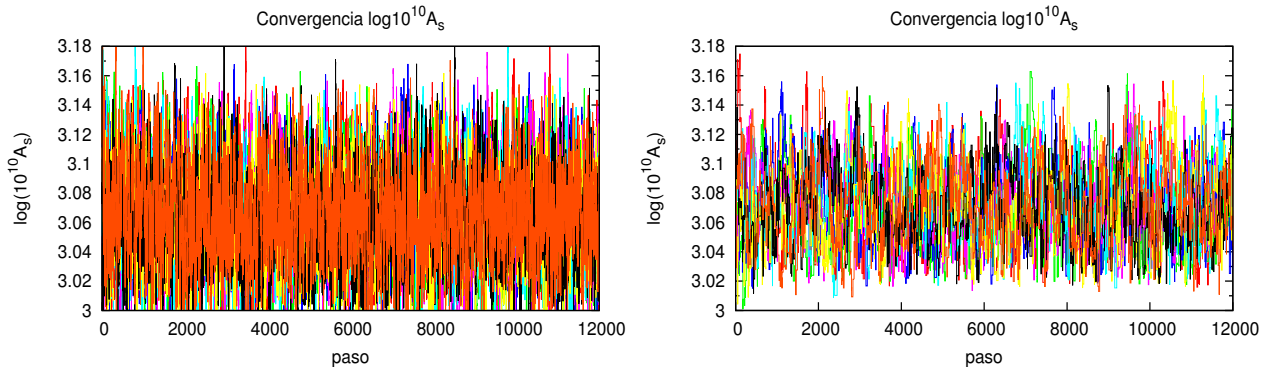


Figura 3.2. Valores para A_s , a izq. CosmoMC y a der. MontePython. Cada color representa una cadena diferente. La superposición de las curvas indica que las cadenas recorrieron el mismo espacio de los parámetros.

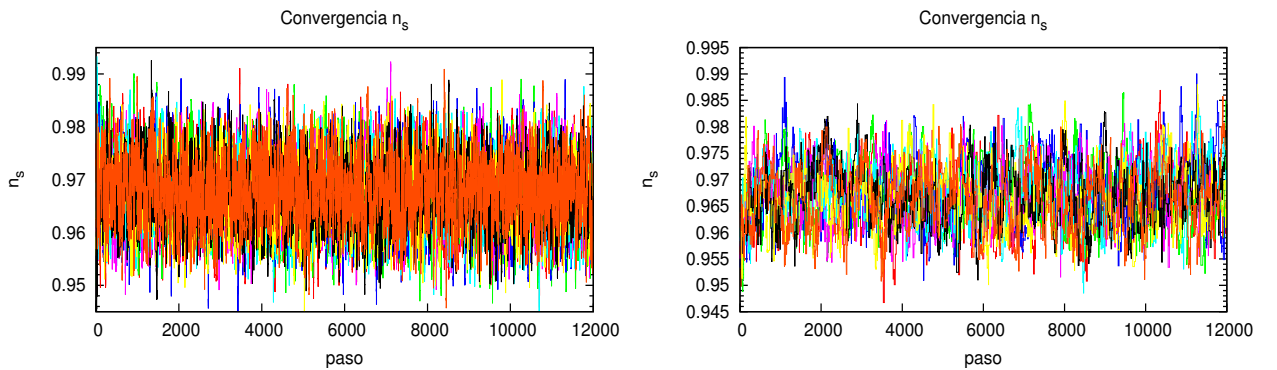


Figura 3.3. Idem para n_s

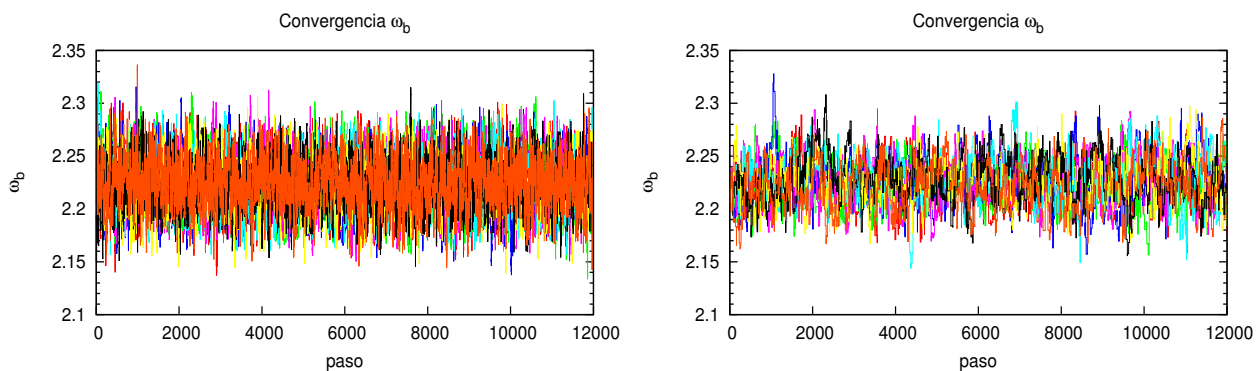


Figura 3.4. Idem para ω_b

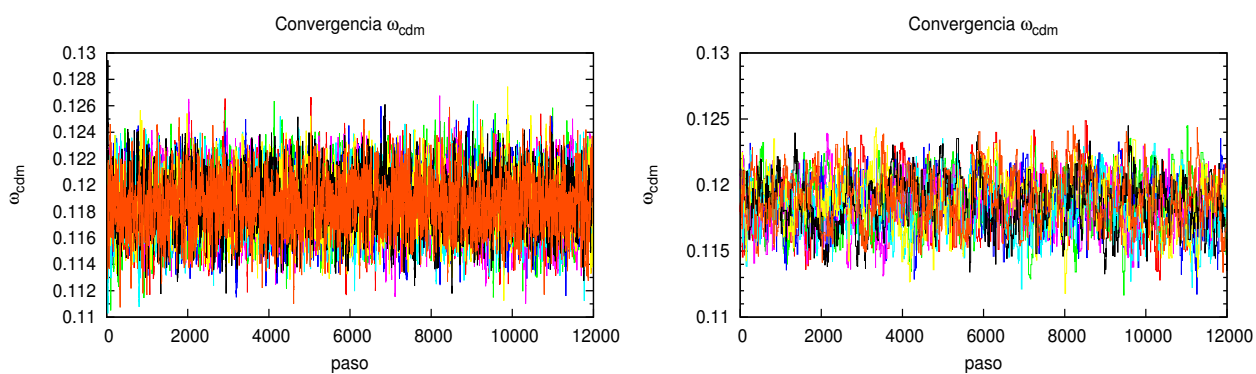


Figura 3.5. Idem para ω_{cdm}

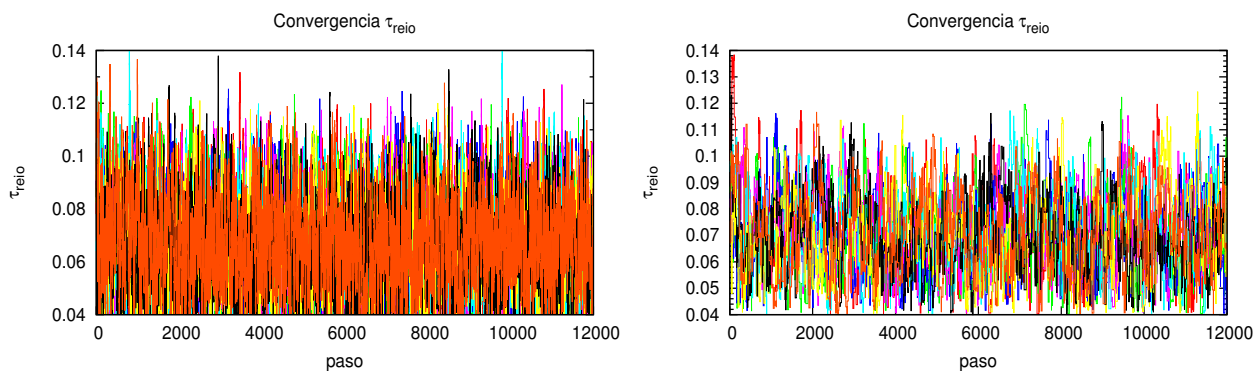


Figura 3.6. Idem para τ_{reio}

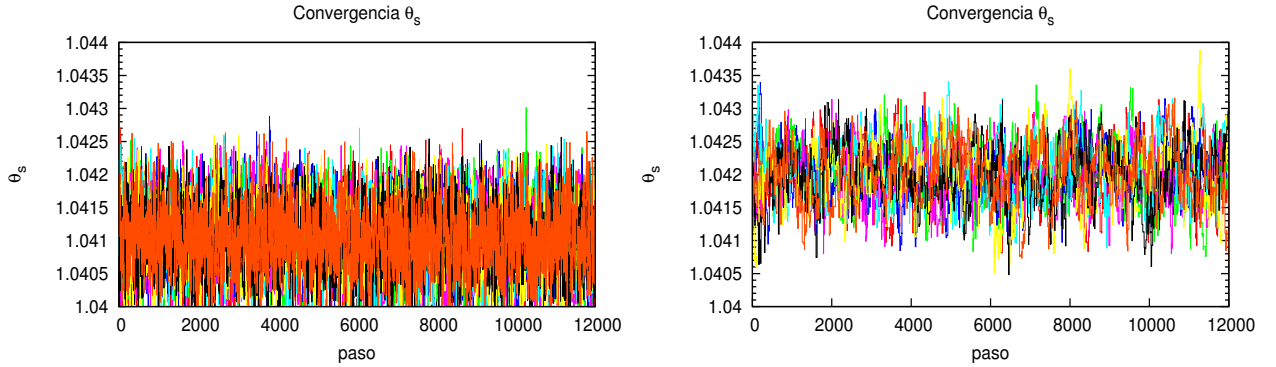


Figura 3.7. Idem para θ_s

Parámetro	R-1 - MontePython	R-1 - CosmoMC
$100\omega_b$	$1.278e-02$	$8.0e-04$
ω_{CDM}	$1.627e-02$	$1.5e-03$
$100\theta_s$	$6.97e-03$	$6.0e-04$
τ_{reio}	$1.32e-03$	$1.37e-03$
$\ln 10^{10} A_s$	$1.12e-02$	$1.21e-03$
n_s	$1.64e-03$	$1.21e-03$

Tabla 3.2. Valores del criterio de convergencia para MontePython y CosmoMC

Hemos utilizado un $N = 32050$. La elección del mismo se realizó luego de haber corrido el código CosmoMC habiendo considerado como criterio de convergencia $R - 1 < 0.01$. Eso dió como resultado un número de pasos, en promedio, de 32050 que fue el utilizado en MontePython. El resultado que dió este último, para cada parámetro se muestra en la tabla 3.2. De los valores de $R - 1$ obtenidos podemos concluir que las cadenas de MontePython han convergido.

Se puede observar en la figura 3.8 la correlación entre los distintos parámetros del modelo Λ CDM. Estas correlaciones coinciden con la correspondiente en la literatura (ver [3] página 17).

El tiempo de cómputo podemos verlo en la tabla 3.3, donde se aprecia que es considerablemente menor para MontePython.

tiempo	CosmoMC	Monte Python
real	43h 58m	13h 19m
CPU	548h 48m (~ 22 d)	111h 58m (~ 4.5 días)

Tabla 3.3. Tiempos de cómputo de las MCMC de CosmoMC y MontePython en el caso de estudio de esta Tesis.

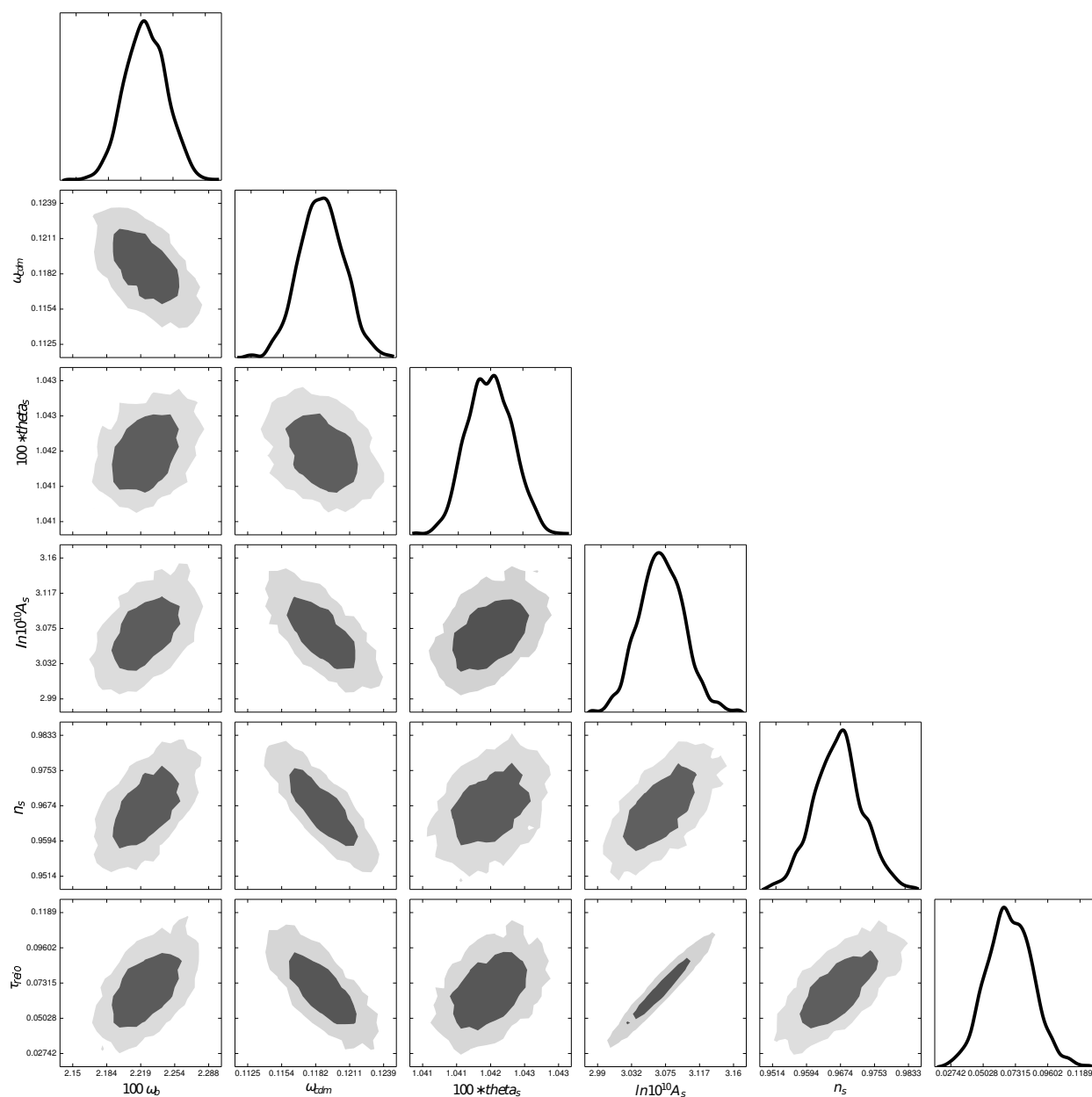


Figura 3.8. Correlación entre todos los parámetros de Λ CDM con Monte Python. Se muestran contornos de confianza de 68% y 95%.

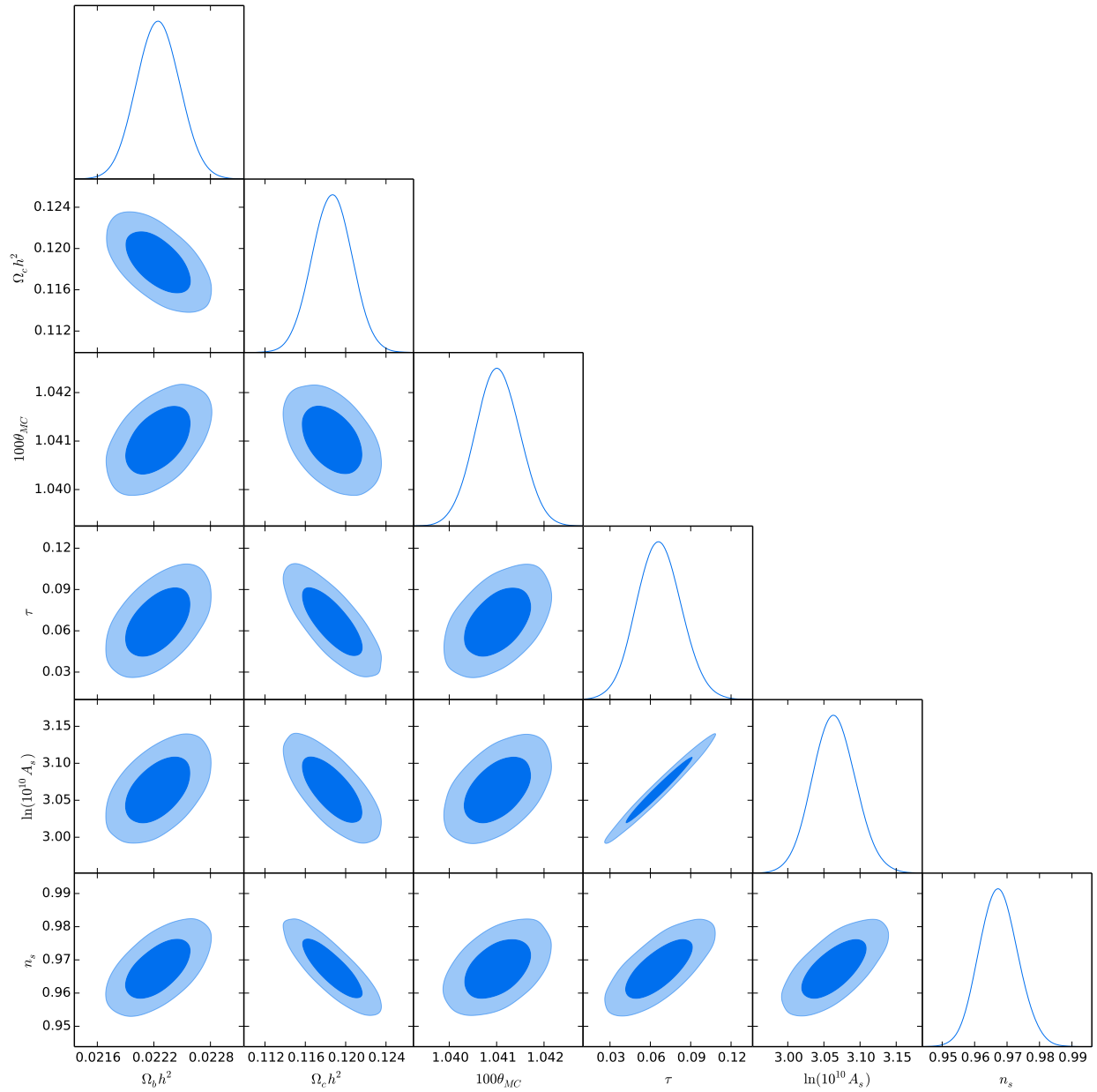


Figura 3.9. Correlación entre todos los parámetros de Λ CDM con CosmoMC. Se muestran contornos de confianza de 68 % y 95 %.

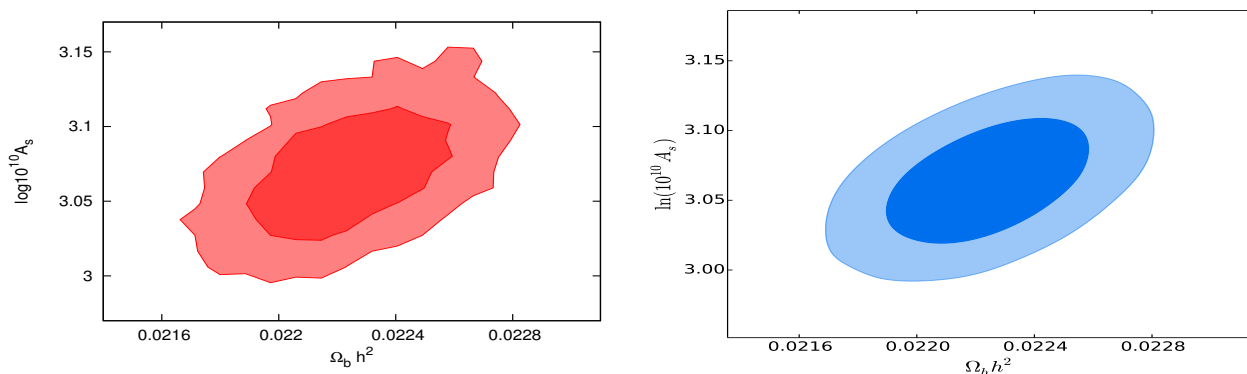


Figura 3.10. Relación entre ω_b y $\log 10^{10} A_s$. A izq. resultados de MontePython y a der. resultados de CosmoMC

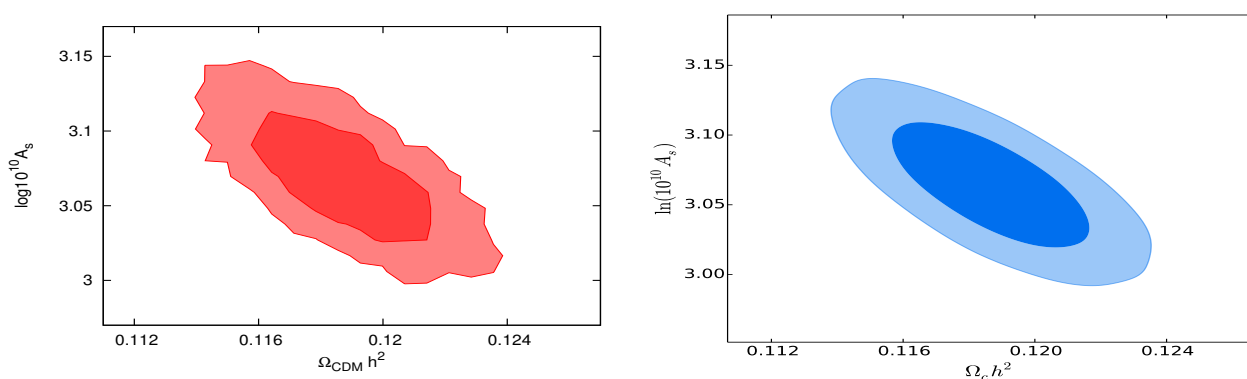


Figura 3.11. Relación entre ω_{cdm} y $\log 10^{10} A_s$. A izq. resultados de MontePython y a der. resultados de CosmoMC

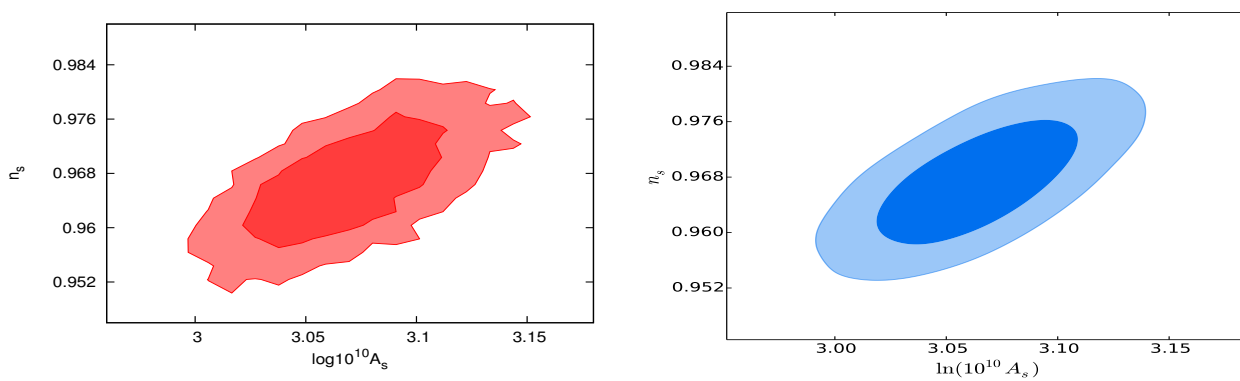


Figura 3.12. Relación entre n_s y $\log 10^{10} A_s$. A izq. resultados de MontePython y a der. resultados de CosmoMC

3. Análisis estadístico

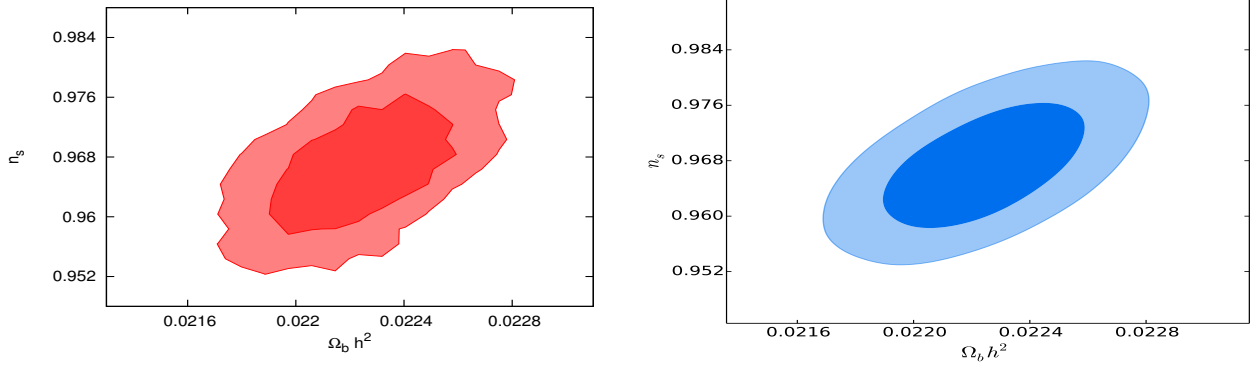


Figura 3.13. Relación entre ω_b y n_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

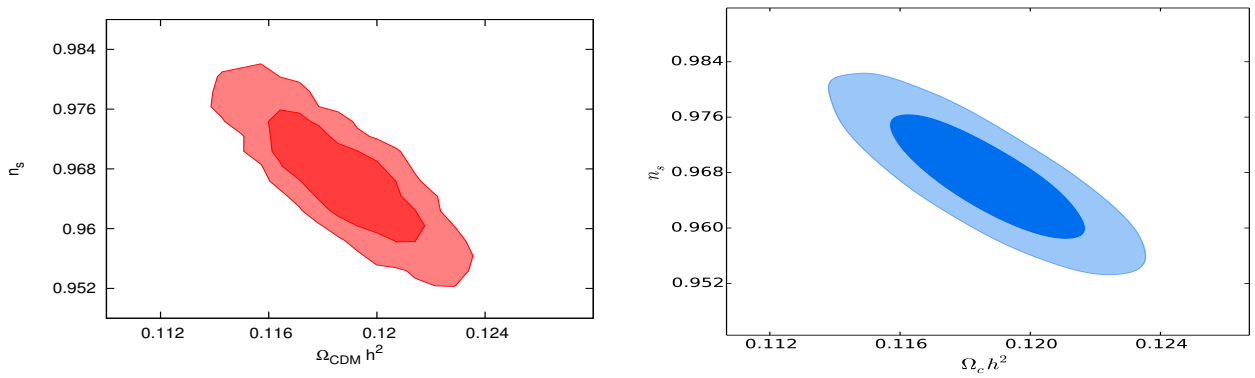


Figura 3.14. Relación entre ω_{cdm} y n_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

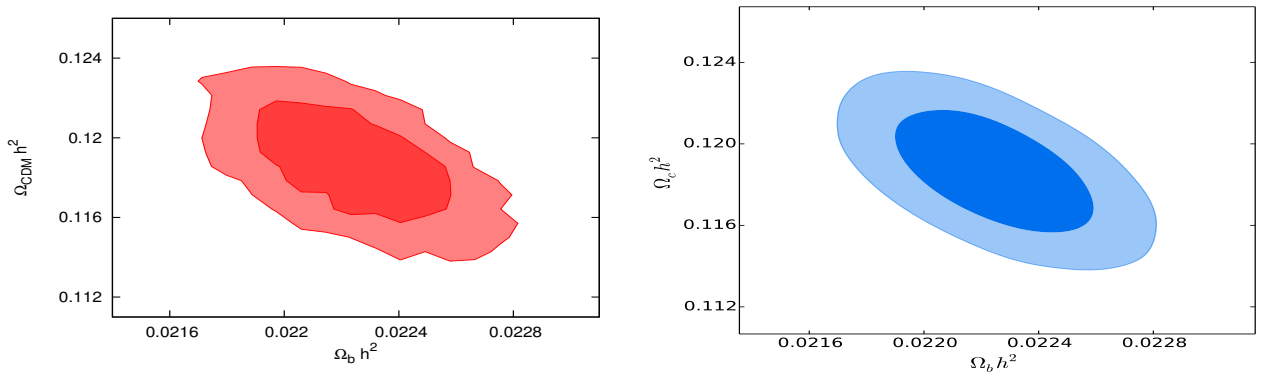


Figura 3.15. Relación entre ω_b y ω_{cdm} . A izq. resultados de MontePython y a der. resultados de CosmoMC

3.4. Comparación de resultados y rendimiento

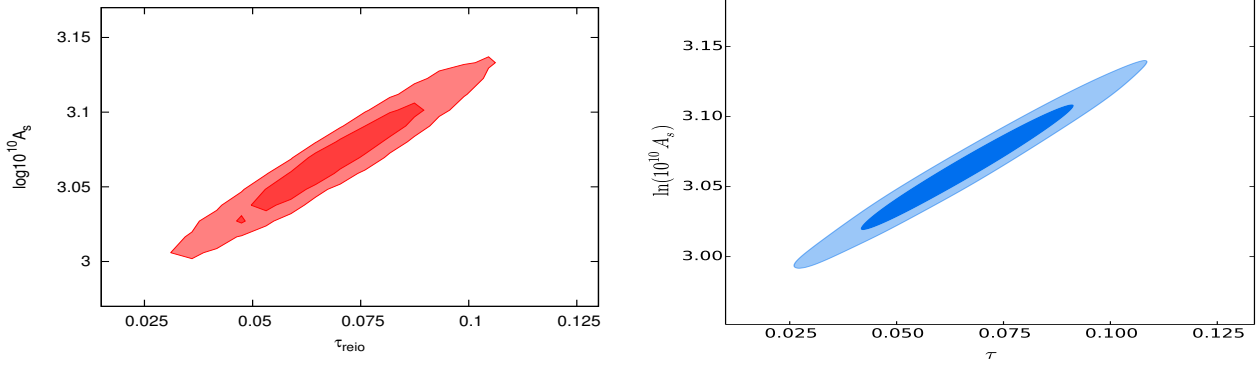


Figura 3.16. Relación entre τ_{reio} y $\log 10^{10} A_s$. A izq. resultados de MontePython y a der. resultados de CosmoMC

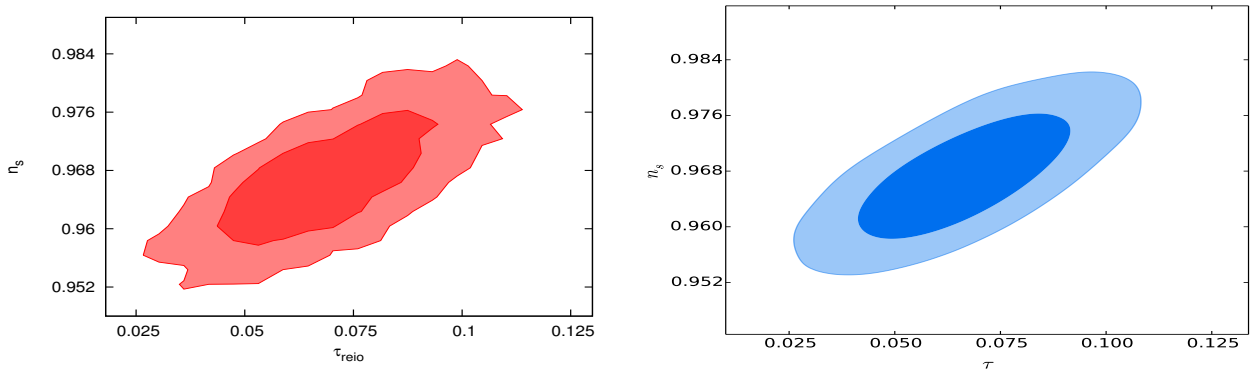


Figura 3.17. Relación entre τ_{reio} y n_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

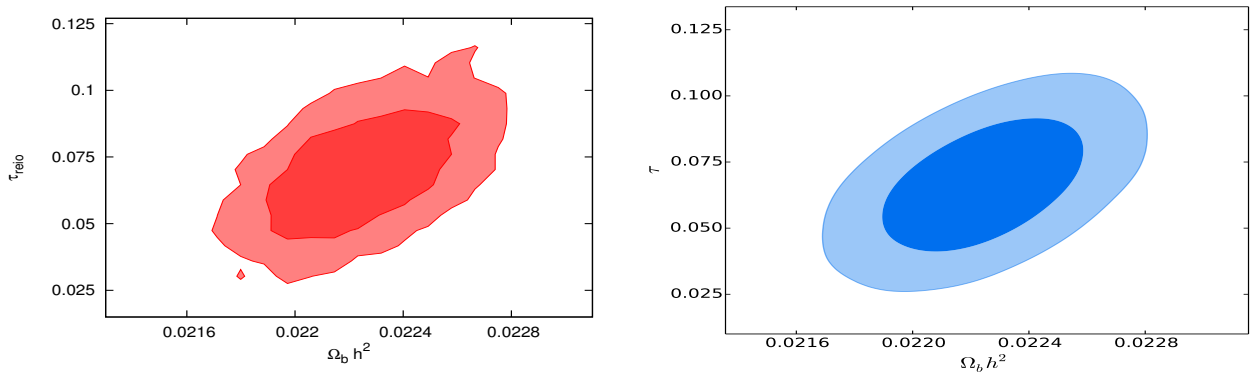


Figura 3.18. Relación entre ω_b y τ_{reio} . A izq. resultados de MontePython y a der. resultados de CosmoMC

3. Análisis estadístico

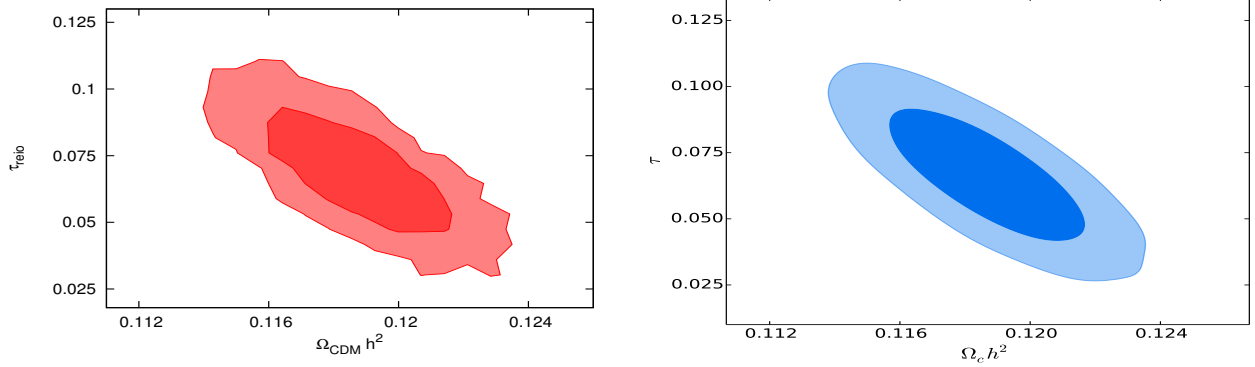


Figura 3.19. Relación entre ω_{cdm} y τ_{reio} . A izq. resultados de MontePython y a der. resultados de CosmoMC

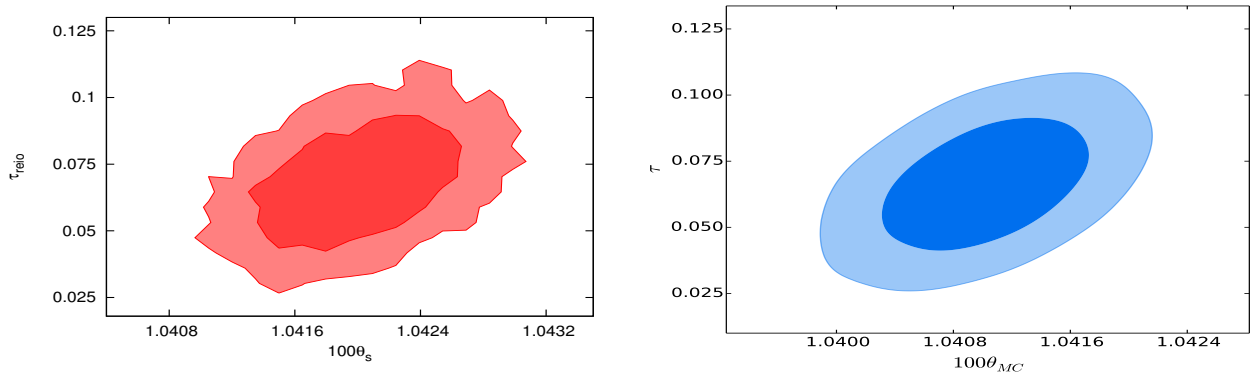


Figura 3.20. Relación entre τ_{reio} y θ_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

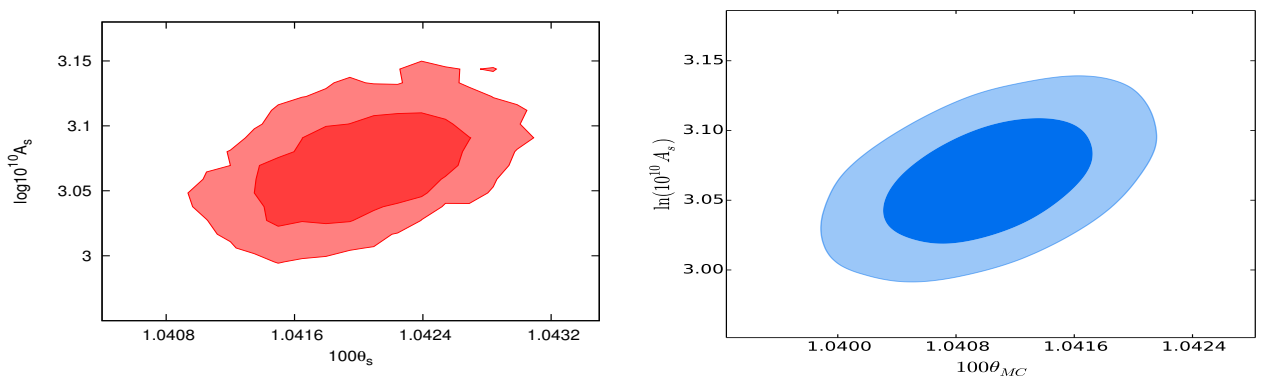


Figura 3.21. Relación entre A_s y θ_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

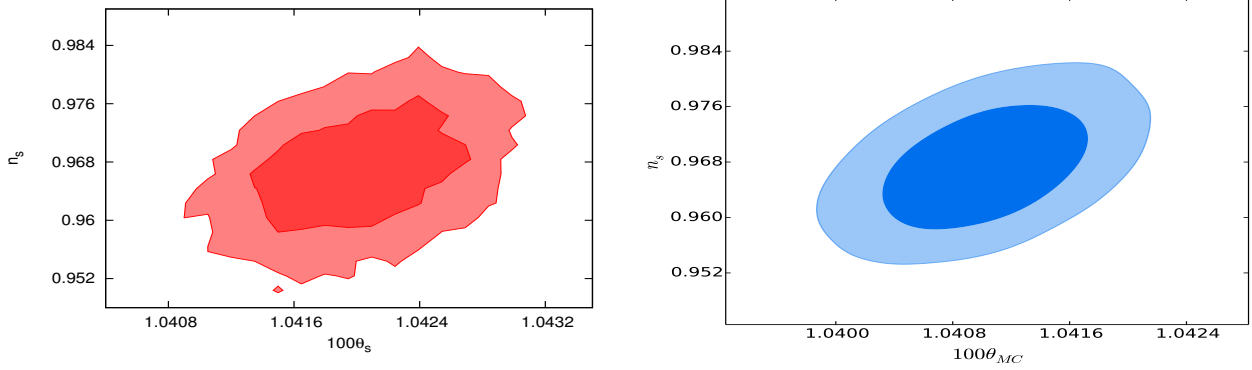


Figura 3.22. Relación entre n_s y θ_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

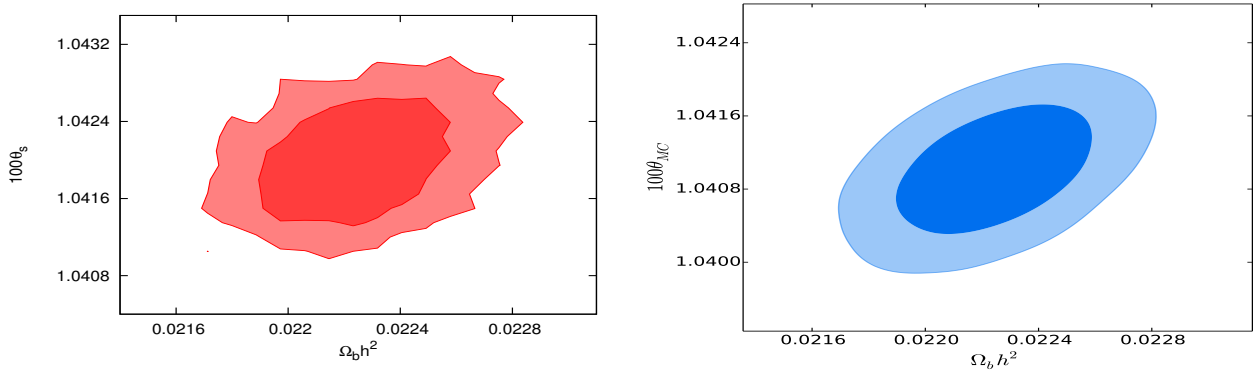


Figura 3.23. Relación entre Ω_b y θ_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

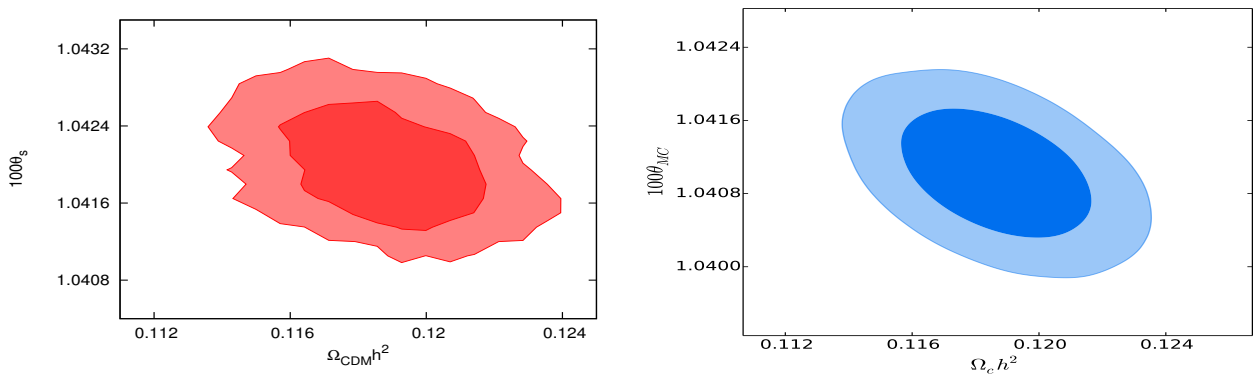


Figura 3.24. Relación entre Ω_{CDM} y θ_s . A izq. resultados de MontePython y a der. resultados de CosmoMC

Capítulo 4

Conclusiones

Para finalizar esta Tesis haremos un breve repaso de lo que hicimos y escribiremos algunas conclusiones respecto de los resultados encontrados.

En un principio contamos brevemente las ecuaciones que gobiernan la Cosmología actualmente y los diferentes modelos cosmológicos que pueden ser descritos por una serie de parámetros. Por otro lado, contamos las misiones satelitales que se fueron realizando a lo largo de la reciente historia hasta llegar al satélite Planck, datos que se han utilizado en la presente Tesis.

Por otro lado, dividimos el estudio de los modelos cosmológicos con los datos observacionales en dos aspectos: integración de las ecuaciones para un dado conjunto de parámetros cosmológicos y la realización de un estudio estadístico para encontrar el conjunto de parámetros que mejor ajusta a las observaciones.

Para el primer aspecto nos encargamos de explicar el funcionamiento de los dos códigos que actualmente están siendo actualizados: **CAMB** y **CLASS**. Hemos integrado un modelo de Λ CDM en sendos códigos utilizando espectros de referencia utilizados en [13]. En los resultados hemos visto que ambos códigos resuelven con gran precisión las ecuaciones del fondo cósmico. Hemos visto en el error relativo entre los resultados de los C_l que la mayor diferencia se da a l chico. Sin embargo, esas diferencias son despreciables frente a la diferencia en tiempo en la que cada código realiza las integraciones.

Por otro lado, **CAMB** posee una descripción detallada en sus archivos dentro del código. Sin embargo, la forma en la que está explicada y ordenada la información dentro del **CLASS** hace que sea más flexible la edición del mismo. Por último, **CAMB** está optimizado para un compilador no-libre y **CLASS** está optimizado para compiladores libres lo que hace una diferencia en favor de este último. Por otro lado, analizando los tiempos de cómputo vemos la gran eficiencia lograda por el paquete **CLASS-MontePython**. A pesar de estas grandes ventajas, en el pos-procesamiento para realizar los gráficos, **CosmoMC** posee mejores librerías y gran bibliografía para consultar. **CLASS** se caracteriza por su fácil uso, la flexibilidad a ser modificado, la precisión y velocidad [9].

En lo que refiere al aspecto estadístico, en el capítulo 3, vimos el método bayesiano y cadenas de Markov (MCMC). Por otro lado, hicimos una descripción del criterio de convergencia de Gelman-Rubin adoptado por ambos códigos. Explicamos brevemente la estructura interna de los códigos.

En el análisis resultante podemos ver que ambos códigos muestrearon el mismo espacio de los parámetros. Además vimos que los resultados son compatibles con la literatura. Recordemos que particularmente, los resultados de Planck 2015 fueron procesados con **CAMB** y **CosmoMC**. Por lo tanto podemos decir que en este modelo cosmológico las soluciones encontradas utilizando el paquete **CLASS-MontePython** fue bueno.

A partir de los resultados obtenidos estamos en condiciones de afirmar que en escenarios clásicos los códigos son igualmente válidos. A futuro se realizarán modificaciones a los códigos para adaptarlos a escenarios alternativos al modelo estándar.

Apéndice A

Relaciones matemáticas

Armónicos esféricos. Los armónicos esféricos son bases ortonormales de funciones. Se utilizan particularmente en el tratamiento del FCR por la simetría esférica con la que se lo modela. Algunas propiedades utilizadas en el texto son:

- la ortonormalidad

$$\int d\Omega Y_{lm}(\hat{n}) Y_{l'm'}^*(\hat{n}) = \delta_{ll'} \delta_{mm'} \quad (\text{A.1})$$

- la relación de clausura

$$\sum_m |Y_{lm}(\hat{n})|^2 = \frac{2l+1}{4\pi} \quad (\text{A.2})$$

- y la representación de una onda plana en armónicos esféricos

$$e^{i\vec{k}\cdot\vec{x}} = 4\pi \sum_{lm} i^l j_l(kx) Y_{lm}(\hat{x}) Y_{lm}^*(\hat{k}) \quad (\text{A.3})$$

Media, varianza teórica y observacional en el espectro de potencia. Sea $x = \Delta(\hat{n})$ un elemento de una muestra cuyo valor esperado es $E(x) = \hat{x} = 0$. La varianza de la muestra será

$$\text{var}(x) = E[(x - \hat{x})^2] \quad (\text{A.4})$$

$$= E(x^2) \quad , \text{ pues } E(x) = \hat{x} = 0 \quad (\text{A.5})$$

$$= E(\Delta(\hat{n})^2) \quad (\text{A.6})$$

$$= \langle \Delta(\hat{n})^2 \rangle. \quad (\text{A.7})$$

Ahora, utilizando la descomposición de la perturbación en armónicos esféricos podemos calcular la varianza teórica de la perturbación

$$\begin{aligned}
\langle (\Delta(\hat{n}))^2 \rangle &= \left\langle \sum_{lm} a_{lm} Y_{lm}(\hat{n}) \sum_{l'm'} a_{l'm'}^* Y_{l'm'}^*(\hat{n}) \right\rangle \\
&= \sum_{ll'} \sum_{mm'} Y_{lm}(\hat{n}) Y_{l'm'}^*(\hat{n}) \langle a_{lm} a_{l'm'}^* \rangle \\
&= \sum_{ll'} \sum_{mm'} Y_{lm}(\hat{n}) Y_{l'm'}^*(\hat{n}) \delta_{ll'} \delta_{mm'} C_l \quad , \text{por 2.14} \\
&= \sum_l \sum_m Y_{lm}(\hat{n}) Y_{lm}^*(\hat{n}) C_l \\
&= \sum_l \frac{2l+1}{4\pi} C_l \quad , \text{por A.2.} \tag{A.8}
\end{aligned}$$

Por otro lado, la varianza observacional de la perturbación se puede expresar de la forma

$$\begin{aligned}
\frac{1}{4\pi} \int (\Delta(\hat{n}))^2 d\Omega &= \frac{1}{4\pi} \int d\Omega \sum_{lm} a_{lm} Y_{lm}(\hat{n}) \sum_{l'm'} a_{l'm'}^* Y_{l'm'}^*(\hat{n}) \\
&= \frac{1}{4\pi} \sum_{lm} \sum_{l'm'} a_{lm} a_{l'm'}^* \int d\Omega Y_{lm}(\hat{n}) Y_{l'm'}^*(\hat{n}) \\
&= \frac{1}{4\pi} \sum_{lm} \sum_{l'm'} a_{lm} a_{l'm'}^* \delta_{ll'} \delta_{mm'}, \quad \text{por A.1} \\
&= \frac{1}{4\pi} \sum_l \sum_m a_{lm} a_{lm}^* \\
&= \sum_l \frac{2l+1}{4\pi} \hat{C}_l. \tag{A.9}
\end{aligned}$$

Apéndice B

Archivo de precisión de CLASS

this file achieves maximum precision for the CMB: each CITT and CIEE, lensed and unlensed, are stable at the 0.01 % level

tol_ncdm_bg = 1.e - 10

recfast_Nz0= 100000

tol_thermo_integration= 1.e - 5

recfast_x_He0_trigger_delta = 0.01

recfast_x_H0_trigger_delta = 0.01

evolver= 0

k_min_tau0= 0.002

k_max_tau0_over_l_max= 3.

k_step_sub= 0.015

k_step_super= 0.0001

k_step_super_reduction= 0.1

start_small_k_at_tau_c_over_tau_h = 0.0004

start_large_k_at_tau_h_over_tau_k = 0.05

tight_coupling_trigger_tau_c_over_tau_h= 0.005

tight_coupling_trigger_tau_c_over_tau_k= 0.008

start_sources_at_tau_c_over_tau_h = 0.006

l_max_g= 50

l_max_pol_g= 25

l_max_ur= 50

l_max_ncdm= 50

tol_perturb_integration= 1.e - 6

perturb_sampling_stepsize= 0.01

radiation_streaming_approximation = 2

radiation_streaming_trigger_tau_over_tau_k = 240.

radiation_streaming_trigger_tau_c_over_tau = 100.

B. Archivo de precisión de CLASS

```
ur_fluid_approximation = 2
ur_fluid_trigger_tau_over_tau_k = 50.

ncdm_fluid_approximation = 3
ncdm_fluid_trigger_tau_over_tau_k = 51.

tol_ncdm = 1.e - 10

l_logstep = 1.026
l_linstep = 25

hyper_sampling_flat = 12.
hyper_sampling_curved_low_nu = 10.
hyper_sampling_curved_high_nu = 10.
hyper_nu_sampling_step = 10.
hyper_phi_min_abs = 1.e - 10
hyper_x_tol = 1.e - 4
hyper_flat_approximation_nu = 1.e6

q_linstep = 0.20
q_logstep_spline = 20.
q_logstep_trapzd = 0.5
q_numstep_transition = 250

transfer_neglect_delta_k_S_t0 = 100.
transfer_neglect_delta_k_S_t1 = 100.
transfer_neglect_delta_k_S_t2 = 100.
transfer_neglect_delta_k_S_e = 100.
transfer_neglect_delta_k_V_t1 = 100.
transfer_neglect_delta_k_V_t2 = 100.
transfer_neglect_delta_k_V_e = 100.
transfer_neglect_delta_k_V_b = 100.
transfer_neglect_delta_k_T_t2 = 100.
transfer_neglect_delta_k_T_e = 100.
transfer_neglect_delta_k_T_b = 100.

neglect_CMB_sources_below_visibility = 1.e - 30
transfer_neglect_late_source = 3000.

l_switch_limber = 40.
accurate_lensing = 1
num_mu_minus_lmax = 1000.
delta_l_max = 1000.
```

Bibliografía

- [1] A. A. Penzias and R. W. Wilson, “A Measurement of Excess Antenna Temperature at 4080 Mc/s,” *ApJ*, vol. 142, pp. 419–421, July 1965.
- [2] J. Lesgourgues, “Cosmology.” <http://lesgourg.web.cern.ch/lesgourg/courses.html>.
- [3] Planck Collaboration, P. A. R. Ade, N. Aghanim, M. Arnaud, M. Ashdown, J. Aumont, C. Baccigalupi, A. J. Banday, R. B. Barreiro, J. G. Bartlett, and et al., “Planck 2015 results. XIII. Cosmological parameters,” *ArXiv e-prints*, Feb. 2015.
- [4] E. Bertschinger, “COSMICS: Cosmological Initial Conditions and Microwave Anisotropy Codes,” *ArXiv Astrophysics e-prints*, June 1995.
- [5] C.-P. Ma and E. Bertschinger, “Cosmological Perturbation Theory in the Synchronous and Conformal Newtonian Gauges,” *ApJ*, vol. 455, p. 7, Dec. 1995.
- [6] U. Seljak and M. Zaldarriaga, “A fast method for computing CMB spectra.,” in *Microwave Background Anisotropies* (F. R. Bouchet, R. Gispert, B. Guiderdoni, and J. Trân Thanh Vân, eds.), pp. 241–244, 1997.
- [7] S. Seager, D. D. Sasselov, and D. Scott, “A New Calculation of the Recombination Epoch,” *ApJ*, vol. 523, pp. L1–L5, Sept. 1999.
- [8] A. Lewis, A. Challinor, and A. Lasenby, “Efficient Computation of Cosmic Microwave Background Anisotropies in Closed Friedmann-Robertson-Walker Models,” *ApJ*, vol. 538, pp. 473–476, Aug. 2000.
- [9] J. Lesgourgues, “The Cosmic Linear Anisotropy Solving System (CLASS) I: Overview,” *ArXiv e-prints*, Apr. 2011.
- [10] H. Kurki–Suonio, “Cosmological Perturbation Theory.” http://theory.physics.helsinki.fi/~cosmology/cosmo2014_12.pdf.
- [11] A. Challinor and H. Peiris, “Lecture notes on the physics of cosmic microwave background anisotropies,” in *American Institute of Physics Conference Series* (M. Novello and S. Perez, eds.), vol. 1132 of *American Institute of Physics Conference Series*, pp. 86–140, May 2009.
- [12] J. Kaplan and for the QUBIC collaboration, “QUBIC, a bolometric interferometer to measure the B modes of the CMB,” *ArXiv e-prints*, Oct. 2009.
- [13] J. Lesgourgues, “The Cosmic Linear Anisotropy Solving System (CLASS) III: Comparison with CAMB for LambdaCDM,” *ArXiv e-prints*, Apr. 2011.
- [14] S. Dodelson, *Modern cosmology*. Academic Press, Elsevier, 2003.
- [15] R. Durrer, *The cosmic microwave background*. Cambridge, 2008.

- [16] B. Audren, J. Lesgourgues, K. Benabed, and S. Prunet, “Conservative Constraints on Early Cosmology: an illustration of the Monte Python cosmological parameter inference code,” *JCAP*, vol. 1302, p. 001, 2013.
- [17] C. G. Scóccola, “Variation of the fine structure constant and the electron mass at early Universe,” *ArXiv e-prints*, June 2009.
- [18] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics, Taylor & Francis, 1995.
- [19] Planck Collaboration, P. A. R. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. J. Banday, and et al., “Planck 2013 results. XVI. Cosmological parameters,” *A&A*, vol. 571, p. A16, Nov. 2014.
- [20] E. Komatsu and Wilkinson Microwave Anisotropy Probe Science Team, “Results from the Wilkinson Microwave Anisotropy Probe (WMAP),” in *American Astronomical Society Meeting Abstracts 221*, vol. 221 of *American Astronomical Society Meeting Abstracts*, p. 429.01, Jan. 2013.
- [21] D. Blas, J. Lesgourgues, and T. Tram, “The Cosmic Linear Anisotropy Solving System (CLASS). Part II: Approximation schemes,” *J. Cosmology Astropart. Phys.*, vol. 7, p. 034, July 2011.