



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN ASTROFÍSICA
INSTITUTO DE ASTRONOMÍA

Nuevos Códigos para predicciones cosmológicas precisas:
Materia Oscura alternativa con supresión del espectro de
fluctuaciones y formación de estructura cósmica en presencia de
neutrinos masivos

T E S I S

Que para optar por el grado de:
DOCTORA EN CIENCIAS (ASTROFÍSICA)

Presenta:
GABRIELA ALEJANDRA AGUILAR ARGÜELLO

Tutor Principal:
Dr. José Octavio Valenzuela Tijerino,
Instituto de Astronomía

CIUDAD UNIVERSITARIA, CD. MX. JUNIO 2022



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Agradezco al CONACyT por la beca de posgrado 455336 que se me fue otorgada durante ocho semestres para poder realizar mis estudios de doctorado, producto del cual surge la presente Tesis.

Reconozco a la UNAM y al Instituto de Astronomía por formarme académicamente desde la Licenciatura y Servicio social, respectivamente. Les agradezco por brindarme los conocimientos, facilidades y recursos para ampliar mis conocimientos y adentrarme en la investigación. Asimismo, aprecio el apoyo económico que se me fue otorgado por el PAEP para asistir a eventos académicos. También, al proyecto PAPIIT IN112518 por otorgarme una beca durante seis meses para concluir esta tesis. Además, ésta recibió apoyo de los proyectos DGAPA-UNAM PAPIIT AG101620, IG101620, IG101222, IN101918 y IN107521.

Esta tesis se benefició de recursos humanos y computacionales del proyecto LAMOD-UNAM, a través de los clusters Atocatl y Tochtli, así como del acuerdo MX28274 con IBM. LAMOD es un esfuerzo colaborativo entre los institutos IA, ICN y el IQ en la UNAM y cuenta con el apoyo de los técnicos: J. Clemente, C. Guzmán, L. Díaz, E. Murrieta, M. Aguilar. También, agradezco a DGTIC-UNAM por brindarme acceso a la supercomputadora Miztli.

Gracias Octavio Valenzuela por ser parte de mi formación académica desde que llegué al IA, por introducirme en el mundo de las simulaciones, por las discusiones científicas, por tu disponibilidad, incluyendo fines de semana y vacaciones, y por todo tu apoyo.

Agradezco a Katrin Heitmann y Salman Habib del Argonne National Laboratory, por motivar en un inicio la realización de este proyecto de investigación. A Arka Bannerjee y Volker Springel por el intercambio útil de información que me ayudó a aclarar muchas de mis dudas sobre sus respectivos trabajos de investigación.

Aprecio las enriquecedoras discusiones científicas con Mariana Jaber, N. Chandrachani, Héctor Velázquez, Arturo Trelles, Santi Roca-Fábrega, y con los miembros de mi comité Tutor: Jorge Cervantes, Axel de la Macorra y Vladimir Ávila. También, agradezco a Julio Clemente y Leobardo Itehua por las discusiones y el valioso soporte técnico que me brindaron para mejorar los códigos desarrollados en la presente tesis.

Finalmente, estoy muy agradecida con mis padres Elodia Argüello y Adalberto Aguilar por todo su cariño, y por motivarme y apoyarme incondicionalmente para cumplir mis metas. A mi hermano Samuel por siempre alentarme. A Louis Breton por todo el amor, apoyo incondicional, escucharme y darme ánimos cuando más lo necesitaba. Ako por ayudarme con mi estrés y acompañarme en mis noches de desvelo.

Resumen

La cosmología contemporánea ha tenido un crecimiento impresionante durante los últimos 20 años, estableciendo el escenario Λ CDM como el más consistente con las observaciones, pero no carente en absoluto de retos como la naturaleza de la materia oscura, la aceleración cósmica o el origen de la asimetría materia-antimateria. Esto se ha logrado en base principalmente a dos factores. El primero corresponde a los sondeos observacionales de la estructura a gran escala del Universo: desde estudios exquisitos de la radiación cósmica de fondo hasta recientemente censos espectroscópicos de millones de galaxias, los cuales se acercan a errores menores al uno por ciento en algunas propiedades del Universo. A diferencia de estudios de la evolución de galaxias, la componente bariónica en este caso si bien tiene un papel, este es menos complejo y el trabajar con la parte no-colisional es una hipótesis razonable. Por lo anterior, el otro factor que ha permitido tal avance han sido las simulaciones cosmológicas no-colisionales de N -cuerpos que siguen el crecimiento lineal y no-lineal de la estructura cósmica. Esto último ha permitido la comparación cuantitativa entre predicciones teóricas y las observaciones. El ir más allá de comparaciones cualitativas ha requerido de simulaciones muy precisas, respecto a efectos antes no importantes dada las limitantes tanto de observaciones como de simulaciones. Un ejemplo de dichos efectos es el ruido de discretización (o *shot-noise* en inglés), el cuál es más crítico para modelos con supresión en el espectro de inhomogeneidades tales como los neutrinos cosmológicos o modelos alternativos de materia oscura (WDM, materia oscura escalar). Las simulaciones requieren, además, tener el tamaño suficiente para controlar la significancia estadística de las conclusiones. Todo esto ha requerido simulaciones cuyo costo computacional es no trivial de alcanzar o en ocasiones es sólo alcanzable por algunos grupos.

Esta tesis se ubica en estos dos últimos puntos. Con tal objetivo, se ha propuesto utilizar un integrador adaptivo: División hamiltoniana jerárquica (HHS, por sus siglas en inglés) con paso de tiempo adaptivo para resolver las ecuaciones de movimiento en simulaciones de N -cuerpos de gran rango dinámico. Este integrador fue originalmente propuesto para problemas colisionales de pocos cuerpos, nosotros lo hemos explorado como alternativa para la parte no-colisional. Se ha encontrado que es aproximadamente reversible, lo cual permite rendimientos mejores, a paridad de precisión, al integrador estándar Leapfrog/Verlet con paso de tiempo adaptivo (le llamaremos AKDK). Lo anterior ocurre debido a que, aunque es aproximada, la naturaleza reversible de HHS permite un crecimiento muy moderado de la energía en contraste con AKDK; además, HHS permite conservación de los momentos lineal y angular a precisión de la máquina. También, se concluyó que la extensión del teorema Kolmogorov-Arnold-Moser (KAM)

para integradores reversibles garantiza la existencia de soluciones estables y favorables. Cabe mencionar que la función de selección de paso de tiempo juega un papel crítico para hallar soluciones con buen rendimiento. Asimismo, se propuso y exploró una optimización de HHS que permite mejorar considerablemente el rendimiento con un impacto moderado en la precisión, siendo aún más competitivo comparado con AKDK. El análisis es similar a implementaciones para dinámica molecular. Finalmente, el integrador HHS se implementó en diferentes arquitecturas computacionales: secuencial, OMP/multicore, ARM, CUDA así como el caso híbrido MPI-CUDA, en donde este último permite usar el código en múltiples GPU's, aumentando de esta forma su competitividad.

Además del tiempo de cálculo, otro de los factores que limitan fuertemente la capacidad predictiva de simulaciones de N -cuerpos, es el ruido de discretización (*shot-noise* en inglés), principalmente a escalas comparables con la escala de flujo libre. Una solución es abandonar la descripción de la materia utilizando partículas y adoptar una descripción continua utilizando las ecuaciones de Boltzmann+Poisson, incluso en el régimen no-lineal. Por lo anterior, en esta tesis se desarrolló un código, al cual nos referiremos como BPM, que resuelve la ecuación de Boltzmann utilizando una aproximación de momentos y técnicas hidrodinámicas de advección, traduciéndolo a problemas reducidos en cada dirección y auxiliándose de un código Partícula-Malla para cerrar la jerarquía de Boltzmann con el tensor de dispersión de velocidades y un método espectral para resolver el campo gravitacional. Esta implementación reduce considerablemente el shot-noise al simular las estructuras de densidad mediante una descripción de medio continuo. Cabe mencionar que lo logra sin necesidad de llegar a los billones de partículas para halos aislados o trillones de partículas para volúmenes cosmológicos. Aunque existe un costo en tiempo de CPU y memoria RAM para esta técnica, no es comparable con las técnicas estándar (p. ej. N -cuerpos completo). Dada la descripción adoptada, el código BPM tiene potencialmente un valor para explorar espacios de parámetros de cosmologías y la interpretación de observaciones con una combinación de eficiencia y precisión. Lo anterior se ilustra en varias aplicaciones a lo largo de esta tesis:

1. El proceso de acreción de galaxias satélites estimula modos de respuesta en los halos de galaxias, esta respuesta depende de las propiedades del medio. Recientemente se anunció la detección de la estela (o *wake* en inglés) y de un modo dipolar en el halo estelar de la Vía Láctea, presumiblemente resultado de la interacción con la Nube Mayor de Magallanes. Capturar tales respuestas suele requerir cientos de millones de partículas, sin embargo, nuestro código BPM, en su versión aislada, lo puede hacer a un costo computacional modesto (128^3 partículas).
2. El llamado problema del exceso de subestructura en el escenario Λ CDM ha estimulado explorar modelos alternativos de materia oscura. La simulación de tales modelos con técnicas de N -cuerpos requiere de millones de horas de CPU y de miles de millones de partículas. Mostramos que nuestro código BPM, en su versión cosmológica, puede simular apropiadamente tales modelos, a un costo reducido, permitiendo exploraciones eficientes y precisas del espacio de parámetros de modelos teóricos.

3. BPM puede simular el efecto de neutrinos masivos en el crecimiento de la estructura cósmica a gran escala. Se inició un estudio de comparación entre BPM y otros códigos como GADGET3 (Quijote Simulations Suite), COLA (*Comoving Lagrangian Acceleration method*) y FastPM. En esta tesis mostramos los resultados preliminares de dicho estudio. Cabe mencionar que éste es un trabajo aún en desarrollo.

Índice general

1. Introducción	10
1.1. Aceleración Cósmica	10
1.2. Explicaciones para la aceleración Cósmica	11
1.3. Composición del Universo	12
1.4. Observables cosmológicos	13
1.5. Sondeos Observacionales para Restringir la Aceleración Cósmica	14
1.5.1. BOSS	15
1.5.2. eBOSS	15
1.5.3. DESI	16
1.5.4. LSST del Observatorio Vera C. Rubin	16
1.6. Simulaciones de la Estructura Cósmica	17
1.7. Objetivo	17
1.8. Estructura de la Tesis	18
2. Neutrinos en cosmología	20
2.1. Modelo estándar y Neutrinos	20
2.2. Más allá del Modelo Estándar: Neutrinos masivos y oscilaciones	22
2.3. ¿Dirac o Majorana?	23
2.4. Jerarquía de los neutrinos	24
2.5. Asimetría Bariónica y Neutrinos	25
2.6. Importancia de los neutrinos en Cosmología	26
2.7. El fondo cósmico de neutrinos	27
2.8. Número efectivo de neutrinos y radiación extra	27
2.9. Efectos de los neutrinos sobre observables cosmológicos	28
2.9.1. Flujo libre de los Neutrinos	28
2.9.2. Efecto de los Neutrinos en el Fondo Cósmico de Microondas	29
2.9.3. Efecto de los Neutrinos en el Espectro de potencias de Materia	29
2.10. Restricciones actuales sobre Σm_ν	30
2.11. Pronósticos para Σm_ν	31
3. Simulaciones Cosmológicas de N-cuerpos	34
3.1. Relevancia de los códigos de N -cuerpos en el entendimiento del Universo a Gran Escala	34
3.1.1. Características de las simulaciones cosmológicas en la era de los censos contemporáneos de galaxias	35

3.2.	Un nuevo Código Paralelo de suma directa con paso de integración adaptivo y reversible: NP _{splitt}	37
3.2.1.	Introducción	37
3.2.2.	Integradores	38
3.2.3.	Implementación del código	40
3.2.4.	Pruebas de precisión del código: Binarias	45
3.2.5.	Caracterización del código	46
3.2.6.	Discusión y conclusiones	62
4.	Un nuevo integrador adaptivo reversible para simulaciones no-colisionales: división hamiltoniana jerárquica	65
4.1.	Introducción: Evaluando la pertinencia del integrador reversible de división hamiltoniana jerárquica	65
5.	Métodos para simular el efecto de neutrinos masivos en el crecimiento de la estructura cósmica	84
5.1.	Relevancia de las simulaciones con neutrinos masivos	84
5.2.	Descripción más general	85
5.3.	Métodos basados en partículas	86
5.4.	Métodos basados en malla	88
5.5.	Métodos perturbativos+ N -cuerpos	88
5.6.	Método de momentos de la ecuación de Boltzmann	89
5.7.	Limitantes y su impacto en las restricciones cosmológicas	90
6.	Código Boltzmann-PM o BPM	93
6.1.	Algoritmo	93
6.2.	Implementación	95
6.2.1.	Código PM	95
6.2.2.	Código Hidro-Boltzmann	96
6.2.3.	Código Boltzmann-PM	98
6.2.4.	Código para dos especies	98
6.3.	Pruebas numéricas de los códigos	100
6.3.1.	Pruebas numéricas para PM	100
6.3.2.	Pruebas numéricas para Hidro-Boltzmann	103
6.3.3.	Pruebas numéricas para Boltzmann-PM	105
6.3.4.	Pruebas numéricas para 2-especies	107
6.4.	Potencial del código Boltzmann-PM	109
7.	Simulando la respuesta de halos oscuros a la acreción de galaxias satélites con BPM	114
7.1.	Introducción: Modos de respuesta en halos de galaxias, fricción dinámica.	114
8.	Simulando la estructura en modelos de materia oscura con supresión en el espectro de potencias.	126
8.1.	Free-Streaming y la función de masa de halos, Espectro de Potencias .	126

8.2.	El problema de los satélites faltantes o la conexión entre subestructura y la longitud de flujo libre de la materia oscura en Cosmología	127
8.3.	Simulaciones WDM	128
8.4.	Shot-Noise y subestructura espuria de materia oscura	128
8.5.	Simulaciones de la subestructura en modelos de materia oscura tibia libres de ruido	131
8.5.1.	Simulaciones WDM con BPM	131
8.5.2.	Simulaciones Híbridas: BPM+GADGET-4	135
8.5.3.	Simulaciones WDM BPM+GADGET-4	137
8.6.	Otros modelos de Materia Oscura con amortiguación en el MPS	140
8.7.	Discusión	141
9.	Estado actual de las técnicas usadas en simulaciones cosmológicas de formación de estructura con neutrinos masivos	143
9.1.	Descripción de los códigos considerados	144
9.1.1.	Quijote	144
9.1.2.	FastPM	145
9.1.3.	MG-PICOLA	145
9.1.4.	BPM	146
9.2.	Resultados preliminares	147
9.3.	Discusión	150
10.	Discusión, trabajo futuro y conclusiones	152
10.1.	Futuro del código	156
A.	Deducción de las ecuaciones usadas en Boltzmann-PM	158
A.1.	Ecuación de Boltzmann no-colisional en la norma newtoniana	158
A.2.	Ecuación cosmológica tipo Continuidad	159
A.3.	Ecuación cosmológica tipo Euler	161
B.	Técnica Partícula-Malla	163
B.1.	Asignación de Densidad	164
B.2.	Cálculo del Potencial	165
B.3.	Cálculo de Fuerzas	166
B.4.	Integración ecuaciones de movimiento	166
C.	Pseudo-códigos del integrador de división hamiltoniana jerárquica	167
D.	Artículos Aceptados	169
D.1.	Artículos Producto de la Presente Tesis	169
D.2.	Otros Artículos	169

Capítulo 1

Introducción

El presente capítulo establece el marco en el cual se ubican los casos científicos que motivan el presente proyecto de tesis. Esta marco es la cosmología contemporánea, en específico los problemas de la materia y energía oscuras, así como las restricciones que la cosmología puede establecer sobre física más allá del Modelo Estándar de Partículas, como la masa u otras propiedades de los neutrinos. El texto a continuación presenta una versión actualizada aunque muy breve de tales potencialidades y su perspectivas a corto plazo.

1.1. Aceleración Cósmica

La relatividad general (GR, por sus siglas en inglés) de Einstein predice que, para un universo homogéneo lleno de materia y/o radiación, la expansión cósmica se ralentizará con el tiempo. Sin embargo, a finales de los 90's, dos estudios independientes de supernovas lejanas descubrieron que la expansión del Universo se había acelerado durante los últimos cinco mil millones de años [1, 2]. Debido a este descubrimiento y a los datos del fondo cósmico de microondas (CMB, por sus siglas en inglés), al agrupamiento de galaxias a gran escala, y a las estimaciones directas de la densidad de materia, la tasa de expansión (H_0), y la edad del Universo, el modelo Einstein-de Sitter ($\Omega_m = 1$, es decir, un Universo dominado por materia) fue descartado y el modelo de constante cosmológica¹ ($\Omega_m = 0.3$, $\Omega_\Lambda = 0.7$) se estableció como la alternativa preferida. Poco después, la evidencia de CMB para un universo espacialmente plano [4, 5] fortaleció el caso para la aceleración cósmica, y eliminó (en conjunto con las otras pruebas observacionales), la alternativa de expansión libre con $\Omega_m \ll 1$ y $\Omega_\Lambda = 0$.

Actualmente, un universo acelerado está bien establecido por múltiples e independientes evidencias a partir de mediciones cosmológicas precisas. La aceleración cósmica podría indicar la necesidad de una modificación a GR para escalas cosmológicas o la existencia de una nueva componente en el Universo con propiedades físicas inusuales

¹El término de constante cosmológica fue originalmente introducido en 1917 por Einstein [3]. Aunque Einstein no lo planteó de esta manera, se puede considerar que la constante cosmológica, Λ , representa una densidad de energía constante del vacío, cuyo efecto gravitacional repulsivo equilibra la gravedad atractiva de la materia.

que provoque dicha aceleración. En cualquier caso, la explicación a la aceleración cósmica tiene implicaciones profundas en la física contemporánea. Por lo que la pregunta crítica es: ¿Por qué el Universo se está acelerando?

1.2. Explicaciones para la aceleración Cósmica

El modelo Λ CDM (es decir, Constante cosmológica, Λ , + Materia Oscura Fría, CDM, por sus siglas en inglés) es el paradigma cosmológico actual, pues es el modelo más simple que proporciona una explicación a las propiedades del cosmos como la existencia y estructura del fondo cósmico de microondas (CMB, por sus siglas en inglés), las abundancias primordiales, la formación de la estructura a gran escala y la expansión acelerada del Universo. Este escenario asume el modelo estándar de partículas y que GR sigue siendo válida en escalas cosmológicas, por lo que para explicar la aceleración cósmica introduce una nueva componente cosmológica llamada “energía oscura” (DE, por sus siglas en inglés), la cuál es asociada a una densidad uniforme con energía constante y parametrizada mediante una constante cosmológica, Λ . Sin embargo, si la energía oscura es asociada a la energía del vacío de todos los campos en el universo, el valor sugerido por las observaciones para Λ es muy pequeño (~ 120 órdenes de magnitud) en comparación al valor esperado por argumentos teóricos en base a teorías cuánticas de campos [6], por lo que la naturaleza de la energía oscura no es explicada. Otro de los problemas del modelo Λ CDM es que no explica por qué el valor de la constante cosmológica es tal que el universo apenas ha entrado en la época de dominio de energía oscura. A pesar de que la naturaleza de sus componentes no es explicada, el éxito del escenario Λ CDM en describir el Universo a gran escala lo convierte en un modelo efectivo el cual es guía de potenciales modelos más detallados, si existen, que busquen explicar la naturaleza de sus componentes.

Una alternativa a la constante cosmológica es que la expansión acelerada es impulsada por un nuevo campo, ϕ , con presión negativa cuya densidad de energía cambia con el tiempo [7]. A este tipo de modelos se les conoce como energía oscura dinámica o campo escalar. Típicamente son caracterizados mediante el parámetro de ecuación de estado, $w(z) \equiv P/\rho$, el cual es el cociente entre la presión, P , y la densidad de energía, ρ , del campo. Un ejemplo de este tipo de modelos es el conocido como “quinta-esencia” [8], en el cual la densidad de energía está dominada por el potencial del campo escalar, dando como resultado una presión efectiva negativa (con $w_\phi(z) \geq -1$) que lleva a la aceleración cósmica. La descripción de esta alternativa a Λ implica resolver simultáneamente la ecuación de Boltzmann en alguno de sus límites (Schrödinger, Klein-Gordon, Euler) acoplada a la ecuación de Poisson.

Alternativamente a introducir una nueva componente de energía, la expansión acelerada del Universo podría ser evidencia de la necesidad de una extensión de GR para escalas cosmológicas. El modificar GR en dichas escalas puede conducir por sí sola a una expansión acelerada (p. ej. [9, 10, 11]). Un ejemplo popular de modelo de gravedad modificada son los modelos $f(R)$ (p. ej. [12, 13]), en los que el escalar de Ricci, R , en la acción Einstein-Hilbert es reemplazado con alguna función $f(R)$ de orden superior. Existe una variedad de modelos de gravedad modificada, sin embargo, construir mo-

delos viables es un reto no completamente explorado, debido a que es fácil introducir inconsistencias teóricas, pero sobre todo porque deben satisfacer las constricciones rigurosas impuestas por mediciones precisas en la escala del Sistema Solar (p. ej. [14, 15, 16]). Es importante enfatizar que este tipo de modelos puede verse de manera unificada como la inclusión de un campo escalar y la modificación del límite a campos débiles y velocidades pequeñas, que es la ecuación de Poisson.

Una extensión al modelo cosmológico estándar Λ CDM es la inclusión explícita de una componente de neutrinos masivos. Los experimentos en laboratorio han establecido que al menos dos estados de neutrinos son masivos (p. ej. [17, 18]). A pesar de que estos experimentos restringen sólo las diferencias cuadráticas de sus masas, han concluido que al menos una de las especies de los neutrinos tiene masa ~ 0.06 eV. Debido al “flujo libre” (FS, por sus siglas en inglés) de los neutrinos, la amplitud del espectro de potencias de materia se vería disminuido para escalas menores a la de FS, por lo que el no considerar el efecto de los neutrinos puede erróneamente llevar a concluir la necesidad de un nuevo tipo de energía o de gravedad en base al análisis de las oscilaciones acústicas de bariones (BAO, por sus siglas en inglés). La nueva generación de sondeos de galaxias permitirá restringir la suma de las masas de neutrinos con precisión de $3-5\sigma$ ([19]) verificando o sugiriendo extensiones para el modelo estándar de partículas. En el siguiente capítulo (cap. 2), se discutirá con mayor detalle a los neutrinos en el contexto cosmológico.

1.3. Composición del Universo

En base a su contribución energética, las componentes del Universo pueden ser catalogadas como: partículas no-relativistas, partículas relativistas y energía oscura.

La evolución de la densidad de energía de las partículas no-relativistas está dada por $\rho \propto a^{-3}$ (con a el factor de escala), por lo que su contribución se diluye con la expansión del Universo. Hay dos componentes del Universo que entran en esta categoría:

- **Materia bariónica o materia ordinaria:** en cosmología, esta componente incluye a todas las partículas no-relativistas del modelo estándar (de partículas), y a sus configuraciones más complejas como átomos, nubes de gas o polvo, estrellas, galaxias, etc.
- **Materia oscura:** observaciones del universo a escalas grandes y pequeñas sugieren la existencia de un tipo de materia que parece ser invisible al espectro electromagnético para poder explicar la dinámica de diferentes sistemas astronómicos (p. ej. curvas de rotación de galaxias espirales). De forma genérica, a este tipo de materia se le denomina “materia oscura” (DM, por sus siglas en inglés). A pesar de que hay candidatos para este tipo de materia, su naturaleza sigue siendo desconocida, y existen diferentes esfuerzos contemporáneos que buscan explicarla. En base a las velocidades al momento del desacople se pueden definir tres tipos genéricos de DM: “caliente” (materia relativista al momento del desacople), “fría” (materia no-relativista en el desacople) y “tibia” (se desacopla siendo relativista

pero sus velocidades no son tan grandes, permitiendo formar estructura a escalas galácticas).

En cuanto a las partículas relativistas, su densidad de energía evoluciona como $\rho \propto a^{-4}$, y hoy día su contribución a la materia-energía del Universo es prácticamente nula en comparación con el resto de las componentes. El fotón es la principal partícula que constituye esta componente. Otra partícula que puede aportar a esta componente es el neutrino, sin embargo, como veremos en el capítulo siguiente (cap. 2), ésta es un caso especial, ya que a través de experimentos de laboratorio se sabe que los neutrinos tienen masa. Lo anterior hace que comiencen teniendo un comportamiento relativista y, conforme el Universo se enfría, se comportan como materia.

Como ya mencionamos previamente, la energía oscura se introduce para explicar la expansión acelerada del Universo, pero su naturaleza es aún desconocida. Para el modelo estándar Λ CDM, esta componente es una constante cosmológica, y su contribución energética es constante. En el caso de los modelos de energía oscura dinámica, su contribución variará con el corrimiento al rojo (z) dependiendo de su ecuación de estado, $w(z)$:

$$\rho \propto \exp \left[3 \int_0^z \frac{[1 + w(z)]}{1 + z} dz \right] \quad (1.1)$$

En el paradigma cosmológico actual, el modelo Λ CDM, tan sólo alrededor del 4% de la materia-energía del Universo está constituido por la materia bariónica, aproximadamente otro 23% está compuesto de materia oscura, mientras que el 73% restante lo conforma la energía oscura. Esta composición del Universo ha sido estimada a través de distintos estudios cosmológicos.

1.4. Observables cosmológicos

Extensiones al modelo estándar de partículas y campos, a la gravedad o a la física de vacío que aspiren explicar el Universo y la naturaleza de sus componentes, están acotadas por las propiedades cosmológicas observadas. Por lo que brevemente resumiré algunas de las pruebas observacionales más relevantes. Para consultar una descripción más detallada ver, p. ej. [20].

- Anisotropías del CMB: proporcionan restricciones sobre la geometría del Universo, a partir de la ubicación de los picos acústicos en su espectro de potencias angular (APS, por sus siglas en inglés). También, ponen restricciones sobre las densidades de materia y bariones, a partir de la amplitud de los picos de su APS. Finalmente, el CMB restringe la normalización de la amplitud de las fluctuaciones de materia en la época de recombinación, z_{rec} , a partir de la amplitud de sus propias fluctuaciones. A pesar de que las anisotropías del CMB por sí solas proveen una información limitada de la energía oscura, son una prueba fundamental cuando se combinan con sondeos de bajo corrimiento al rojo.
- Oscilaciones acústicas de bariones (BAO, por sus siglas en inglés): antes de la época de recombinación, la presión de radiación de los fotones previene la formación de estructura mientras los bariones y fotones están fuertemente acoplados.

Esta competencia entre gravedad y la presión generan oscilaciones en el plasma primordial, las cuales imprimen una escala característica en la agrupación de materia y galaxias. Esta huella puede ser observada usando cantidades estadísticas como la función de correlación de galaxias o el espectro de potencias de materia (MPS, por sus siglas en inglés). Con BAO se pueden medir la distancia de diámetro angular y la tasa de expansión del Universo.

- Supernovas de tipo Ia (SNeIa): son producidas por la explosión termonuclear de una (estrella) enana blanca en un sistema binario. Dichas explosiones tienen un pico de luminosidad que está correlacionado con la forma de sus curvas de luz, lo que potencialmente proporciona una estimación de distancia. Los sondeos de SNeIa buscan medir las curvas de luz y su correspondiente corrimiento al rojo, z , de cientos o miles de SNeIa para obtener una correlación distancia- z con alta precisión. Sin embargo, esta prueba observacional aún tiene sistemáticos (p. ej. calibración fotométrica, evolución del corrimiento al rojo de la propia población de SNeIa) que pueden producir algunos sesgos. Esta prueba es complementaria a BAO.
- Determinaciones directas de la constante de Hubble a $z = 0$, H_0 : combinar una medición independiente y exacta de la constante de Hubble, H , con datos del CMB daría como resultado una prueba fuerte para los modelos de energía oscura e impondría restricciones fuertes a los parámetros cosmológicos, más aún si H fuera medida al día de hoy. Sin embargo, medir de forma directa H_0 tiene el desafío de establecer las distancias de las galaxias que están lo suficientemente lejos como para que sus velocidades peculiares sean pequeñas en comparación con la velocidad de expansión.
- Distorsiones espacio-corrimento al rojo (RSD, por sus siglas en inglés): los movimientos peculiares de las galaxias provocan que el agrupamiento de galaxias a gran escala sea anisotrópico en el espacio de corrimiento al rojo. Estas distorsiones pueden verse reflejadas, por ejemplo, en el pico de BAO, provocando un ensanchamiento y desplazamiento de éste. Caracterizar dichas distorsiones proporciona una prueba al crecimiento de estructura en el Universo, y puede ayudar a romper degeneraciones entre algunos parámetros cosmológicos. Sin embargo, aún hay incertidumbres en el modelado de la evolución no-lineal y el sesgo no-lineal entre las distribuciones de galaxias y materia.

1.5. Sondeos Observacionales para Restringir la Aceleración Cósmica

El establecimiento de cualquiera de las explicaciones para la energía oscura causaría un cambio dramático en nuestra comprensión del Universo en su conjunto. Es por esto que se ha creado un amplio rango de esfuerzos experimentales (p. ej. BOSS [21], eBOSS [22], DESI [23], LSST [24]), que tienen como objetivo medir la historia de expansión y el crecimiento de estructura en el cosmos con una precisión del $\sim 1\%$ o

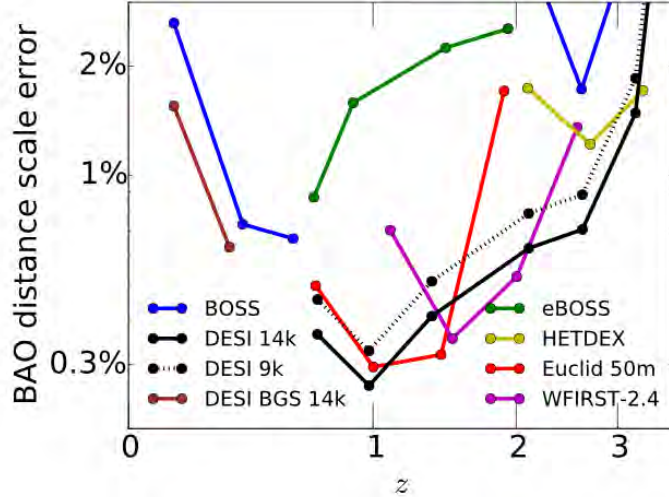


Figura 1.1: Error fraccional en la escala de BAO como función de z para sondeos de galaxias contemporáneos y futuros.

mejor en BAO (ver figura 1.1). En la figura 1.1 se muestra el error fraccional en la escala de distancia de BAO como función del corrimiento al rojo, z , presentado en bins comparables para DESI, BOSS, eBOSS, Euclid [25], WFIRST [26] y HETDEX [27]. En esta figura podemos observar que DESI proporcionará las mejores mediciones en gran parte de la región y que compite con los sondeos de tipo “spaced-based”, que llegarán más tarde.

El logro de las metas científicas de estos sondeos requiere de catálogos sintéticos realistas basados en simulaciones numéricas que permitan calcular el crecimiento no-lineal de la estructura cósmica y predigan la dependencia de las cantidades observables en los sondeos con los parámetros cosmológicos.

1.5.1. BOSS

El sondeo espectroscópico BOSS (del inglés Baryon Oscillation Spectroscopic Survey, [21]) mapeó la distribución espacial de 1.5 millones de galaxias luminosas rojas (LRGs, por sus siglas en inglés) a $z = 0.7$ y 160,000 cuásares (QSOs) en el rango de corrimientos al rojo $2.2 < z < 3.0$ cubriendo una área en el cielo de $10,000 \text{ deg}^2$. BOSS empleó el telescopio de 2.5 metros del SDSS-III (del inglés Sloan Digital Sky Survey-III) ubicado en el Observatorio Apache Point (APO) y tomó datos desde otoño de 2009 hasta otoño de 2016.

1.5.2. eBOSS

El sondeo eBOSS (del inglés Extended Baryon Oscillation Spectroscopic Survey, [22]) es una extensión de BOSS, y usa el espectrógrafo de BOSS en APO. eBOSS obtuvo el espectro de 375,000 LRGs en el rango de corrimientos al rojo $0.6 < z < 0.8$ sobre una área de $7,500 \text{ deg}^2$; 260,000 galaxias de línea de emisión (ELGs, por sus

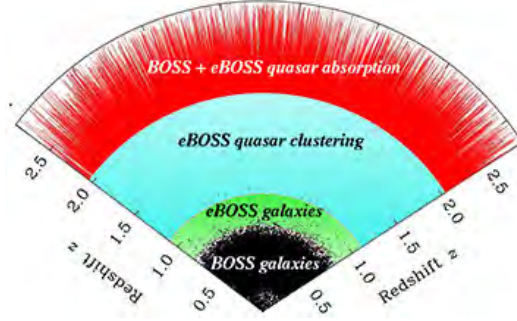


Figura 1.2: Cobertura de eBOSS del Universo.

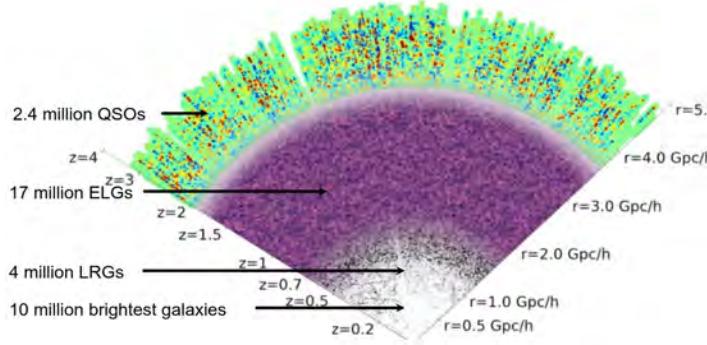


Figura 1.3: Cobertura de DESI del Universo.

siglas en inglés) en el rango de corrimientos al rojo $0.6 < z < 1.0$ sobre una área de $1,500 \text{ deg}^2$; y 740,000 QSOs en el rango de corrimientos al rojo $0.9 < z < 3.5$ sobre una área de $7,500 \text{ deg}^2$ (ver figura 1.2), de esta forma complementa a mapeos tridimensionales previos de la estructura a gran escala como BOSS. El sondeo comenzó a tomar datos en otoño de 2014 y terminó en primavera de 2020.

1.5.3. DESI

DESI (del inglés Dark Energy Spectroscopic Instrument, [23]) es un sondeo espectroscópico de cinco años que observará 4 millones de LRGs en el rango de corrimientos al rojo $0.4 < z < 1.0$, 17.1 millones de ELGs a $0.6 < z < 1.6$ y 1.7 millones de QSOs a $z < 2.1$ (ver figura 1.3), cubriendo una área del cielo de $14,000 \text{ deg}^2$ y abarcando un volumen de $\sim 50 (\text{Gpc } h^{-1})^3$. El instrumento DESI ha sido montado en el Telescopio Mayall de 4 metros en el Observatorio Nacional de Kitt Peak desde 2018, y comenzó a tomar datos en otoño de 2019.

1.5.4. LSST del Observatorio Vera C. Rubin

LSST (del inglés Legacy Survey of Space and Time, [24]) es un experimento fotométrico en el óptico, montado en el observatorio Vera C. Rubin en el norte de Chile. Con un telescopio de 8.4 metros observará alrededor de 37 mil millones de galaxias y estrellas cubriendo un área de $18,000 \text{ deg}^2$ en el cielo. Se espera que la totalidad de

las operaciones para un estudio de diez años comience a partir del año 2024. En particular el lente gravitacional y el corrimiento al rojo fotométrico permitirán estudiar la estructura cósmica a diversas escalas, con una significancia estadística sin precedentes.

1.6. Simulaciones de la Estructura Cósmica

La precisión que están alcanzando los sondeos obliga a que las simulaciones numéricas requeridas para orientar campañas de observación e interpretación de sus resultados, tengan una precisión similar; lo cual se traduce a un alto número de partículas ($\sim 10^{12}$) y volúmenes comparables ($\gtrsim 1$ Gpc). Por lo anterior, las simulaciones requieren de métodos de cálculo e infraestructura con una alta eficiencia de cómputo. Durante los últimos 40 años se ha gestado un importante desarrollo en la disponibilidad y utilidad de las técnicas y recursos computacionales. Esto ha llevado a un aumento igualmente impresionante en la calidad y la precisión de las simulaciones de la formación de la estructura cósmica, principalmente utilizando la aproximación de N -cuerpos, desde trabajos pioneros con apenas partículas hasta los esfuerzos contemporáneos con nueve o más órdenes de magnitud de incremento en el número de partículas ([28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40]). Recientemente, el código híbrido HACC ([41, 42, 43]) revolucionó el campo de las simulaciones de N -cuerpos, superando la barrera de 10^{12} partículas, alcanzando de esta forma los niveles de precisión estadística y espacial necesarios para interpretar los sondeos de galaxias contemporáneos y futuros, dentro del modelo estándar Λ CDM. Cabe mencionar que a pesar de tan impresionante desarrollo, incluir la física de modelos alternativos a Λ CDM es aún un reto, aunque el problema esta siendo activamente estudiado. De manera particular la pregunta sobre si la técnica de N -cuerpos es la estrategia adecuada para simular el crecimiento de la estructura cósmica, está en amplio debate y la exploración de estrategias alternativas es una labor activa.

1.7. Objetivo

Dicho lo anterior, este proyecto doctoral se ubica dentro de la Cosmología de alta precisión y las simulaciones que requiere tal etapa. En particular, nos interesan los problemas abordables por la colaboración DESI ([23]), en donde el principal esfuerzo en simulaciones numéricas, al momento, ha sido enfocado al modelo estándar Λ CDM, por lo que el nicho de modelos alternativos y bien fundamentados desde el punto de vista teórico, como es el caso de la inclusión explícita de una componente de neutrinos masivos o de materia oscura alternativa a la fría, son los que pretendemos abordar. Existen más posibilidades, pero solo lo mencionaremos en el capítulo final. Por lo tanto, **este proyecto doctoral tiene como objetivo desarrollar o evaluar, según sea el caso, una nueva generación de herramientas computacionales para realizar predicciones precisas, mediante simulaciones numéricas completas, para el crecimiento cuasi-lineal o no-lineal de la estructura cósmica a gran escala y para la distribución de galaxias en modelos alternativos al Λ CDM, enfocándonos principalmente en cosmologías con neutrinos masivos o con**

supresión en el espectro de potencias, con un costo computacional reducido. Estas predicciones pueden ser utilizadas para la interpretación de las observaciones de los sondeos BOSS, eBOSS, DESI, LSST cuando los datos estén disponibles.

Para cumplir con el objetivo del proyecto, implementé un nuevo integrador para simulaciones de N -cuerpos de división hamiltoniana jerárquica (HHS, por sus siglas en inglés) con pasos de tiempo adaptivos y discutí su eficiencia en simulaciones no-colisionales. Dicho algoritmo lo implementé en un nuevo código de integración directa desarrollado por mí: Np_{splitt} (uno de los productos de la presente tesis). Se desarrolló además un código Partícula-Malla (PM). Finalmente, desarrollé el código BPM (principal producto de mi proyecto doctoral) que permite realizar simulaciones cosmológicas precisas para modelos alternativos al Λ CDM representados por un campo escalar (p. ej. WDM, DE dinámica) el cual se resuelve de forma auto-consistente con las partículas de materia oscura, y utilizamos dicho código para realizar simulaciones con una inclusión explícita de una componente de neutrinos masivos. El código BPM tiene una versión aislada multi-especie la cual es aplicable a problemas dinámicos.

1.8. Estructura de la Tesis

Dado que la motivación principal para desarrollar el código BPM es realizar simulaciones de cosmologías con neutrinos masivos. En el capítulo 2 presentamos una revisión del papel de los neutrinos en el contexto cosmológico, discutiendo la importancia de ser estudiados por sus implicaciones en nuestra comprensión de la física, y cómo la cosmología contemporánea puede aportar al estudio de éstos.

El capítulo 3 discute la importancia de las simulaciones de N -cuerpos dentro de la cosmología actual. También, en este capítulo presento el código Np_{splitt} de integración directa que es uno de los que desarrollé y es uno de los productos de esta tesis. Algunos códigos contemporáneos de alto rendimiento como HACC utilizan tal técnica para la parte de alta resolución, esto motivó que Np_{splitt} tuviera la posibilidad de aprovechar diferentes arquitecturas computacionales. Como mencioné previamente, en este código implementé el nuevo integrador HHS. El capítulo 4 evalúa la viabilidad de usar dicho integrador para simulaciones no-colisionales en términos de precisión y rendimiento, lo que constituye otro producto de esta tesis.

El capítulo 5 discute las distintas técnicas en el estado del arte para simulaciones cosmológicas con neutrinos masivos, y motiva el uso de la técnica implementada en el código BPM (otro de los productos de esta tesis). En el capítulo 6 se describe el algoritmo de dicha técnica y su implementación en BPM, así como algunas pruebas numéricas a dicho código. BPM tiene la posibilidad de incluir de manera natural física de los neutrinos o partículas de materia oscura, en el término no-homogéneo de la ecuación tipo Euler, tal posibilidad es complicada y no muy precisa en simulaciones de N -cuerpos tradicionales.

En el capítulo 7 se muestra una aplicación novedosa del código BPM: el estudio de los modos de respuesta en halos de galaxias originados por la acreción de galaxias satélite. Capturar tales modos en simulaciones es un reto computacional, p. ej. para simulaciones de N -cuerpos tradicionales se requiere de un gran número de partículas (\sim

$10^8 - 10^9$) para disminuir el efecto del shot-noise inherente de los códigos de N -cuerpos. Por lo que esta aplicación representa una prueba exigente para BPM independiente de las simulaciones cosmológicas a gran escala. Cabe mencionar que este estudio de los modos de respuesta con BPM es otra contribución de la presente tesis, ya que además de usar por primera vez la técnica de momentos de Boltzmann de BPM en este contexto dinámico se estudia la dependencia de la respuesta en halos con diferente perfil de densidad.

En el capítulo 8 proponemos usar el código BPM en el contexto de cosmologías WDM y comparamos los resultados con GADGET-4. Por ello, en el capítulo se discuten brevemente los aspectos más importantes en dichos modelos. También se muestran los resultados de las simulaciones realizada con BPM, GADGET-4 y una combinación de estos, mostrando en particular la abundancia de subestructura estimada en cada uno de ellos.

Como hemos mencionado, la precisión que están alcanzando los sondeos contemporáneos y futuros requieren de simulaciones con una precisión similar. Por lo tanto, los métodos de cálculo de las simulaciones deben capturar de forma precisa los modelos físicos, además de que deben ser altamente eficientes, computacionalmente hablado. En el contexto de simulaciones con neutrinos masivos existen diferentes técnicas en el estado del arte, sin embargo, aún no hay un estudio cuantitativo que compare dichas técnicas entre sí. Motivados por esto, en el capítulo 9 presentamos resultados preliminares de un estudio de comparación (aún en desarrollo) entre algunas de las técnicas contemporáneas, incluyendo el código BPM.

Finalmente, en el capítulo 10 se presenta una discusión y conclusiones del trabajo realizado en la presente tesis, así como una breve perspectiva del trabajo futuro de la misma.

Capítulo 2

Neutrinos en cosmología

En este capítulo presentamos una breve revisión de la fenomenología física de los neutrinos, enfatizando en aspectos que potencialmente podrán ser abordados por los censos de galaxias futuros y los cuales se beneficiarán de la exploración de las consecuencias de la física de los neutrinos, a través de simulaciones.

2.1. Modelo estándar y Neutrinos

En 1930 W. Pauli planteó la hipótesis de la existencia de una partícula sin carga y con una masa despreciable para compensar la aparente pérdida de energía y momento angular en el decaimiento beta de los neutrones. Dicha partícula fue llamada Neutrino, que por las propiedades que debía tener no podía ser detectada en esa época. Fue hasta 1956 que C. Cowan y F. Reines observaron la primera evidencia de neutrinos ([44, 45]), mediante decaimiento beta inverso. Para ello, bombardearon tanques de agua con un haz generado por un reactor nuclear, que se esperaba generara flujos de neutrinos del orden de 10^{12} a 10^{13} neutrinos por segundo por cm^2 , y observaron la emisión subsecuente de rayos gamma producidos por la interacción de los neutrinos con los protones del tanque de agua, demostrando de esta forma la existencia de los neutrinos.

En 1962 L. Lederman, M. Schwartz y J. Steinberger descubrieron la existencia de un segundo tipo de neutrino: el neutrino muónico, ν_μ ([46]). El experimento que realizaron consistió en usar un haz energético de protones para producir piones que decaen en muones y neutrinos. Estos últimos son los únicos que llegan al detector, en donde un evento de neutrino será detectado mediante la aparición de un muón en el detector. Con este experimento se demuestra que los neutrinos vienen en “sabores”, pues previo al experimento sólo se conocía el neutrino electrónico, ν_e .

En 1975, M. Perl et al. [47] anunció evidencia de la producción de eventos que contienen electrones y muones. Estos eventos indicaron la producción de un nuevo leptón pesado que podía decaer para producir electrones o muones, junto con los neutrinos asociados: el tauón. Debido a que tanto el electrón como el muón tenían un neutrino asociado, se teorizó que el tauón también debía tener un tipo de neutrino asociado: el neutrino tauónico, ν_τ , y que se produciría cuando el tauón decayera. Fue hasta el

2000 que la colaboración DONUT (del inglés Direct Observation of Nu Tau), en Fermilab, descubrió experimentalmente el neutrino tauónico, mediante la observación de los tauones y su decaimiento a partir de interacciones de corrientes cargadas de neutrinos tauónicos.

Ahora bien, el modelo estándar de física de partículas (SM, por sus siglas en inglés) fue desarrollado a mediados de los 70's y constituye una teoría para describir tres de las cuatro fuerzas fundamentales que gobiernan las interacciones de la materia: las fuerzas electromagnética, débil y fuerte. Además, describe la estructura fundamental de la materia y el vacío considerando a las partículas elementales como irreducibles.

SM ha sido una teoría exitosa que provee una descripción excelente para gran parte de los fenómenos en física de partículas, sin embargo deja algunos fenómenos sin explicar (p. ej. la asimetría de bariones, la aceleración del Universo, etc.) y por ello no alcanza a ser una teoría completa de las interacciones fundamentales. A pesar de ello, SM es usada como una teoría base para construir modelos más exóticos que incorporan partículas hipotéticas, dimensiones extra, y simetrías más elaboradas.

En SM, se asume que los neutrinos tienen las siguientes propiedades:

- Los neutrinos no tienen masa.
- Hay sólo tres tipos (o “sabores”) de neutrinos: neutrino electrónico (ν_e), neutrino muónico (ν_μ) y neutrino tauónico (ν_τ), cada uno asociado a los tres sabores de leptones con carga: electrón (e), muón (μ) y tauón (τ).
- El número leptónico es conservado de forma separada para cada una de las tres familias leptónicas: (e, ν_e), (μ, ν_μ) y (τ, ν_τ).
- Los neutrinos y anti-neutrinos son distintos.
- Todos los neutrinos son “left-handed” (dirección de espín opuesta al momento lineal) y todos los antineutrinos son “right-handed” (dirección del espín coincide con el momento lineal).

Estas propiedades eran las conclusiones de 20 años de trabajo experimental desde el descubrimiento de los neutrinos. Estaban bien descritas por el modelo de “dos-componentes” de los neutrinos [48, 49, 50], en el que al asumir que los neutrinos no tienen masa, concluían que la helicidad de éstos es una propiedad permanente, y que no hay forma de que un neutrino creado como un tipo se cambie a otro tipo.

Dado que la helicidad de una partícula es una cantidad fija (la misma para todos los observadores) sólo si la partícula no tiene masa (debido a que viajará a la velocidad de la luz y no podrá ser superada), entonces las partículas masivas pueden verse como “right-handed” o “left-handed”, según el observador. Por lo tanto, si la masa de los neutrinos es diferente de cero, el modelo de dos componentes deja de ser válido y SM debe modificarse. Resulta que las otras propiedades del modelo original de neutrinos (la distinción de los tres sabores y la diferencia entre neutrinos y antineutrinos) también dependen de manera crítica de la falta de masa de los neutrinos.

2.2. Más allá del Modelo Estándar: Neutrinos masivos y oscilaciones

En 1957, B. Pontecorvo planteó la hipótesis de que si los neutrinos fueran masivos podrían existir procesos en donde el sabor de los neutrinos no es conservado, a esto es lo que se le conoce como oscilaciones de neutrinos. En 1968, R. Davis realizó el primer experimento para detectar neutrinos electrónicos producidos en el Sol (conocidos como neutrinos solares), en donde sus resultados mostraron que el flujo de neutrinos del Sol era sólo un tercio del flujo predicho por modelos teóricos del interior del Sol (modelo estándar del Sol, J. Bahcall), a esto se le conoce como el “problema de los neutrinos solares”. A dicho problema no se le dio mayor importancia, ya que se creía que podía ser un error en las mediciones de R. Davis o un error en el modelo de J. Bahcall. Sin embargo, en 1989 el grupo de Kamiokande realizó el segundo experimento para detectar neutrinos solares, y confirmó el déficit en el flujo de neutrinos encontrado por R. Davis. Poco tiempo después, los experimentos GALLEX y SAGE también encontraron los mismos resultados. Por lo que a mediados de los 90’s el problema de los neutrinos solares se convirtió en un problema importante.

Por otro lado, en 1985 las colaboraciones Kamiokande (Kamioka Nucleon Decay Experiment) e IMB (Irvine, Michigan, Brookhaven) detectaron neutrinos atmosféricos, los cuales son producidos cuando los rayos cósmicos impactan con la atmósfera de la Tierra, y consisten de neutrinos muónicos y electrónicos. De los experimentos encontraron que la tasa de los neutrinos muónicos entre los electrónicos era menor a la esperada. A esto se le conoce como la “anomalía de neutrinos atmosféricos”. Al principio se creía que el déficit estaba relacionado al tipo de detector usado, ya que sólo los detectores Cherenkov (Kamiokande e IMB) reportaban dicho déficit, mientras que los calorímetros de seguimiento (Fréjus, NUSEX) no. Sin embargo, en 1994, el calorímetro de seguimiento Soudan-2 también reportó la anomalía de neutrinos atmosféricos, por lo que dicha anomalía adquirió credibilidad.

Además de dar una explicación tanto al problema de neutrinos solares como a la anomalía de neutrinos atmosféricos, se debía encontrar una explicación a que el déficit era diferente para ambos problemas, es decir, los neutrinos solares tenían un déficit diferente a los neutrinos atmosféricos.

En 1998, el experimento Super-Kamiokande anunció la primera evidencia de la oscilación de neutrinos. El experimento mostró un déficit en la distribución del ángulo cenital de algunos de los neutrinos muónicos (atmosféricos), los cuales habían pasado por la Tierra antes de ser detectados. La probabilidad de interacción de dichos neutrinos con la Tierra es demasiado baja como para explicar que el déficit se debía a la absorción de éstos por la Tierra. Sin embargo, si los neutrinos oscilan de un sabor a otro, la probabilidad de medir un sabor particular para un neutrino varía periódicamente a medida que éste se propaga, por lo que el déficit de neutrinos muónicos quedaba explicado mediante la oscilación de neutrinos.

La idea fundamental detrás de las oscilaciones de neutrinos es el concepto de mecánica cuántica de estados mixtos. Para los neutrinos la oscilación se da a partir de una mezcla entre el sabor y los estados propios de masa, es decir, los tres estados de sabor

(ν_e, ν_μ y ν_τ) son cada uno una superposición diferente de tres estados de masa definidos (m_1, m_2 y m_3). Por lo que, cuando un neutrino es producido su estado de sabor se encuentra bien definido, pero a medida que este se propaga, los tres estados de masa de los que está formado avanzan a velocidades ligeramente diferentes debido a la diferencia en sus respectivas masas. Por lo tanto, después de que éste ha recorrido cierta distancia, las proporciones relativas de los diferentes estados de masa habrán cambiado a la configuración inicial, dando como resultado una mezcla diferente de estados de sabor. Entonces, cuando el neutrino vuelve a interactuar, hay una probabilidad de que éste interactúe como un neutrino de otro tipo, aunque también puede interactuar como un neutrino del mismo tipo al que fue generado (pues la fase mecánico-cuántica avanza de manera periódica), todo dependerá de la distancia recorrida.

El experimento realizado por Super-Kamiokande, sólo daba respuesta a la anomalía de los neutrinos atmosféricos, por lo que el problema de los neutrinos solares aún seguía sin ser explicado. Sin embargo, en 2002, la colaboración SNO (del inglés Sudbury Neutrino Observatory) anunció evidencia concluyente de la oscilación de neutrinos solares, quedando resuelto, así, el problema detectado por R. Davis: los neutrinos del Sol son transformados de neutrinos electrónicos a otros sabores (muónicos o tauónicos) mientras viajan hacia la Tierra.

Dicho lo anterior, los resultados obtenidos por la colaboración Super-Kamiokande y SNO establecieron fuertemente que los neutrinos son partículas con masa (pues sólo así pueden ocurrir las oscilaciones), lo cual es una evidencia concluyente de nueva física más allá del modelo estándar de partículas.

2.3. ¿Dirac o Majorana?

Casi todas las partículas tienen su propia antipartícula (o antimateria): una partícula con la misma masa pero con otras propiedades cuánticas opuestas, como la carga o el espín. Como se mencionó antes, el modelo estándar de partículas asume que los neutrinos tienen su propia antipartícula: los antineutrinos.

La idea de antipartícula surgió en 1928, cuando P. Dirac desarrolló la que ahora se conoce como ecuación de Dirac. De dicha ecuación se obtenía como consecuencia que las antipartículas debían existir. En 1932, C. Anderson descubrió por primera vez una antipartícula: el positrón, una partícula como el electrón pero con carga positiva.

Dirac predijo que, además de tener cargas opuestas, la antimateria debería tener otra característica opuesta: la quiralidad (propiedad fundamental intrínseca parecida a la helicidad). La ecuación de Dirac permite a los neutrinos y antineutrinos ser partículas diferentes, y, como resultado, son posibles cuatro tipos de neutrinos (debido a que éstos tienen masa): neutrinos con quiralidad tanto zurda como diestra, y antineutrinos también con ambas quiralidades. Sin embargo, hasta el día de hoy sólo se ha observado que los neutrinos son “left-handed” y los antineutrinos son “right-handed”, lo que lleva a preguntarse ¿En dónde están los neutrinos “right-handed” y los antineutrinos “left-handed”?

Por otro lado, en 1937, E. Majorana presentó otra teoría: los neutrinos son en realidad su propia antipartícula. La ecuación de Majorana describió como los neutrinos

con masa podrían convertirse en antineutrinos y luego nuevamente en neutrinos. Este comportamiento podría ser observado en un tipo muy raro de decaimiento radioactivo llamado “decaimiento beta doble”, en donde dos neutrones se convierten simultáneamente en protones liberando dos partículas beta (electrones) y dos antineutrinos electrónicos. A este proceso se le conoce como $2\nu\beta\beta$ y es completamente consistente con los neutrinos sin masa del SM. Sin embargo, si el neutrino es una partícula de Majorana, es posible una forma modificada de este decaimiento, en la cual la partícula emitida como un neutrino por una de las desintegraciones beta es absorbida, como un antineutrino, por la otra, produciendo un decaimiento beta doble sin neutrinos (conocido como $0\nu\beta\beta$). Este tipo de decaimiento ($0\nu\beta\beta$) viola la conservación del número leptónico, pues dos electrones son creados sin antileptones correspondientes. Este proceso sólo puede ocurrir si los neutrinos son su propia antipartícula.

Ahora bien, aún es posible que los neutrinos y antineutrinos sean partículas diferentes, y que existan dos estados de neutrino y antineutrino, como se establece en la ecuación de Dirac. Los dos estados faltantes (neutrino “right-handed” y antineutrino “left-handed”) podrían ser mucho más pesados, o podrían no interactuar a través de la fuerza débil, sino que sólo interactúan a través de la gravedad. Sin embargo, si se logra detectar evidencia del decaimiento beta doble sin neutrinos sería una señal de que los neutrinos son de Majorana (son su propia antipartícula).

Si resultará que los neutrinos son de Majorana, entonces esto podría ayudar a explicar la asimetría entre materia y antimateria en el Universo. Se sabe que el Universo esta formado de materia, sin embargo, la teoría del Bing Bang predice que el Universo temprano contenía la misma cantidad de materia y antimateria. Entonces, ¿Por qué hay materia en el Universo? Un posible mecanismo apunta a la existencia de neutrinos pesados que eran su propia antipartícula y, por lo tanto, podrían descomponerse en materia y antimateria. Si ocurre un segundo fenómeno, llamado violación de la carga y la paridad (es decir, si el neutrino favorece ligeramente en su descomposición la producción de materia sobre la de la antimateria, como ocurre en $0\nu\beta\beta$), entonces podría haber inyectado un exceso del primero sobre el segundo. Después de que toda la materia y la antimateria en el Universo fueran aniquiladas (con la excepción de este pequeño exceso), el resultado sería un Universo hecho solo de materia.

2.4. Jerarquía de los neutrinos

De los experimentos de oscilación de los neutrinos, se sabe no sólo que los neutrinos tienen masa, sino que cada estado de masa (m_1, m_2, m_3) tiene una masa diferente. Sin embargo, estos experimentos sólo son sensibles a la diferencia de masa cuadrada, Δm^2 .

Con la observación de cómo los neutrinos solares (electrónicos) se transforman en neutrinos muónicos y tauónicos, se pudo determinar que los estados m_1 (el que tiene la fracción más grande de neutrino electrónico) y m_2 tienen masas similares, siendo m_2 ligeramente más pesado que m_1 . Mientras que, de los experimentos de oscilación de neutrinos atmosféricos (muónicos) se sabe que las diferencias de las masas al cuadrado entre los estados de masa m_3 y los otros dos estados es mucho más grande que las diferencias entre los estados m_1 y m_2 . Sin embargo, en este caso no se ha podido

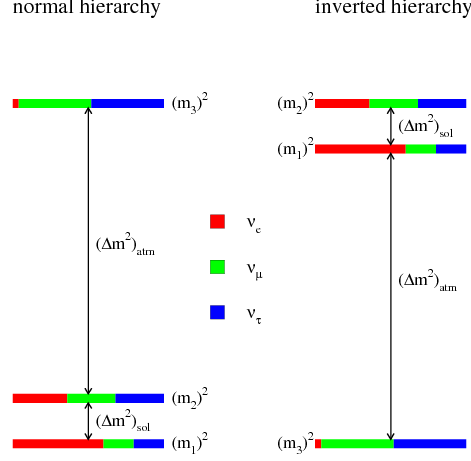


Figura 2.1: Representación pictórica de las posibles jerarquías de masas de los neutrinos.

determinar si m_3 es más pesado o más ligero que los otros dos. Para el caso en el que m_3 sea más masivo que m_1 , se le conoce como “jerarquía normal” (NH, por sus siglas en inglés) de los neutrinos ($m_3 \gg m_2 > m_1$), mientras que al caso contrario se le conoce como “jerarquía inversa” (IH, por sus siglas en inglés) de los neutrinos ($m_2 > m_1 \gg m_3$, ver figura 2.1).

Los valores actuales para la diferencia de masa cuadrada, obtenida de los experimentos de oscilación son los siguientes:

$$\Delta m_{21}^2 \equiv \Delta m_{\text{sol}}^2 = m_2^2 - m_1^2 \simeq 7.6 \times 10^{-5} \text{eV}^2 \quad (2.1)$$

$$|\Delta m_{31}^2| \equiv |\Delta m_{\text{atm}}^2| = |m_3^2 - m_1^2| \simeq 2.5 \times 10^{-3} \text{eV}^2 \quad (2.2)$$

Estos valores, imponen un límite inferior para la suma de las masas de los neutrinos, Σm_ν : $\Sigma m_\nu \equiv 0.06 \text{eV}$ para la jerarquía normal, y $\Sigma m_\nu \equiv 0.10 \text{eV}$ para la jerarquía inversa.

Ahora bien, el poder distinguir entre una jerarquía y otra tiene implicaciones en la comprensión del Universo, ya que está conectado con los esfuerzos para determinar si los neutrinos son de Majorana o de Dirac. De hecho, si resulta que la jerarquía es la normal entonces los neutrinos son de Majorana, por lo que serían su propia antipartícula y podrían llegar a explicar la relación de materia y antimateria observada en el Universo.

2.5. Asimetría Bariónica y Neutrinos

Como se discutió antes, la mayoría de las partículas tienen su propia antipartícula (o antimateria), incluidos los bariones. Ahora bien, para el Universo temprano, se asume de forma natural que los bariones y antibariones se produjeron en la misma cantidad, lo que conllevaría a un Universo sin materia, pues al interaccionar ambas partículas, éstas se aniquilan. Sin embargo, en el Universo sí hay materia, además la abundancia

de antibariones es mucho menor a la de los bariones, por lo que existe un desequilibrio notable entre ambas partículas, a lo cuál se le conoce como la “asimetría de bariones”.

En 1967, A. Sakharov propuso que el Universo es asimétrico debido a que el número bariónico, B , no se conserva. En este escenario, el Universo podía comenzar su expansión desde un estado simétrico, y conforme se va expandiendo, las reacciones físicas de partículas fuera de equilibrio térmico con no conservación tanto del número bariónico, B , como de las simetrías C y CP producirían un exceso de partículas sobre las antipartículas.

La simetría C (simetría de carga) es una transformación que cambia a todas las partículas con sus antipartículas correspondientes, y es también conocida como traslación de carga conjugada. Por lo que la no conservación de ésta es necesaria para explicar la asimetría bariónica, ya que de lo contrario las interacciones que producen más bariones que antibariones podrían ser compensadas por las interacciones que producen más antibariones.

Por otro lado, la simetría P (simetría de paridad) es una transformación que invierte el signo de una coordenada espacial, mientras que la simetría CP es una combinación de las simetrías C y P , y establece que las leyes de la física deberían ser las mismas si una partícula es intercambiada con su antipartícula mientras sus coordenadas espaciales son invertidas. La no conservación de la simetría CP es requerida para poder explicar la asimetría bariónica debido a que si dicha simetría se conservara, entonces la materia y antimateria (generadas en iguales cantidades) serían tratadas por la naturaleza de igual forma, y por tanto se hubieran aniquilado entre sí, dando como resultado un Universo sin materia.

Por lo anterior, la violación de la simetría CP es fundamental para poder explicar la asimetría de materia. Dicha violación ya ha sido observada experimentalmente en los quarks, sin embargo, la discrepancia no resulta suficiente para explicar la abundancia de materia en el Universo. Otro lugar donde podría observarse una violación CP es en el comportamiento de los neutrinos. Como ya mencione antes, se sabe que los neutrinos oscilan (cambian de sabor mientras viajan), y si dichas oscilaciones son fundamentalmente diferentes a las de los antineutrinos esto implicaría una violación en la simetría CP , y de ser así podrían llevarnos al entendimiento de la asimetría bariónica.

2.6. Importancia de los neutrinos en Cosmología

La teoría del Big Bang predice la generación de un fondo cósmico de neutrinos (neutrinos reliquia) en las primeras etapas de éste, cuyo desacople del resto de la materia ocurrió en los primeros segundos. Dado que los neutrinos tienen masa, los neutrinos reliquia son relativistas en el universo temprano, por lo que se comportan como radiación y contribuyen a la densidad de energía de radiación. Sin embargo, conforme el universo se expande, se vuelven no relativistas, por lo que se comportan como materia y contribuyen a la densidad total de materia. Por lo tanto, los neutrinos reliquia dejan una huella característica en los observables cosmológicos, como discutiré más adelante, alterando tanto la evolución de fondo y el espectro de perturbaciones de materia como a las anisotropías del CMB (del inglés Cosmic Microwave Background).

Por lo anterior, resulta muy importante tomar en cuenta la contribución de una componente de neutrinos masivos en los modelos cosmológicos, ya que de no considerarlos podría sesgar a la necesidad de un nuevo tipo de energía o gravedad.

Por otro lado, dado que los neutrinos tienen efectos sobre observables cosmológicos, la cosmología de precisión puede ser usada, de forma complementaria, para medir algunas de las propiedades de los neutrinos que resultan difíciles de medir en experimentos de laboratorio, como lo son la suma total de la masa de los neutrinos, Σm_ν , su número efectivo de especies y, muy posiblemente, sobre la jerarquía de sus estados propios de masa; que como se discutió anteriormente, tienen implicaciones, a su vez, en el entendimiento de las primeras etapas del Universo (como la bariogénesis y leptogénesis).

2.7. El fondo cósmico de neutrinos

La existencia de un fondo cósmico de neutrinos (CNB, por sus siglas en inglés) es una predicción de la teoría del Big Bang, que hasta el momento no ha sido detectada, sin embargo, su presencia está indirectamente establecida por el preciso acuerdo entre las abundancias primordiales, de los elementos ligeros, observadas y predichas, así como por el CMB y otros observables cosmológicos.

Los neutrinos reliquia fueron producidos en el universo temprano a gran temperatura mediante interacciones débiles frecuentes. Conforme el Universo se fue enfriando (por su expansión) dichas interacciones comenzaron a ser menos frecuentes hasta que la tasa de interacción de éstas cayó por debajo de la tasa de expansión del Universo, ocasionando que los neutrinos se desacoplaran del resto del plasma a una temperatura de $T \approx 1 \text{ MeV}$ (estimada al igualar el valor promedio térmico de las interacciones débiles y la tasa de expansión del Universo).

Poco después del desacople de los neutrinos, el CNB juega dos papeles principales en la nucleosíntesis del Big Bang (BNN, por sus siglas en inglés, el periodo en el cual las abundancias primordiales de elementos ligeros son creadas). El primero es que, en esa época, los neutrinos contribuyen a la densidad de energía relativista del Universo (siempre que $m_\nu \ll T_\nu$, donde m_ν y T_ν son la masa y temperatura de neutrino, respectivamente), fijando de esta forma la tasa de expansión de éste. Por otro lado, BBN es sensible al sabor de los neutrinos debido a que los neutrinos electrónicos y antineutrinos juegan un papel principal en las interacciones débiles. Por lo anterior, BBN puede poner límites al número efectivo de neutrinos, N_{eff} .

2.8. Número efectivo de neutrinos y radiación extra

Como ya vimos, los neutrinos inicialmente contribuyen a la densidad de radiación, ρ_r del Universo, la cual puede ser parametrizada a través del número efectivo de neutrinos, N_{eff} :

$$\rho_r = \rho_\gamma + \rho_\nu = \left[1 + \frac{7}{8} \left(\frac{4}{11} \right)^{4/3} N_{\text{eff}} \right] \rho_\gamma \quad (2.3)$$

donde ρ_γ es la densidad de energía de los fotones.

Ahora bien, dado que la tasa de expansión del Universo fija las abundancias primordiales de los elementos ligeros como el helio-4, entonces N_{eff} puede ser estimado experimentalmente mediante observables cosmológicos.

Por otro lado, el valor teórico estimado, para N_{eff} , a partir del análisis de desacople de la componente de neutrinos es $N_{\text{eff}} \simeq 3.046$. Por lo que, cualquier desviación de este valor implicaría nuevas características no-estándar para los neutrinos, por ejemplo, un neutrino estéril, o bien, que hay una contribución adicional de una partícula reliquia relativista diferente, dando como consecuencia posible nueva física. Sin embargo, hasta el momento, el valor de N_{eff} observado se encuentra en acuerdo con el valor teórico estimado.

2.9. Efectos de los neutrinos sobre observables cosmológicos

El efecto físico de neutrinos masivos sobre la estructura a gran escala (LSS, por sus siglas en inglés) está determinado por el hecho de que éstos se comportan como partículas similares a la radiación en el universo temprano, y como partículas tipo materia en el universo tardío ([51, 52]). En términos generales, los neutrinos se enfrían por la expansión del universo, y la transición de ser gas ultra-relativista a ser gas frío ocurre alrededor de:

$$z_{\text{nr}} \sim 2.0 \times 10^3 \frac{m_\nu}{1 \text{ eV}} \quad (2.4)$$

Debido a que la densidad numérica de neutrinos es comparable a la de los fotones, contribuirán considerablemente a la densidad de energía relativista cuando sus energías son relativistas, por lo que influirán en la tasa de expansión del universo temprano, pudiendo retrasar la época de igualdad de la materia y radiación. Cuando los neutrinos se vuelven no-relativistas, contribuyen a la densidad total de materia, afectando el agrupamiento de ésta. El tamaño de dicho efecto dependerá de la contribución de los neutrinos a la densidad total de materia. Dada la alta densidad numérica de los neutrinos, una pequeña masa de neutrinos contribuye considerablemente a la densidad crítica, incluso la suma de las masas de neutrinos mínima ($\sim 0.06\text{eV}$) contribuye a una fracción de materia $f_\nu \equiv \Omega_\nu/\Omega_m$ de aproximadamente 0.4%.

2.9.1. Flujo libre de los Neutrinos

Dado que los neutrinos sólo pueden interactuar con otras partículas mediante fuerza débil o gravitacionalmente, entonces después del desacople térmico, los neutrinos constituyen un fluido no-colisional, donde las partículas individuales fluyen libremente con una velocidad característica que, en promedio, es la velocidad térmica ν_{th} . Después de la transición no-relativista y durante el dominio de materia, la longitud de “flujo libre” (FS, por sus siglas en inglés) decrece pero más lento que el factor de escala, lo que conlleva a que la longitud de FS comóvil decrezca, ocasionando que para neutrinos volviéndose no-relativistas durante el dominio de materia, el número de onda de FS

comóvil pase por un mínimo al tiempo de transición [53]. Este valor mínimo está dado por:

$$k_{\text{nr}} \sim 0.018\Omega_{\text{m}}^{1/2} (m/1\text{eV})^{1/2} h\text{Mpc}^{-1} \quad (2.5)$$

El efecto físico del FS es amortiguar las fluctuaciones de densidad de neutrinos a pequeñas escalas: los neutrinos no pueden confinarse en regiones menores a la longitud de FS, debido a que sus velocidades son mayores a las de la velocidad de escape de los pozos de potencial en esas escalas.

2.9.2. Efecto de los Neutrinos en el Fondo Cósmico de Microondas

Para neutrinos en el rango de masas de 10^{-3} eV a 1 eV, se tiene que en la época de igualdad entre radiación y materia, contribuían a la densidad de radiación, mientras que en la época actual contribuyen a la densidad de materia (ver ec. 2.4). La densidad total no-relativista ($\omega_{\text{m}} = \Omega_{\text{m}}h^2$) depende de la masa total de los neutrinos ($M_{\nu} = \Sigma_i m_i$), entonces, cuando M_{ν} varía debe ocurrir una variación en z_{eq} o en ω_{m} al día de hoy. Un desplazamiento en z_{eq} afecta la amplitud de los picos del Fondo Cósmico de Microondas (CMB, por sus siglas en inglés), por ejemplo, si z_{eq} es retrasado, entonces los neutrinos conducen a un efecto Sachs-Wolfe integrado temprano (EISW) enriquecido, lo que implica un aumento de amplitud en el primer pico acústico. También, un desplazamiento de z_{eq} afecta la posición de los picos, por ejemplo, si z_{eq} es retrasado, entonces el tamaño del horizonte acústico comóvil a la época de desacople se reduciría, ocasionando que el tamaño angular del horizonte acústico en esa época también se redujera y por ende ocasionaría un desplazamiento de los picos hacia multipolos más grandes.

Por otro lado, un cambio en la densidad de materia no-relativista a tiempos tardíos podría ocasionar un cambio en la distancia del diámetro angular a la última superficie de dispersión, controlando la posición general de las características del espectro del CMB en el espacio multipolar.

Ahora bien, estos efectos sobre el CMB también podrían ser ocasionados al variarse otros parámetros cosmológicos diferentes a M_{ν} , o incluso podrían llegar a cancelarse si se varían simultáneamente con M_{ν} . Sin embargo, si además del CMB se combinan otros datos de observables cosmológicos diferentes (p. ej. el espectro de potencias de materia, la escala de BAO, mediciones directas de H_0), entonces se puede romper la degeneración entre los parámetros y por tanto se pueden obtener límites para M_{ν} .

2.9.3. Efecto de los Neutrinos en el Espectro de potencias de Materia

A escalas grandes (es decir, $k < k_{\text{nr}}$), el FS de neutrinos puede ser ignorado, debido a que las velocidades de neutrinos pueden ser consideradas despreciables, por lo que las perturbaciones de neutrinos son indistinguibles de las perturbaciones de CDM. En estas escalas, el espectro de potencias de materia $P(k, z)$ (MPS, por sus siglas en inglés) sólo depende de la fracción de densidad de materia hoy, Ω_{m} , y del espectro de

perturbaciones primordial. Sin embargo, a escalas pequeñas (es decir, $k > k_{\text{nr}}$), el MPS es afectado por las masas de neutrinos, por esencialmente tres razones [53]:

- **Los neutrinos masivos no se aglomeran en esas escalas.** En escalas de interés y en el universo reciente, las fluctuaciones de bariones, δ_b , y de CDM, δ_{cdm} , son iguales entre sí, mientras que las fluctuaciones de neutrinos, $\delta_\nu \ll \delta_{\text{cdm}}$. Por lo que el espectro de potencias estará reducido por un factor de $(1 - f_\nu)^2$.
- El corrimiento al rojo de la época de igualdad entre radiación y materia, z_{eq} o la relación barión-CDM, $\omega_b/\omega_{\text{cdm}}$, podrían verse ligeramente afectados, con un impacto en el espectro de potencias a pequeñas escalas. Las masas de los neutrinos pueden tener efectos indirectos adicionales a través del cambio en la evolución del fondo, dependiendo de qué parámetros se mantienen fijos cuando se varía Σm_ν .
- **La tasa de crecimiento de las perturbaciones de CDM es reducido** en la ausencia de efectos de reacción inversa (o “back-reaction”) gravitacional de los FS neutrinos. En un universo en el que todas las especies presentes en la ecuación de Friedmann se agrupan, como es el caso en un universo dominado por materia (sólo bariones y CDM), el factor de crecimiento lineal es proporcional al factor de escala, $\delta_{\text{cdm}} \propto a$. Sin embargo, cuando alguna de las especies que contribuye a la expansión de fondo no se agrupa eficientemente, puede ser despreciada en la ecuación de Poisson, lo que ocasiona que el agrupamiento gravitacional se vuelve menor que la fricción de Hubble, provocando que CDM (al igual que los bariones) se agrupen a una tasa más lenta. Por lo anterior, para $k \gg k_{\text{nr}}$ la tasa de crecimiento lineal durante el dominio de materia es igual a $a^{1-3f_\nu/5}$, en presencia de neutrinos [54].

En resumen, el espectro de potencias de materia (MPS) tiene una supresión en escalas pequeñas ($k \geq k_{\text{nr}}$) por la presencia de neutrinos masivos (ver figura 2.2) debido a la ausencia de perturbaciones de neutrinos en el MPS total y por una tasa de crecimiento de perturbaciones de CDM/bariones más lenta a tiempos tardíos. A un corrimiento al rojo bajo, $z \simeq 0$, la supresión del MPS comienza en $k \geq k_{\text{nr}}$ y se satura a $k \sim 1h/\text{Mpc}$ con una amplitud constante $\Delta P(k)/P(k) \simeq -8f_\nu$ [55, 56, 54].

2.10. Restricciones actuales sobre Σm_ν

Como acabamos de ver, los neutrinos masivos afectan la formación de estructura en el Universo, dejando una huella característica en varios observables cosmológicos, por lo que es posible restringir su masa usando una combinación de datos de estos observables.

Existen diferentes grupos que han puesto restricciones a la suma de masas de neutrinos, Σm_ν , usando diferentes conjuntos de datos. En la tabla 2.1 se muestran los resultados de algunos de ellos. De esta tabla se puede observar que los límites (superiores) actuales para Σm_ν están entre 0.11 eV y 0.20 eV a 2σ para un modelo $\nu\Lambda\text{CDM}$. Sin embargo, estos límites se degradan si, aparte de Σm_ν , se permite una variación de

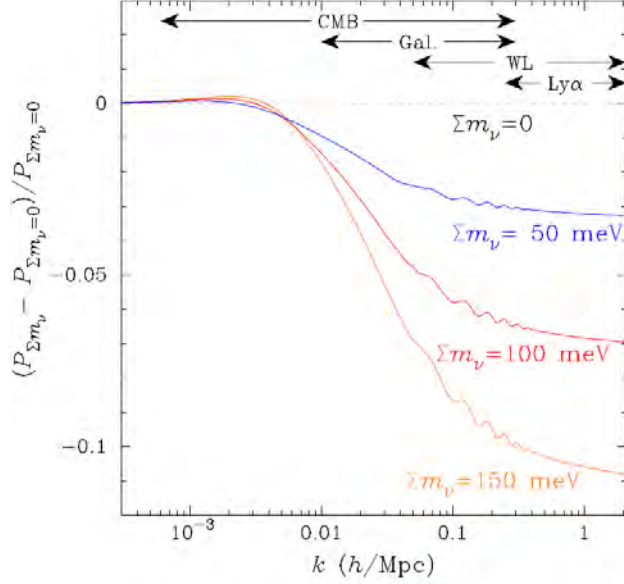


Figura 2.2: Supresión del espectro de potencias de materia (MPS) como función de Σm_ν

la ecuación de estado de energía oscura, w (modelo $w_0 w_a \nu$ CDM), debido a que hay una degeneración entre ambos parámetros.

Es importante recordar que de los experimentos de oscilación, en física de partículas, se tiene que el límite inferior para Σm_ν es 0.06 eV para la jerarquía NH y 0.10 eV para la jerarquía IH, por lo que con los límites superiores actuales estamos más cerca de distinguir el tipo de jerarquía que tienen los neutrinos.

2.11. Pronósticos para Σm_ν

Los recientes avances en la cosmología observacional han dado lugar a estrictas restricciones para la suma de las masas de neutrinos, y se espera que los futuros sondeos, como DESI, mejoren estos resultados, gracias a conjuntos de datos más grandes. Con ayuda de Planck y posiblemente sondeos de lensing, estos nuevos sondeos podrían mejorar las restricciones actuales sobre Σm_ν en ~ 0.01 o ~ 0.02 eV en los próximos años [19]. Esto dará una fuerte detección de la suma mínima posible de masas de ~ 0.06 eV, sin embargo, la jerarquía de masas sólo será distinguible con suerte, si ocurre que la verdadera suma de masas está justo en el mínimo.

En la tabla 2.3 se muestran las constricciones considerando el modelo cosmológico $\nu\Lambda$ CDM, obtenidas mediante proyecciones de la matriz de Fisher para actuales y futuros sondeos [19]. En dicha tabla se observa que DESI podrá medir la suma de masas de los neutrinos a 0.024 eV para $k_{max,eff} = 0.1 h\text{Mpc}^{-1}$ o 0.017 eV para $k_{max,eff} = 0.2 h\text{Mpc}^{-1}$. Se espera que para el final del sondeo base de DESI la masa mínima de neutrinos de 0.057 eV debería de detectarse a $\sim 3\sigma$.

De la tabla 2.3 también vemos que usando las tres diferentes técnicas independien-

Referencia	Conjunto de datos	Cosmología	
		$\nu\Lambda\text{CDM}$	$w_0w_a\nu\text{CDM}$
[57]	BOSS DR9 CMASS P(k)+ WMAP7 + SNLS3	<0.340 eV	<0.618 eV
[58]	BOSS DR11 + 6dFGS + WiggleZ + Planck	<0.151 eV	
[59]	BOSS DR11CMASS P(k) + PLANCK + JLA	<0.200 eV	<0.770 eV
[60]	BOSS DR9 Ly α + Planck	<0.110 eV	
[61]	BOSS DR12 (BAO+RSD) + eBOSS DR16 (BAO+RSD)	<0.0934 eV	

Cuadro 2.1: Restricciones actuales sobre Σm_ν , usando diferentes conjuntos de observables cosmológicos.

Abbreviation Data Set						
<i>P</i>	Planck CMB (and a 5% constraint on H_0 that only matters in severely under-constrained cases).					
<i>BgB</i>	BOSS galaxy BAO.					
<i>BIB</i>	BOSS Ly α forest and high- z quasar BAO.					
<i>BgAk_{max,eff}</i>	BOSS galaxy broadband to $k < k_{\text{max,eff}}$ $h\text{Mpc}^{-1}$ (plus BAO beyond that).					
<i>DES</i>	DES lensing and galaxy clustering.					
<i>BBgB</i>	DESI galaxy BAO.					
<i>BBIB</i>	DESI Ly α forest and high- z quasar BAO.					
<i>BBAk_{max,eff}</i>	DESI galaxy broadband to $k < k_{\text{max,eff}}$ $h\text{Mpc}^{-1}$ (plus BAO beyond that).					
<i>euAk_{max,eff}</i>	Euclid galaxy broadband to $k < k_{\text{max,eff}}$ $h\text{Mpc}^{-1}$ (plus BAO beyond that).					
<i>LSST</i>	LSST lensing and galaxy clustering.					
<i>l1D</i>	~ 100 high resolution Ly α forest spectra.					
<i>BBIA</i>	DESI Ly α forest broadband (including relatively small, $\sim 1D$ scales).					
<i>BB24</i>	24 is appended to BB to indicate 24000 sq. deg. DESI instead of the baseline 14000 sq. deg.					
value	ω_m	ω_b	θ_s	Σm_ν	$\log_{10}(A)$	n_s
<i>P</i>	0.141	0.0221	0.597	0.0600	-8.66	0.961
<i>P + BgB + BIB</i>	0.0037	0.00015	0.00035	0.35	0.0039	0.0038
<i>P + BgA0.1 + BIB</i>	0.00074	0.00015	0.00014	0.10	0.0038	0.0038
<i>P + BgA0.2 + BIB</i>	0.00070	0.00013	0.00014	0.068	0.0037	0.0031
<i>P + BgA0.2 + BIB</i>	0.00071	0.00012	0.00015	0.046	0.0037	0.0028
<i>P + DES</i>	0.0013	0.00013	0.00017	0.041	0.0036	0.0032
<i>P + BBgA0.1 + BBIB</i>	0.00044	0.00011	0.00014	0.024	0.0036	0.0024
<i>P + BBgA0.2 + BBIB</i>	0.00042	0.00010	0.00014	0.017	0.0035	0.0022
<i>P + LSST</i>	0.00080	0.00011	0.00015	0.020	0.0030	0.0029
<i>P + BBgA0.1 + BBIB + LSST</i>	0.00042	0.00010	0.00013	0.015	0.0028	0.0021
<i>P + BBgA0.2 + BBIB + LSST</i>	0.00041	0.00010	0.00013	0.014	0.0026	0.0020
<i>P + BB24gA0.2 + BB24IA + l1D + euA0.2 + LSST</i>	0.00032	9.5e - 05	0.00013	0.011	0.0024	0.0014

Figura 2.3: Pronóstico para los parámetros del modelo $\nu\Lambda\text{CDM}$ para distintos conjuntos de actuales y futuros sondeos [19]. Fueron obtenidos a partir de las proyecciones de la matriz de Fisher.

tes: Planck + RSDs (DESI), Planck + weak gravitational lensing (LSST), y Planck + CMB S4 + BAO, se obtendrán errores mejores que 0.02 eV, lo cual se traduce a detecciones con $\sim 3\sigma$ para $\Sigma m_\nu = 0.06$ eV, y para el modelo $\nu\Lambda$ CDM. Más aún, si se usan prácticamente todos los sondeos (Planck+DESI+Ly α +Euclid+LSST), ¡Se podrán tener detecciones con $\sim 5\sigma$!

Aunque no hay que perder de vista que los neutrinos tienen una degeneración con la ecuación de estado de DE, w , sin embargo, la combinación de restricciones en el crecimiento de estructura y en la evolución del fondo pueden ayudar a romper la degeneración entre DE y masa de neutrinos. Asimismo, permitir desviaciones de relatividad general (GR) degrada significativamente las restricciones de la masa de neutrinos.

Capítulo 3

Simulaciones Cosmológicas de N -cuerpos

El presente capítulo analiza el papel de las simulaciones de N -cuerpos en la cosmología contemporánea, así como cuantifica algunas de las características necesarias para auxiliar el diseño y la interpretación de los censos de galaxias en desarrollo. Se presenta una breve revisión de las técnicas que utilizan diferentes códigos, con énfasis en aquellos como la técnica híbrida P3M (Partícula-Partícula + Partícula-Malla) en la que la parte de menor resolución se aborda con algoritmos de malla como PM (Partícula-Malla) y la de más alta resolución se aborda con algoritmos de suma directa (Partícula-Partícula) como los códigos contemporáneos HACC y SWIFT. Se presenta un nuevo código de suma directa (desarrollado como parte del proyecto doctoral) con gran flexibilidad para el esquema de integración y el cual es capaz de aprovechar diferentes tipos de arquitecturas, y representa uno de los productos de esta tesis. Este código puede eventualmente combinarse con los otros desarrollados en la tesis o con alguno ya existente para aumentar el rango dinámico de las simulaciones.

3.1. Relevancia de los códigos de N -cuerpos en el entendimiento del Universo a Gran Escala

Durante los últimos 40 años se ha gestado un importante desarrollo en la disponibilidad y utilidad de las técnicas y recursos computacionales. Esto ha llevado a un aumento igualmente impresionante en la calidad y la precisión de las simulaciones de la formación de la estructura cósmica principalmente utilizando la aproximación de N -cuerpos, desde trabajos pioneros con apenas ~ 1000 partículas hasta los esfuerzos contemporáneos con 9 órdenes de magnitud de incremento en el número de partículas [28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40]. Las llamadas simulaciones de N -cuerpos se han constituido como un pilar importante de nuestra comprensión del modelo estándar cosmológico. Las predicciones de los modelos numéricos son ahora esenciales para casi todos los aspectos de los estudios de precisión de la materia y la energía oscura, debido a la no-linealidad intrínseca de la evolución gravitacional del acumulamiento de la materia en el Universo. De esta manera las simulaciones propor-

cionan un eslabón esencial en la cadena que une la física de partículas a la cosmología, y de manera similar entre los primeros segundos del universo a su estado actual.

3.1.1. Características de las simulaciones cosmológicas en la era de los censos contemporáneos de galaxias

Es un requisito que las simulaciones cosmológicas a utilizarse para interpretar los sondeos de galaxias en desarrollo cubran grandes volúmenes equiparables a lo observado (p. ej. los volúmenes de BOSS y DESI son, respectivamente, $\sim 5 \text{ Gpc}^3$, $\sim 50 [\text{Gpc } h^{-1}]^3$), con el fin de cuantificar la significancia estadística de las estructuras y calcular covarianzas entre observables, lo cual requiere de múltiples realizaciones de tales volúmenes grandes. Dados estos volúmenes el número de partículas se dispara con el fin de amortiguar las fluctuaciones estadísticas en la evolución y en las propiedades finales de la estructura ocasionadas por el número de partículas. En las escalas más grandes, el universo está poblado por cúmulos de galaxias, conectados por filamentos, circundados por vacíos cósmicos. Las propiedades estadísticas de estas estructuras se pueden utilizar en una variedad de métodos para restringir parámetros cosmológicos fundamentales.

Como ilustración del uso de las simulaciones podemos describir los siguientes casos. Por ejemplo, el número de objetos en el Universo de una masa dada, la función de masa, en particular para los cúmulos de galaxias más masivos o las galaxias mismas, es sensible a parámetros cosmológicos tales como la densidad de la materia, Ω_m , el espectro de potencia inicial de fluctuaciones en densidad, la naturaleza de la materia oscura y la ecuación de estado de la energía oscura. El modelado de la función de masa o de velocidad circular máxima es un enorme desafío para las simulaciones numéricas, y los errores tanto estadísticos como sistemáticos conspiran con la dinámica no-lineal para dificultar el cálculo de una predicción teórica confiable a precisiones menores al 1%, que pueda contrastarse contra los catálogos de galaxias en desarrollo o las imágenes de lente gravitacional fuerte [62]. Lograr lo anterior implicará, por ejemplo, generar condiciones iniciales yendo más allá de la aproximación de Zeldovich, utilizando teoría perturbativa del crecimiento de fluctuaciones en el régimen cuasi-lineal [62] o iniciar a corrimientos al rojo muy grandes, así como controlar las fuentes de ruido en condiciones iniciales, durante la evolución y al medir los resultados. Por otro lado, para garantizar la significancia estadística y convergencia en el cálculo numérico de las simulaciones se requiere de múltiples simulaciones que abarquen un amplio rango dinámico (desde varias escalas de BAO hasta los trazadores observacionales). Sin embargo, esto no es una tarea simple ya que incluso una sola simulación en el estado del arte, por ejemplo Outer Rim [63] ($\sim 1.07 \times 10^{12}$ partículas en un volumen de $\sim (3 h^{-1} \text{Gpc})^3$), resulta muy cara computacionalmente y no cuenta por sí sola con suficiente poder estadístico (sólo es una realización). Por ello, otra avenida es construir un conjunto de simulaciones en diferentes resoluciones espaciales, que busquen cubrir el amplio rango dinámico de forma precisa.

Las llamadas oscilaciones acústicas bariónicas (BAO, por sus siglas en inglés) detectadas en el Fondo Cósmico de Microondas (CMB, por sus siglas en inglés) [64, 65, 23] se han medido posteriormente en la distribución espacial de la estructura a gran escala

utilizando los sondeos de galaxias. En específico, la señal de BAO se ha detectado en el SDSS-IV (del inglés Sloan Digital Sky Survey IV) con una significancia de $\sim 8.4\sigma$ [66] dentro de la doceava liberación de datos (DR12, por sus siglas en inglés) correspondiente al sondeo de galaxias BOSS (del inglés Baryon Oscillation Spectroscopic Study) [67]. Estudios posteriores como el sondeo de galaxias DESI han comenzado a realizar observaciones (se han concluido al menos dos campañas de validación del sondeo) que en el futuro cercano se espera que alcanzarán precisiones menores al uno por ciento [23], así como con los estudios futuros de baja frecuencia en radio [68, 69]. En resumen, las observaciones cosmológicas han llegado a un nivel de precisión tal que las simulaciones necesarias para su interpretación requieren de enormes volúmenes ($\gtrsim 1$ Gpc por lado) y miles de millones de partículas (p. ej. la simulación Outer Rim), por lo que para garantizar la significancia estadística y por lo mismo para realizarse, se requiere de métodos de cálculo e infraestructura con una alta eficiencia en el cómputo. [70] ha revisado recientemente el estado de las simulaciones numéricas de grandes volúmenes cosmológicos con solo materia–energía oscura en el contexto Λ CDM, la más grande es la “DEUS FUR” incluyendo 0.55×10^{12} partículas [40] realizado con el código de malla adaptiva RAMSES [71]. Otras simulaciones/códigos capaces de manejar números de partículas en la escala de 10^{11} son HR3 [72] utilizando GOTPM [73], Millennium-XXL [39] con GADGET-3 [36], Jubilee [74] con CUBEP3M [75], y Bolshoi/BolshoiP [38] con ART [76], así como HACC [41] y GreeM [77, 78]. Los códigos ChaNGa [79, 80]/PKDGRAV3, SWIFT¹ y ABACUS [81] tiene capacidades similares, todos son multi-arquitectura y multi-técnica de N -cuerpos. Como ejemplo, recientemente la colaboración DESI comenzó un estudio de la robustez de las predicciones de las simulaciones de N -cuerpos a cambios en el esquema de condiciones iniciales, resolución y algoritmos utilizando los códigos ABACUS, SWIFT, GADGET-2, y FASTPM [82] mostrando convergencia en el espectro de potencia de materia, de halos, la función de correlación así como la función de masa de halos y la cinemática de los halos. Tal estudio es un gran precedente para la interpretación robusta de las futuras observaciones cosmológicas. Las técnicas computacionales han avanzado tan rápidamente que el tiempo para ejecutar una simulación en el estado del arte siguiendo 13.8 mil millones de años de historia cósmica toma sólo días en arquitecturas de supercomputadoras modernas utilizando decenas o centenas de miles de núcleos y tarjetas GPU’s como Summit en Oak Ridge, EUA o Marenostrum en el BSCC en Barcelona, España. En México se comienza a tener infraestructuras que permitirían realizar cálculos comparables a un orden de magnitud menor.

Las simulaciones mencionadas tienen como propósito explorar la física del modelo en el que se enmarcan y sus consecuencias para la distribución de materia y halos (galaxias). Con el crecimiento de los censos observacionales del cielo, el cuantificar la representatividad estadística de las simulaciones es un requisito para su utilidad en la inferencia de parámetros cosmológicos. Como ejemplo, a pesar de que en [83] usaron 1000 realizaciones de EZ-mocks [84] (usa la aproximación de Zeldovich [85] para calcular el campo de densidad) encontraron un error sistemático en la escala de BAO, al comparar con la simulación Outer Rim, y no saben si es por la naturaleza aproximada

¹www.swiftsim.com

de EZ-mocks o un error en su metodología al estimar la escala de BAO. Es imposible tener ensambles suficientemente grandes de simulaciones como las mencionadas, por lo mismo ha surgido un gran número de técnicas para realizar simulaciones “rápidas”, algunas utilizan directamente métodos perturbativos calibrados con simulaciones completas (p. ej. HALOFIT [86, 87, 88]), otras combinan métodos de N -cuerpos con métodos perturbativos (p. ej. EZ-mocks: Zeldovich+halos PDF’s, MG-PICOLA [89, 90], GADGET3+kspace neutrinos), algunos otros utilizan degradaciones en la resolución y/o paso de integración y, posteriormente, corrigen para buscar que cantidades estadísticas de interés como el espectro de potencias o la función de masa de halos (FastPM [91]) sean consistentes con simulaciones costosas completas. Finalmente, técnicas de Machine Learning se han utilizado con el fin de ser entrenadas utilizando simulaciones completas y permitir generar decenas o miles de realizaciones a un costo computacional moderado. Cabe mencionar que esta última generación de simulaciones ha permitido el estudio de predicciones cuantitativas aplicadas a los censos observacionales.

3.2. Un nuevo Código Paralelo de suma directa con paso de integración adaptivo y reversible: $\text{NP}_{\text{splitt}}$

A continuación introducimos a $\text{NP}_{\text{Splitt}}$, un nuevo código de N -cuerpos de suma directa en el cual implementé los integradores Leapfrog, AKDK, el integrador colisional Hermite de 4o. orden y el integrador de división hamiltoniana jerárquica (HHS, por sus siglas en inglés). El código está además implementado en diferentes arquitecturas computacionales contemporáneas.

3.2.1. Introducción

Históricamente, las simulaciones astrofísicas realistas de N -cuerpos totalmente auto-consistentes son un problema desafiante [92, 93]. En el caso puramente gravitacional, el cuello de botella de los códigos de N -cuerpos de suma directa está dado por la cantidad de trabajo computacional necesario para evolucionar con precisión las órbitas de las partículas. Dichas limitaciones desencadenaron el desarrollo de sofisticados métodos híbridos aproximados, como los códigos TreePM (en inglés, Tree Particle-Mesh)/P3M (del inglés Particle-Particle Particle-Mesh) [94, 95, 96, 97] donde los códigos de suma directa pueden manejar el régimen de alta resolución, y los códigos AMR (del inglés Adaptive Mesh Refinement) [98, 99, 76], apelando a un equilibrio entre precisión y eficiencia computacional. Las simulaciones de N -cuerpos requieren tanto una forma rápida de calcular las fuerzas como un método de integración preciso para hacer evolucionar partículas en el tiempo. Típicamente, el integrador simpléctico Leapfrog de segundo orden es el más utilizado en simulaciones de N -cuerpos. Este integrador es estrictamente simétrico cuando se adopta un paso de tiempo común global; sin embargo, esto no es eficiente para los códigos híbridos modernos. En la búsqueda de la eficiencia computacional, es necesario adoptar pasos de tiempo múltiples o adaptativos, lo que implica un esquema de Leapfrog modificado y aproximado. En los últimos años, la búsqueda de un integrador simpléctico derivó en varias propuestas. [100] propuso

esquemas de bloque de pasos de tiempo y recientemente [101] propuso un esquema de segundo orden de división hamiltoniana jerárquica (HHS, por sus siglas en inglés), que ofrece un compromiso interesante entre precisión en energía y, principalmente, en conservación del momento y la eficiencia computacional. Al igual que Leapfrog, HHS es verdaderamente simpléctico sólo si las partículas comparten un paso de tiempo global y constante, pero, como mostraremos más adelante, incluso con un esquema de pasos de tiempo individual para las partículas las propiedades de conservación son bastante notables, a diferencia de Leapfrog. En la última década, los códigos de N -cuerpos se han beneficiado enormemente del poder de cómputo de las supercomputadoras modernas que explotan dispositivos como CPUs multinúcleo y GPGPUs [102, 103, 104, 105]; últimamente, la posibilidad de nuevas alternativas como procesadores ARM está captando interés. En las siguientes subsecciones presentamos implementaciones en CPU, GPU y CPU-GPU del integrador HHS, y como referencia implementamos los conocidos integradores Leapfrog y Hermite en los mismos sistemas informáticos. Realizamos algunas exploraciones sobre la dependencia del rendimiento de la arquitectura de la computadora, por ejemplo, arquitecturas CISC (AMD Ryzen, CPU Epyc; Intel: Xeon); arquitecturas RISC (IBM Power9, NVIDIA Jetson Nano CPU: Quad-core ARM A57) y tarjetas NVIDIA GPU (Fermi C2050 y Tesla V100).

3.2.2. Integradores

$\text{NP}_{\text{splitt}}$ es un código de N -cuerpos partícula-partícula (también conocido como suma directa) que implementa cuatro integradores diferentes²: Leapfrog, Hermite de cuarto orden, KDK adaptivo (AKDK, por sus siglas en inglés) y el integrador de división hamiltoniana jerárquica (HHS, por sus siglas en inglés). En el siguiente capítulo, 4, se describirá en detalle cada uno de los integradores, a excepción del integrador Hermite, el cual será descrito en detalle en este capítulo. Para una descripción y discusión exhaustivas de los algoritmos utilizados en simulaciones astrofísicas de N -cuerpos, consulte [93].

Leapfrog

El integrador Leapfrog es un integrador simpléctico (únicamente con paso de tiempo constante) de segundo orden que es muy fácil de implementar y ampliamente utilizado. Los integradores simplécticos están diseñados para preservar numéricamente las integrales de movimiento y el volumen de espacio-fase del sistema a simular.

AKDK

El integrador KDK adaptivo, es similar a Leapfrog pero implementa pasos de tiempo adaptivos, por lo que es más eficiente que Leapfrog para sistemas con diferentes rangos

²Los integradores implementados en el código no son los únicos en la literatura, pero son los más usados (a excepción del HHS, que apenas se comienza a usar) debido a la relación precisión/costo-computacional. Para una discusión exhaustiva de integradores usados en simulaciones (astrofísicas) de N -cuerpos, consultar, por ejemplo, [93]

dinámicos, sin embargo, no es simpléctico. Este integrador es el más utilizado para sistemas no-colisionales.

División Hamiltoniana Jerárquica

El método de División Hamiltoniana Jerárquica [101] (HHS, por sus siglas en inglés) es un integrador de segundo orden que, a diferencia de Leapfrog, permite pasos de tiempo individuales para las partículas manteniendo propiedades de conservación bastante notables, como la energía y, principalmente, el momento. En el siguiente capítulo estudiaremos estas propiedades.

Este integrador consiste en dividir adaptativa y recursivamente el hamiltoniano con base en el paso de tiempo actual, dt , asignado a las partículas.

Hermite

El integrador de Hermite [106] es un esquema de cuarto orden que se usa ampliamente. Es muy utilizado para problemas colisionales porque considera información del “jerk” (derivada de la aceleración), sin embargo, también se considera en problemas no colisionales [107]. Este integrador es un esquema predictor-corrector que utiliza derivadas de alto orden para construir interpolaciones polinomiales de fuerzas y considera pasos de tiempo individuales.

En el esquema de Hermite con paso de tiempo individual, una partícula i tiene su propio tiempo (t_i), paso de tiempo (Δt_i), posición (\mathbf{x}_i), velocidad (\mathbf{v}_i), aceleración (\mathbf{a}_i) y jerk ($\dot{\mathbf{a}}_i$). Entonces, el esquema es el siguiente:

1. Seleccionar a todas las partículas i con un tiempo $t_i + \Delta t_i$ mínimo y establecer el tiempo global de integración $t = t_i + \Delta t_i$.
2. Predecir las posiciones y velocidades de todas las partículas en el momento t :

$$\begin{aligned}\mathbf{x}_{p,j} &= \mathbf{x}_j + \mathbf{v}_j (t - t_j) + \frac{\mathbf{a}_j}{2} (t - t_j)^2 + \frac{\dot{\mathbf{a}}_j}{6} (t - t_j)^3 \\ \mathbf{v}_{p,j} &= \mathbf{v}_j + \mathbf{a}_j (t - t_j) + \frac{\dot{\mathbf{a}}_j}{2} (t - t_j)^2\end{aligned}\tag{3.1}$$

3. Calcular aceleración (\mathbf{a}_i) y jerk ($\dot{\mathbf{a}}_i$) de las partículas i al tiempo t usando $\mathbf{x}_{p,j}$ y $\mathbf{v}_{p,j}$.
4. Corregir las posiciones y velocidades de las partículas i :

$$\begin{aligned}\mathbf{x}_i(t_i + \Delta t_i) &= \mathbf{x}_{p,i} + \frac{\mathbf{a}_{0,i}^{(2)}}{24} \Delta t_i^4 + \frac{\mathbf{a}_{0,i}^{(3)}}{120} \Delta t_i^5 \\ \mathbf{v}_i(t_i + \Delta t_i) &= \mathbf{v}_{p,i} + \frac{\mathbf{a}_{0,i}^{(2)}}{6} \Delta t_i^3 + \frac{\mathbf{a}_{0,i}^{(3)}}{24} \Delta t_i^4\end{aligned}\tag{3.2}$$

donde,

$$\begin{aligned}\mathbf{a}_{0,i}^{(2)} &= \frac{-6(\mathbf{a}_{0,i} - \mathbf{a}_{1,i}) - \Delta t_i(4\dot{\mathbf{a}}_{0,i} + 2\dot{\mathbf{a}}_{1,i})}{\Delta t_i^2} \\ \mathbf{a}_{0,i}^{(3)} &= \frac{12(\mathbf{a}_{0,i} - \mathbf{a}_{1,i}) + 6\Delta t_i(\dot{\mathbf{a}}_{0,i} + \dot{\mathbf{a}}_{1,i})}{\Delta t_i^3}\end{aligned}\tag{3.3}$$

De ahora en adelante, los valores con subíndices 0 y 1 son los valores en t_i y t , respectivamente.

5. Calcular los nuevos pasos de tiempo y actualizar t_i :

$$\Delta t_{c,i} = \sqrt{\eta \frac{|\mathbf{a}_{1,i}| |\mathbf{a}_{1,i}^{(2)}| + |\dot{\mathbf{a}}_{1,i}|^2}{|\dot{\mathbf{a}}_{1,i}| |\mathbf{a}_{1,i}^{(3)}| + |\mathbf{a}_{1,i}^{(2)}|^2}} \quad (3.4)$$

donde,

$$\mathbf{a}_{1,i}^{(2)} = \mathbf{a}_{0,i}^{(2)} + \mathbf{a}_{0,i}^{(3)} \Delta t_i \mathbf{a}_{1,i}^{(3)} = \mathbf{a}_{0,i}^{(3)} \quad (3.5)$$

Para sincronizar a las partículas en determinados tiempos, y con ello poder calcular algunas cantidades físicas del sistema, podemos usar el esquema de pasos de tiempo por bloques [108], en el cual los pasos de tiempo están “cuantizados” en números enteros de dos [109]:

$$\Delta t_{i,\text{new}} = \begin{cases} \frac{1}{2} \Delta t_i & \text{if } \Delta t_c < \Delta t_i, \\ \min(2\Delta t_i, \Delta t_{\text{max}}) & \text{if } \Delta t_c > \Delta t_i \text{ and } t_i \bmod 2\Delta t_i = 0, \\ \Delta t_i & \text{otherwise} \end{cases} \quad (3.6)$$

donde Δt_{max} es el máximo paso de tiempo.

6. Regresar al paso (1).

3.2.3. Implementación del código

En los códigos de suma directa de N -cuerpos, la evaluación de las fuerzas entre las N partículas del sistema escala como N^2 debido al cálculo de los elementos de la matriz de distancia euclidiana r_{ij} . Esto significa que el tiempo computacional requerido para integrar un sistema de N -cuerpos usando la suma directa, supera el poder de cada plataforma de CPU para valores grandes de N y/o para una integración larga del sistema. Por esta razón, es práctico utilizar la computación paralela. Dicho esto, NP_{splitt} está implementado en una GPU para acelerar las simulaciones de N -cuerpos, combinando herramientas de C y CUDA³ (del inglés Compute Unified Architecture, ver CUDA-zone).

Como se mencionó anteriormente, el código puede manejar cuatro integradores diferentes: Leapfrog, AKDK, Hermite de cuarto orden y división hamiltoniana jerárquica (HHS, por sus siglas en inglés). Para los integradores Leapfrog y Hermite, la integración temporal completa del problema de N -cuerpos se implementó en la GPU, para evitar el cuello de botella generado por las comunicaciones CPU-GPU. La figura 3.1

³CUDA es una plataforma de computación paralela y un modelo de programación desarrollado por NVIDIA para computación general en GPU. Con CUDA, los desarrolladores pueden expresar grandes cantidades de paralelismo a través de extensiones en forma de algunas palabras clave básicas y dirigir al compilador a la parte de la aplicación que se asigna a la GPU.

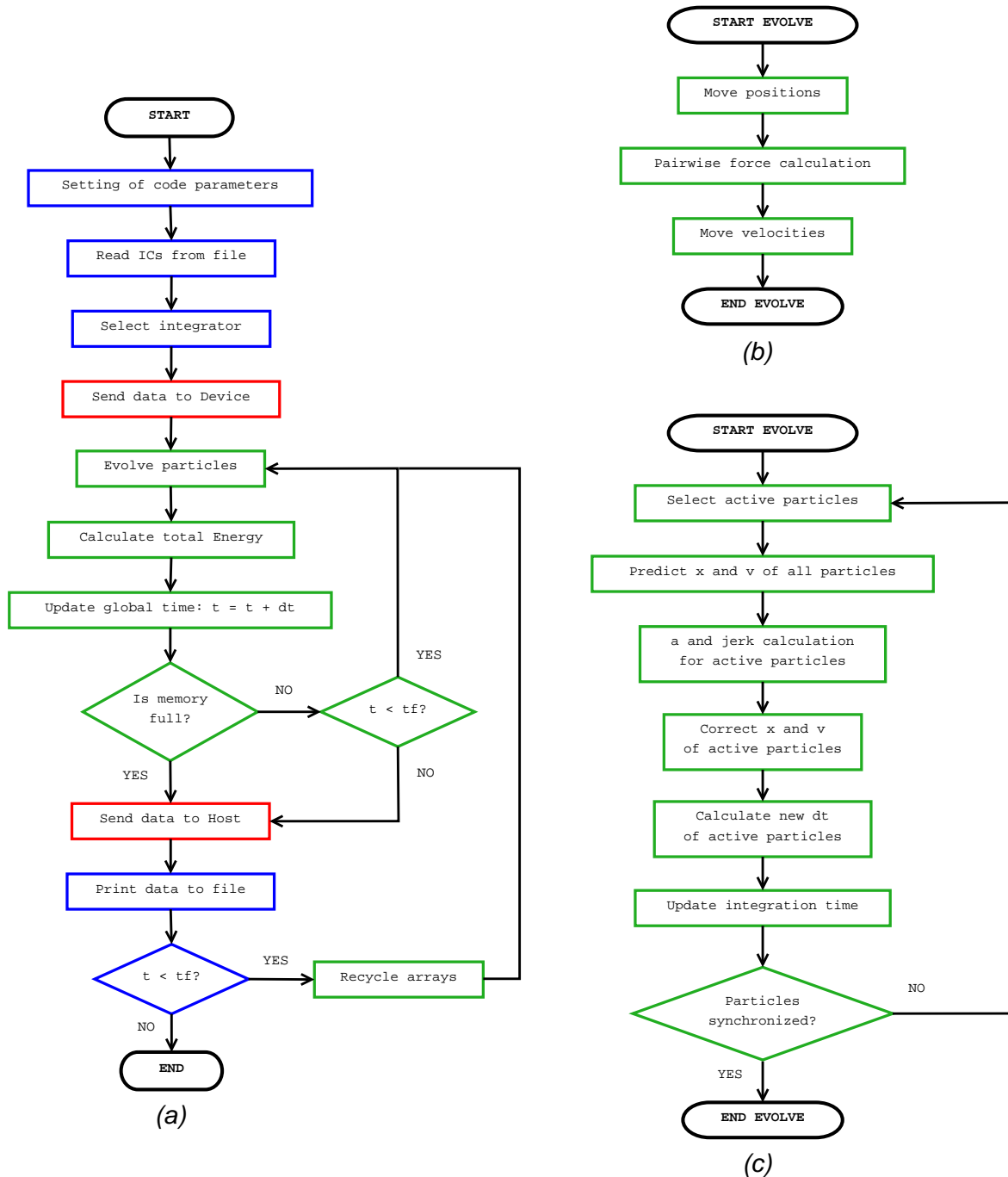


Figura 3.1: (a) Diagrama de flujo de la implementación básica del código en GPU. Diagrama de flujo de la rutina de evolución para los integradores (b) Leapfrog y (c) Hermite. Los recuadros azules representan la ejecución en la CPU, los recuadros verdes la ejecución en la GPU y los recuadros rojos la comunicación CPU-GPU.

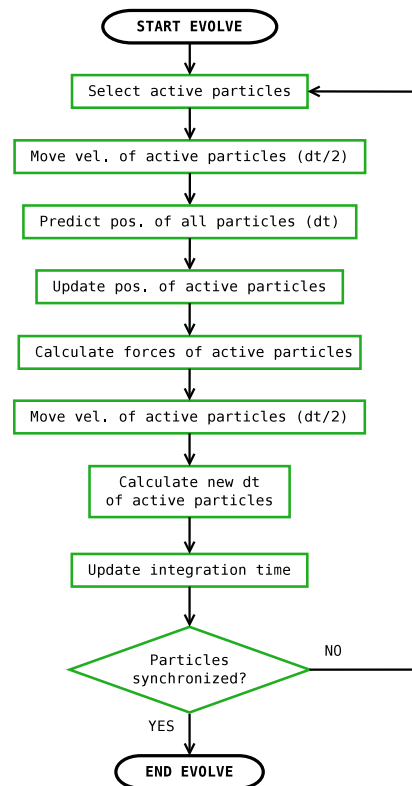


Figura 3.2: Diagrama de flujo del integrador AKDK. Todo el algoritmo se ejecuta en la GPU.

(a) muestra el diagrama de flujo de la implementación paralela básica del código. En esta figura, el color azul representa las partes del código que se ejecutan en la CPU, el color verde las partes ejecutadas en la GPU y el color rojo la comunicación CPU-GPU (en el resto del capítulo seguiremos esta convención de color para los diagramas de flujo). En la misma figura, se muestra el diagrama de flujo de la rutina de evolución para los integradores (b) Leapfrog y (c) Hermite.

Aparte de la implementación en la GPU, el algoritmo Leapfrog también se implementó utilizando las directivas OMP⁴ (del inglés Open Multi-Processing), para las rutinas *move positions*, *pairwise force calculation* y *move velocities* (ver figura 3.1 (b)).

La figura 3.2 muestra el diagrama de flujo de la rutina de evolución del integrador AKDK en la GPU. Al igual que los integradores Leapfrog y Hermite, la integración temporal completa del problema se implementa en la GPU. Notar que la implementación de AKDK requiere predecir/extrapolar las posiciones de las partículas no activas para calcular correctamente la aceleración de las activas. Al igual que Leapfrog, el algoritmo AKDK también tiene una implementación usando OMP.

El algoritmo de división hamiltoniana HHS consta de cuatro secciones principales (ver figura C.1 para el pseudocódigo): (1) cálculo de pasos de tiempo de las partículas (*calculate_timestep*), (2) dividir en sistemas S y F , (3) *drift* de las posiciones del sistema S y (4) *kick* de las velocidades del sistema S , que incluye los cálculos de fuerza por pares. Dicho esto, para el integrador HHS, tenemos diferentes versiones en la implementación del algoritmo. La primera es completamente serial (HHSv1), es decir, se ejecuta en un solo procesador (CPU). Para la segunda versión de la implementación (HHSv2), paralelizamos la primera, tercera y cuarta sección del algoritmo usando OMP.

En la tercera versión del programa (HHSv3), implementamos la paralelización en CUDA de las rutinas *calculate_timestep*, *drift* y *kick* (recuadros verdes en la fig. 3.3) mientras que las rutinas restantes las realiza una sola CPU (recuadros azules en la fig. 3.3). Dado que solo se implementan partes del algoritmo en la GPU, existe una comunicación constante entre CPU-GPU (recuadros rojos en 3.3), lo que puede ser un potencial cuello de botella de las aplicaciones de la GPU. Es importante aclarar que realizamos esta implementación con la configuración mencionada debido a que la recursividad intrínseca del método no puede ser manejada en la GPU (es decir, CUDA no admite algoritmos recursivos).

La cuarta versión (HHSv4) es muy similar a la tercera, pero en esta versión, hacemos uso de la componente de memoria unificada⁵ del modelo de programación CUDA (ver CUDA-c-programming-guide) que está disponible para la CUDA capability 6.0 o superior. Usamos memoria unificada en esta versión ya que la implementación del algoritmo se puede beneficiar de la tecnología NVIDIA NVLink⁶ (consulte NVIDIA-

⁴OMP es un modelo de múltiples subprocesos con direcciones de memoria compartida, que permite la programación paralela en C/C++ y Fortran. Consiste en un conjunto de directivas de compilador, rutinas de biblioteca y variables de entorno para especificar las regiones paralelas y controlar el comportamiento del código en tiempo de ejecución.

⁵La memoria unificada define un espacio de memoria administrado accesible desde cualquier procesador en un sistema, eliminando la necesidad de llamadas explícitas para copiar memoria.

⁶NVIDIA NVLink es una interfaz de interconexión de alto ancho de banda que facilita la comunicación de alta velocidad entre GPUs, y entre la CPU y GPU, ofreciendo una alternativa más rápida que el ancho de banda PCIe tradicional.

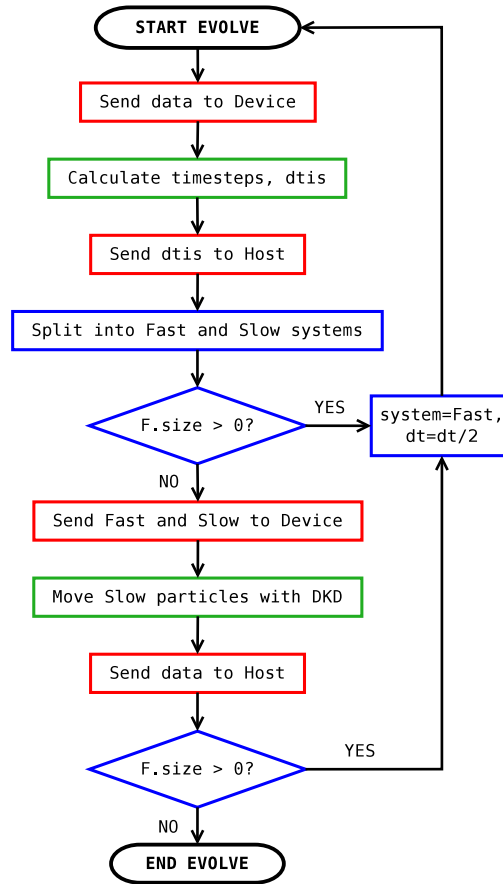


Figura 3.3: Diagrama de flujo del integrador HHS en su versión recursiva. Los colores de las cajas siguen la misma convención que la figura 3.1.

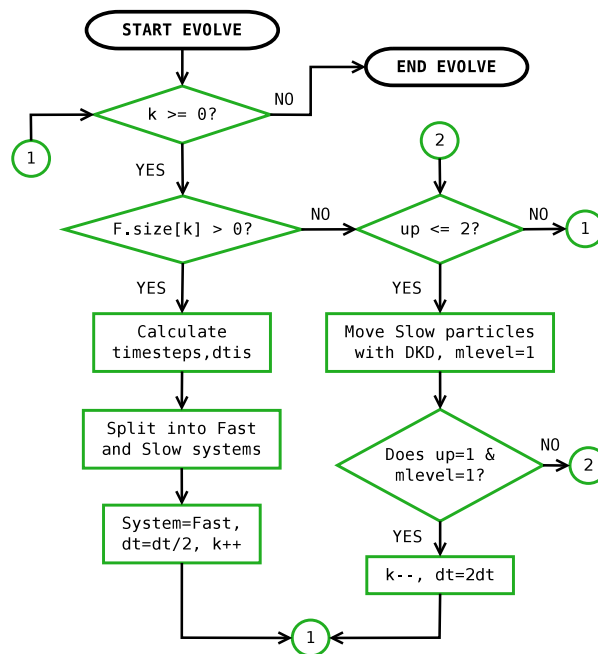


Figura 3.4: Diagrama de flujo de la versión iterativa del integrador HHS.

NVlink), que reduce el costo computacional de la comunicación CPU-GPU.

Buscando reducir el potencial costo computacional que representa el uso de funciones recursivas, y también, para evitar las constantes comunicaciones CPU-GPU, presentes en la tercera versión del código (HHSv3), realizamos una versión iterativa del algoritmo original (ver figura C.2 para el pseudocódigo). En esta versión, es necesario guardar los sistemas S y F de cada nivel de división, k , a diferencia de la versión recursiva. En la quinta versión del código de HHS (HHSv5), realizamos una implementación completamente en serie de la versión iterativa del algoritmo.

La sexta versión del código (HHSv6) es la implementación paralela CUDA del algoritmo iterativo (ver fig. 3.4 para el diagrama de flujo), que está completamente implementado en la GPU y, por lo tanto, evita la constante comunicación CPU-GPU, presente en la tercera versión del código (HHSv3). Finalmente, después de perfilar el código, implementamos una versión optimizada del integrador HHS, que incluye un código multi-GPU.

3.2.4. Pruebas de precisión del código: Binarias

La precisión de la integración de la ecuación de movimiento es crucial para una simulación de N -cuerpos confiable. Por lo anterior, en esta subsección, presentamos los resultados del código en términos de precisión y comparamos entre los integradores implementados, en el contexto de sistemas binarios. En el capítulo 4 de la tesis (siguiente capítulo), presentaremos los resultados de precisión del código en el contexto de un halo aislado, así como de una fusión menor de halos de galaxias.

La prueba más básica pero muy útil para códigos de N -cuerpos es el problema de 2-cuerpos, porque posee una solución analítica. Para explorar dicha prueba, simulamos binarias circulares y excéntricas ($e = 0.9$) para períodos orbitales de 2×10^6 y 10^5 , respectivamente, siguiendo [110] que mostró que un tiempo de integración prolongado restringe mejor la naturaleza del integrador.

Es importante aclarar que en este conjunto de pruebas no tenemos niveles de paso de tiempo (es decir, no hay una estructura de pasos de tiempo de las partículas), por lo tanto, es principalmente una prueba de paso de tiempo adaptivo (el paso de tiempo varía en el tiempo, pero es el mismo para las partículas). Decidimos comparar los integradores restringiéndolos a un límite de conservación de energía.

Para el caso de la binaria circular, elegimos un límite de conservación de energía del orden de 10^{-7} . Los paneles izquierdos de la figura 3.5 muestran, de arriba a abajo, el error fraccional de energía, el error del semieje mayor, el error en la excentricidad, y el error fraccional de la posición respecto a la solución analítica, para el sistema binario circular. Todas las cantidades muestran oscilaciones a lo largo de la integración temporal de la órbita. A excepción del error en la energía, el resto de las cantidades muestran que el límite inferior crece linealmente, lo cual está relacionado con la frecuencia de muestro de los datos calculados. Después de 500 períodos orbitales, el error en la energía comienza a crecer linealmente para el integrador AKDK (línea azul), mientras que para HHS (línea negra) ocurre después de 10^5 períodos orbitales. Por el contrario, Leapfrog (línea roja) solo sigue oscilando. En cuanto al resto de las cantidades orbitales, la desviación de la conservación comienza después de $\sim 10^5$ períodos

orbitales para AKDK, mientras que los otros dos integradores se mantienen oscilando. Respecto al espacio-fase (columna derecha de la figura 3.5) los integradores se comportan igual, preservando con precisión las órbitas. Todos los integradores pueden resolver correctamente el problema, sin embargo, el tiempo de ejecución para HHS es 0.8 el correspondiente a AKDK; además, HHS conserva mejor la energía que AKDK después de 500 tiempos orbitales.

Para el caso de la binaria excéntrica ($e = 0.9$), elegimos un límite de conservación de energía del orden de 10^{-3} . Los resultados se muestran en la figura 3.6, en donde los paneles de la izquierda muestran, de arriba a abajo, el error fraccional de la energía, el error del semieje mayor y el error en la excentricidad, mientras que los paneles de las columnas central y derecha muestran el espacio-fase al inicio y alrededor de 10^4 periodos orbitales, respectivamente. De la figura se observa que hasta los primeros 20 periodos orbitales todos los integradores coinciden, incluyendo el espacio-fase (columna central). Después de los 20 periodos orbitales, el error en la energía, semieje mayor y excentricidad comienzan a desviarse sistemáticamente para AKDK (línea azul, columna izquierda). HHS (línea negra) también comienza a desviarse (columna izquierda) pero después de 5000 periodos orbitales. En contraste, Leapfrog (línea roja) solo continúa oscilando sin desviarse de la conservación. En cuanto al espacio-fase, se observa que, alrededor de 10^4 periodos orbitales (columna derecha de la figura 3.6), el semieje mayor se conserva para Leapfrog y HHS, mientras que para AKDK el semieje mayor se encogió. También, del espacio-fase se observa que hay una precesión de las órbitas, sin embargo, todos los integradores parecen mostrar la misma cantidad de precesión de la fase orbital. Dado que las órbitas precesaron, no es obvio rastrear virtualmente una desviación o un acuerdo con las condiciones iniciales de las proyecciones del espacio-fase. Por esta razón, en la figura 3.7 seleccionamos momentos específicos en donde la fase del plano de las condiciones iniciales (en color amarillo) coincide con la órbita a un tiempo posterior. De la columna izquierda de la figura podemos ver que, además de conservar el semieje mayor, HHS conserva la proyección del espacio fase, pues la órbita en $t \sim 9.5 \times 10^3 t_{\text{orb}}$ es igual a la órbita inicial. Los paneles de la derecha de la misma figura muestran un caso donde la fase del plano tanto de Leapfrog como de AKDK coincide con la condición inicial. Como es de esperarse Leapfrog conserva el espacio-fase, pero AKDK no, pues su semieje mayor se encoge y se modifica el espacio-fase respecto a la condición inicial. Para esta prueba de un sistema binario excéntrico, HHS es ~ 5 veces más rápido que AKDK, y además parece ser más preciso y estable, pero no más que Leapfrog.

De estas pruebas con sistemas binarios, concluimos que para integraciones cortas los tres integradores coinciden entre ellos. Además, las pruebas sugieren que para una integración larga, HHS parece más estable y rápido que AKDK, pero no más que Leapfrog. En el siguiente capítulo (capítulo 4), se estudiará más a fondo esta aparente ventaja de HHS sobre AKDK.

3.2.5. Caracterización del código

Las capacidades adaptativas del algoritmo HHS abren la posibilidad de un código altamente eficiente adecuado para ser utilizado en simulaciones de N -cuerpos grandes. En esta subsección cuantificamos la diferencia de rendimiento en comparación con

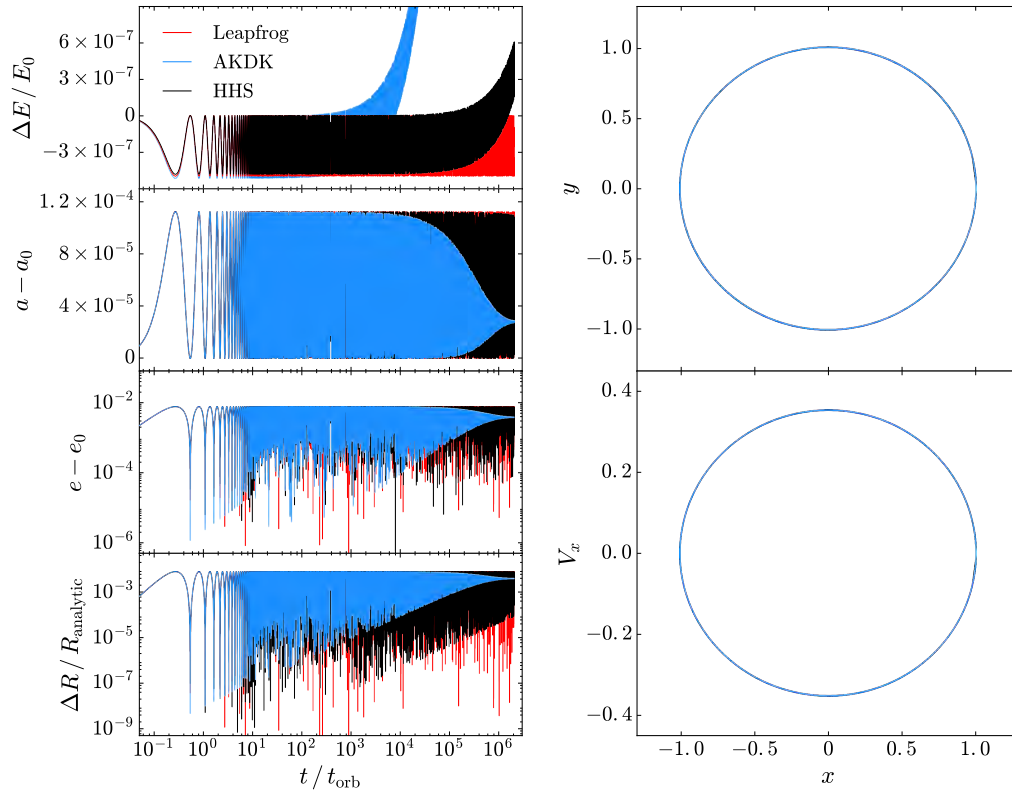


Figura 3.5: Binaria circular simulada por 2×10^6 períodos orbitales usando los tres integradores: Leapfrog (línea roja), HHS (negro) y AKDK (azul). El lado izquierdo muestra, de arriba a abajo, el error de energía, cambio del eje semi-mayor (a), cambio en la excentricidad (e), y error en la posición al comparar con la solución analítica, en función del tiempo. El lado derecho muestra el plano XY (arriba-derecha) y el espacio fase (abajo) para las primeras 10^3 órbitas. Leapfrog y HHS manejan con mucha precisión la evolución de la binaria, mientras que AKDK muestra una desviación sistemática de la conservación después de 10^3 períodos orbitales.

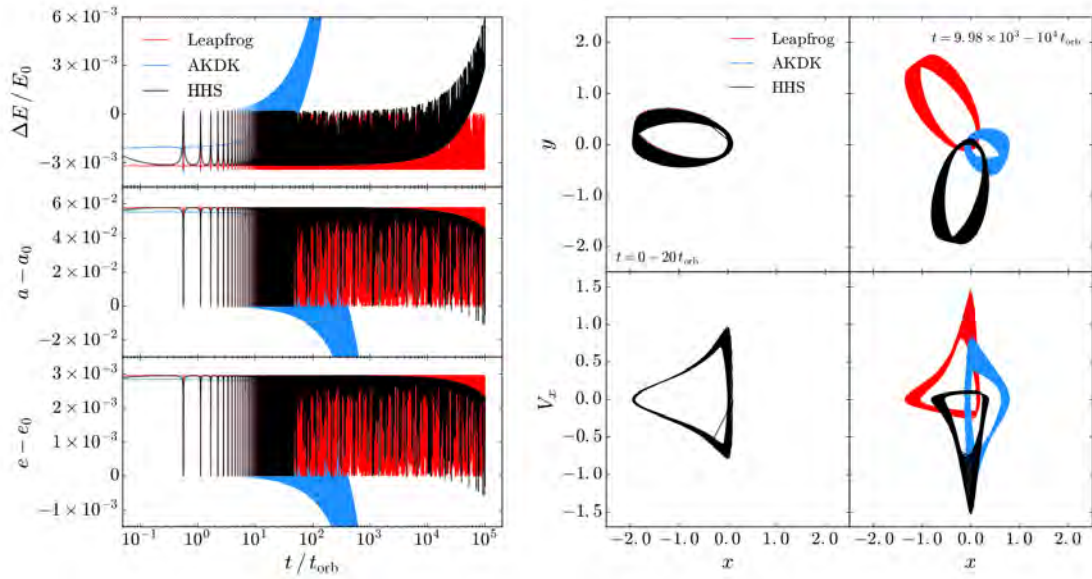


Figura 3.6: Como la figura 3.5, pero para la binaria excéntrica ($e = 0.9$) simulada por 10^5 periodos orbitales. El plano XY (arriba-derecha) y el espacio fase (abajo-derecha) se muestran para las primeras 20 órbitas (columna central) y para las últimas 20 órbitas después de 10^4 periodos (columna derecha). Leapfrog y HHS manejan con buena precisión la evolución de la binaria, mientras que AKDK muestra una desviación sistemática de la conservación después de 20 periodos orbitales y tiene órbitas inestables.

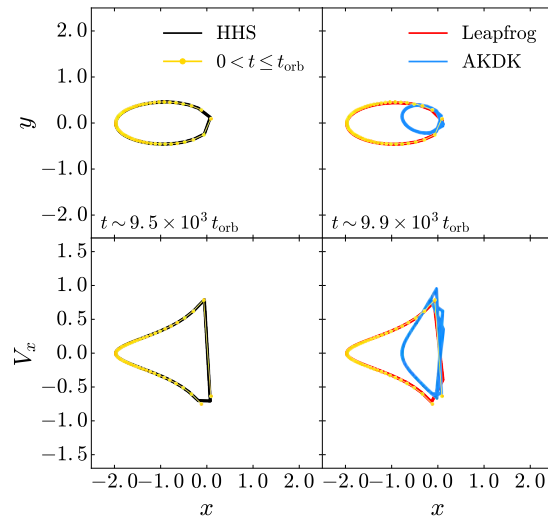


Figura 3.7: Comparación con las condiciones iniciales (amarillo) después de 10^4 órbitas para los integradores de la figura 3.6. Los paneles de la izquierda muestran un periodo orbital después de evolucionar al sistema binario durante 9.5×10^3 periodos para HHS, y los paneles de la derecha después de 9.9×10^3 periodos para Leapfrog y AKDK. Elegimos estos tiempos para mostrar que el espacio-fase no cambia apreciablemente sobre la integración para Leapfrog y HHS, mientras que para AKDK es notorio el cambio. Es importante mencionar que, para tiempos de integración más cortos, no hay diferencia entre la precisión de los tres métodos, sin embargo, la eficiencia es diferente.

algunos de los integradores más utilizados en códigos de N -cuerpos: Leapfrog, AKDK y Hermite de cuarto orden. Para caracterizar el rendimiento del código, optamos por utilizar algunas métricas: 1) tiempo total de pared o de ejecución (wall-clock time) y, para la versión paralela de OMP, 2) aceleración (en inglés se conoce como “speed-up”⁷) y 3) eficiencia computacional⁸. Finalmente, realizamos algunos perfiles del código con el fin de definir posibles vías de mejora computacional.

Para hacer la caracterización del código, adoptamos como modelo de referencia un halo aislado con partículas de igual masa siguiendo el perfil de densidad NFW predicho por cosmologías de materia oscura no colisionales [111]. El amplio rango de densidad y los diferentes tiempos dinámicos correspondientes lo convierten en un sistema adecuado para un código de paso de tiempo adaptativo. Para la integración de nuestro modelo fiducial, adoptamos $G = 1$ (constante gravitacional), $M_{\text{vir}} = 1$ (masa virial) y $r_s = 1$ (longitud de escala, también llamado radio característico), como unidades físicas.

Rendimiento: Código serial

Ejecutamos la versión serial del código para nuestro halo aislado fiducial con partículas que van desde 2000 a 3.2×10^4 , durante dos tiempos dinámicos. Para la prueba asumimos una conservación de energía comparable entre los integradores. Buscamos y encontramos parámetros de integración (niveles de pasos de tiempo, parámetro de precisión η ; ver capítulo 4 para una descripción de dichos parámetros) que hacen que el código HHS sea 65 % más rápido (basado en el tiempo de pared) que el Leapfrog para el caso con 2000 partículas, y 9 % para el caso de 8000 partículas usando un procesador Xeon. Es importante mencionar que ejecutamos dichas pruebas diez veces para cada caso, y la desviación estándar de los datos es del 0.1 %; por lo tanto, los números citados son robustos a tales cambios. Como era de esperar, los parámetros de los integradores, como el número de niveles de pasos de tiempo y la ocupación de partículas en dichos niveles tienen un efecto sobre el rendimiento de HHS. Haciendo una estimación del tiempo de ejecución de HHS (T_{HHS}), para un paso de tiempo, tenemos:

$$T_{\text{HHS}} \propto \sum_{k=0}^K 2^k n_k^2 \quad (3.7)$$

donde K es el número total de niveles de recursividad y n_k es el número de partículas en el nivel de recursividad k (con $\sum n_k = N$). Entonces, el tiempo de ejecución de HHS depende de la población de los niveles de refinamiento. Por lo tanto, un histograma de pasos de tiempo para HHS puede ayudarnos a buscar combinaciones de parámetros que ayuden a la eficiencia de HHS. La figura 3.8 ilustra dicha situación. El panel de la izquierda muestra la ocupación de partículas en los niveles de pasos de tiempo para el caso con 2000 partículas. Es fácil notar que sólo las partículas que requieren un paso

⁷La aceleración o Speed-up es una medida del rendimiento de un programa paralelo. Mide la relación entre el tiempo de ejecución secuencial y el tiempo de ejecución en paralelo. Más adelante en el capítulo se verá con mayor detalle.

⁸La eficiencia es una medida del uso de los recursos computacionales. Mide la relación entre el desempeño y los recursos utilizados para lograr ese desempeño. Más adelante en el capítulo se verá con mayor detalle.

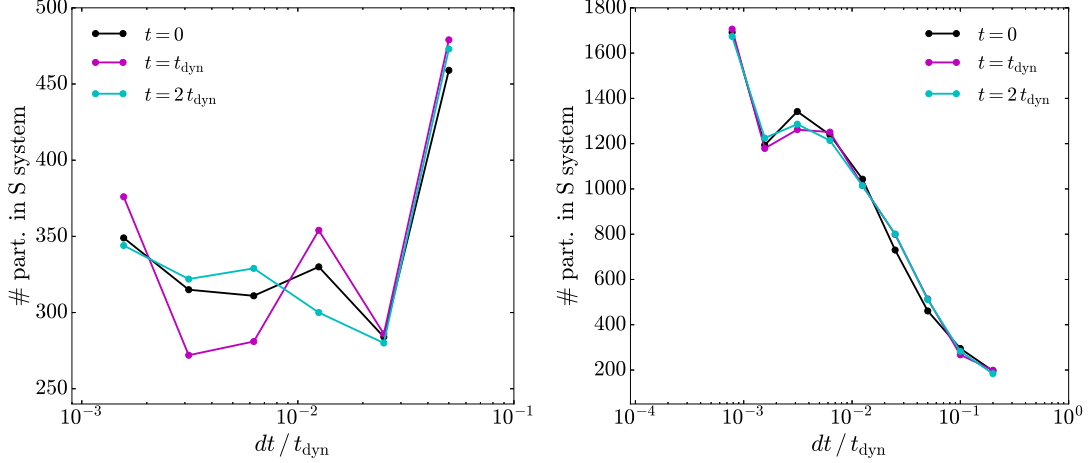


Figura 3.8: Histogramas de los pasos de tiempo para el modelo fiducial del halo aislado con $N = 2000$ (*izquierda*) y 8000 (*derecha*) partículas, usando los mismos parámetros del integrador HHS para ambos casos.

de tiempo pequeño pueblan los niveles profundos, y el resto mantiene pasos de tiempo grandes. Es justo mencionar que con 2000 partículas las regiones internas más densas no están representadas. El panel de la derecha muestra el caso con 8000 partículas sin actualizar los parámetros del código. En este caso la población de los niveles tiene una pendiente con el signo opuesto; como consecuencia, el código es solo 9% más rápido que Leapfrog, claramente se debe hacer más trabajo para poder empujar el rendimiento de HHS.

Por otro lado, Hermite parece menos eficiente, sin embargo, no intentamos investigar vías de mejora; esto es probablemente el resultado de su naturaleza no simpléctica. No podemos afirmar una búsqueda exhaustiva de parámetros de los integradores para todas las pruebas, pero mostramos una prueba de concepto de que el esquema de integración HHS potencialmente podría ser altamente eficiente. En el capítulo 4, mostramos otros casos favorables para HHS en comparación con AKDK.

La tabla 3.1 muestra el tiempo de ejecución empleado para integrar el halo aislado fiducial hasta dos tiempos dinámicos para diferentes números de partículas, N , y para diferentes procesadores. Dichos tiempos están normalizados respecto a la simulación con $N = 2000$ partículas usando HHS y ejecutada en un el procesador Intel Xeon. De esta tabla vemos que la eficiencia del integrador HHS respecto a Hermite depende tanto del procesador como del tamaño del problema. Por ejemplo, ejecutar el código serial en la arquitectura RISC (procesador Power9) muestra que HHS es ~ 7 veces más rápido que Hermite para el caso con $N = 2000$, mientras que ejecutar el mismo código en uno de los procesadores CISC (AMD Epyc) da como resultado que HHS es ~ 10 más rápido que Hermite, lo que muestra un efecto en la arquitectura CPU. De hecho, el tiempo de ejecución para todos los integradores es menor en los procesadores CISC (exceptuando Hermite con $N \gtrsim 8000$ en AMD Epyc) que en el procesador RISC. A medida que el tamaño del problema aumenta, la diferencia de rendimiento se reduce. Por ejemplo, de las pruebas con el procesador Xeon vemos que para $N = 2000$

Integrador	N	Xeon	Power9	AMD EPYC	AMD Ryzen7	AMD Ryzen5
HHS	2000	1.00	1.98	1.25	1.13	1.02
	4000	5.22	10.67	6.54	6.02	5.19
	8000	24.67	50.48	31.18	28.62	24.89
	16000	121.52	247.96	156.08	141.21	121.99
	32000	588.11	1170.62	756.34	671.27	574.31
Hermite	2000	5.96	14.13	12.66	5.64	7.12
	4000	26.37	58.41	50.51	26.81	29.32
	8000	123.40	233.10	250.57	109.61	133.29
	16000	485.51	946.45	2145.61	585.01	
	32000	1589.01	3998.09	4459.32	1924.65	

Cuadro 3.1: Tiempo de ejecución, normalizado respecto a la simulación de halo aislado con $N = 2000$ partículas usando HHS y ejecutada en un procesador Xeon. Consideramos diferentes valores para el número de partículas y diferentes arquitecturas de CPU. Claramente, los parámetros de HHS deben ajustarse para cada problema, sin embargo, los datos de la tabla muestran la ventaja de HHS sobre Hermite con la misma conservación de energía.

partículas, HHS es ~ 6 veces más rápido que Hermite, mientras que para $N = 32000$ HHS es sólo ~ 3 veces más rápido. Sin embargo, es importante aclarar que el número de niveles de pasos de tiempo y el criterio de apertura de éstos juegan un papel muy importante en la dependencia con el número de partículas (como se mostró en la figura 3.8 y se discutió anteriormente) y no sólo la arquitectura del procesador. Nuestro ejercicio es solo exploratorio, porque el código nunca se optimizó para una arquitectura en particular o se probó el compilador nativo de cada procesador, más aún, la frecuencia del procesador no es la misma para cada CPU. Encontramos que de hecho existe una dependencia del rendimiento de HHS con la arquitectura, obteniendo el mayor beneficio respecto a Hermite en la arquitectura RISC (procesador Power9), especulamos que la recursividad del integrador se maneja mejor en tal caso. Está abierto a investigar cuánto se puede mejorar el rendimiento en las CPU RISC. En general, llegamos a la conclusión de que para obtener un buen rendimiento, todas estas características: los parámetros del algoritmo (niveles de pasos de tiempo, etc.) y la arquitectura deben considerarse simultáneamente, pero un código HHS altamente eficiente parece alcanzable. Es importante mencionar que la implementación serial de la versión iterativa de HHS (denominada HHSv5) tiene la misma precisión y tiempo de ejecución comparable que la recursiva (resultados presentados en esta subsección).

Rendimiento: Código Paralelizado en OMP

Para analizar el rendimiento de la versión paralela en OMP de nuestro código, medimos el tiempo de ejecución, la aceleración (“speed-up” en inglés), la eficiencia y la fracción secuencial invertidos para integrar nuestro halo aislado fiducial hasta dos tiempos dinámicos en un procesador AMD Opteron. La aceleración, $S(p)$, mide la rapidez de una aplicación paralela con respecto a la secuencial correspondiente, y se define como:

$$S(p) = \frac{T(1)}{T(p)} \quad (3.8)$$

donde $T(p)$ es el tiempo de ejecución de la aplicación usando p unidades de cálculo. Si una aplicación paralela alcanza $S(p) = p$, lo que se conoce como *aceleración lineal*, entonces se considera que está *perfectamente* codificada. La ley de Amdahl establece que la aceleración máxima que puede alcanzar una aplicación paralela usando p unidades de procesamiento es:

$$S(p) \leq \frac{1}{f + (1 - f)/p} \quad (3.9)$$

donde f es la fracción secuencial de la aplicación, que es la fracción de cómputo que sólo se puede realizar secuencialmente ($0 \leq f \leq 1$).

La eficiencia, $E(p)$, mide la fracción de tiempo durante la cual un elemento de procesamiento es empleado de manera útil por la aplicación paralela, y se define como:

$$E(p) = \frac{S(p)}{p} \quad (3.10)$$

La eficiencia tiene valores entre cero y uno, donde el valor ideal para una aplicación en paralelo es uno. Valores bajos de eficiencia significan que la efectividad con la que

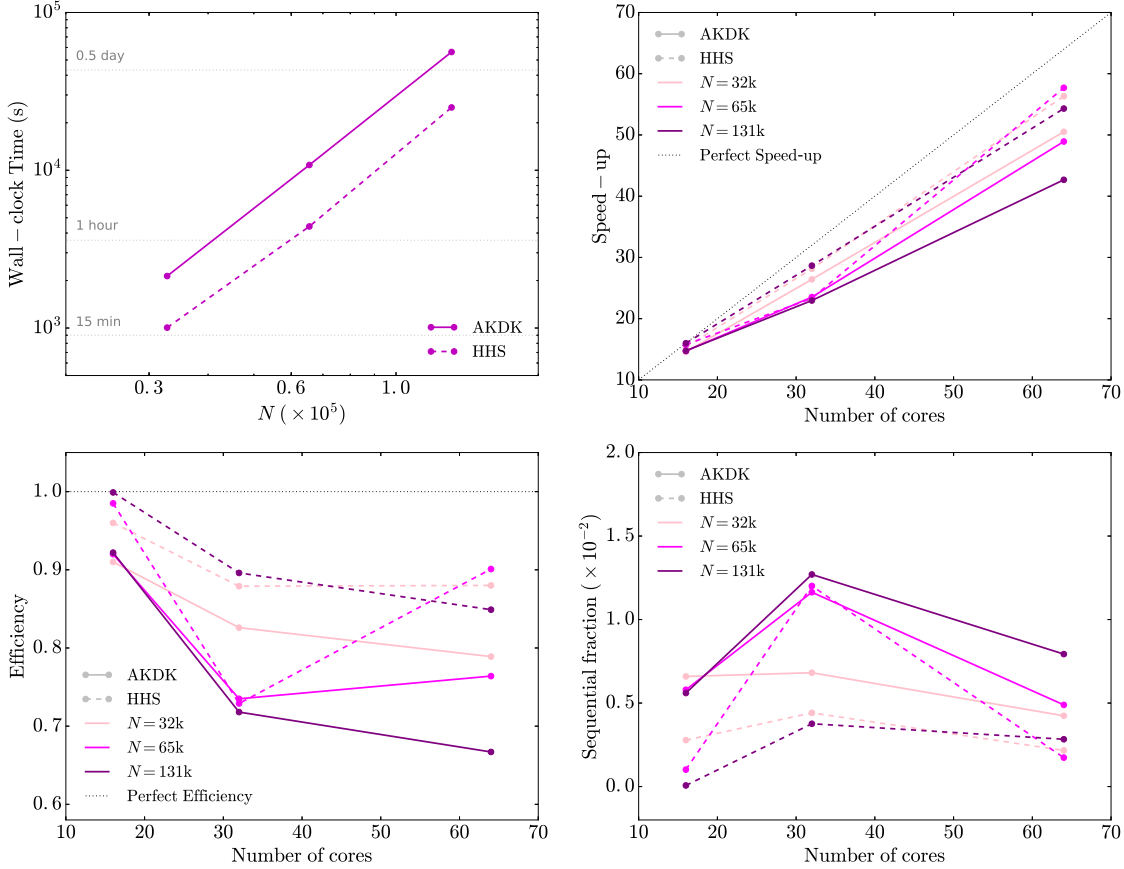


Figura 3.9: Métricas de rendimiento. Tiempo de pared (*arriba-izquierda*) como función del número de partículas N para la versión OMP del código para AKDK (línea continua) y HHS (línea punteada) usando 64 núcleos en un procesador AMD Opteron. La aceleración (*arriba-derecha*), la eficiencia (*abajo-izquierda*) y la fracción secuencial (*abajo-derecha*) de nuestro código en función del número de núcleos usados en la simulación del halo aislado fiducial con diferente número de partículas ($32k < N < 131k$).

la aplicación utiliza los elementos de procesamiento es baja. Es importante mencionar que, al menos para las simulaciones de N -cuerpos, la mayor aceleración y eficiencia se alcanza siempre que se logra un equilibrio adecuado entre el número de unidades de cálculo y el tamaño del problema (en términos de N).

La figura 3.9 muestra los resultados de rendimiento de la versión OMP del código para los integradores AKDK (línea continua) y HHS (línea discontinua) al simular el halo aislado fiducial. El panel superior izquierdo muestra el tiempo de pared invertido para integrar el sistema de N -cuerpos en función del número de partículas, N , usando 64 núcleos en un procesador AMD Opteron. Se observa que el tiempo de ejecución para HHS es menor que AKDK en todos los casos, siendo en promedio $\sim 50\%$ más rápido HHS que AKDK.

Además, la figura 3.9 también muestra las métricas de aceleración (arriba a la derecha), eficiencia (abajo a la izquierda) y fracción secuencial (abajo a la derecha) usando diferentes números de partículas $N = 3.2 \times 10^4$ (color rosa), 6.5×10^4 (color

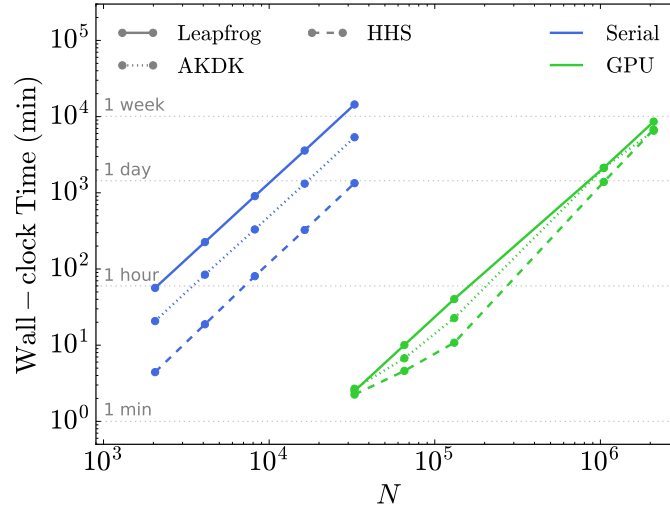


Figura 3.10: Escalamiento del código con el número de partículas N siguiendo el halo aislado fiducial durante dos tiempos dinámicos. Se muestra el escalamiento para los integradores Leapfrog (línea continua), AKDK (línea punteada) y HHS (línea discontinua); y para las versiones serial (color azul) y GPU (color verde) del código. En general, HHS tiene los tiempos más cortos de ejecución.

fucsia) y 1.31×10^5 (color morado), en función del número de núcleos utilizados. En general, las métricas de HHS (líneas discontinuas) se encuentran más cercanas a los valores ideales (marcados con líneas punteadas negras) que AKDK (líneas continuas) para los tres tamaños del problema. Cabe mencionar que una exploración más fina para el número de núcleos ayudaría a suavizar el comportamiento de las métricas pero con los datos obtenidos tenemos un panorama del comportamiento general.

Rendimiento: Código Paralelizado en GPU

Las GPU modernas son una opción valiosa para la arquitectura paralela para el problema de N -cuerpos porque tienen una gran cantidad de núcleos y los modelos recientes incluyen 32 gigabytes de memoria RAM, que permiten acomodar en su interior problemas no despreciables. Para medir el rendimiento de la versión paralela en GPU del código, descrita en la sección 3.2.3, comparamos el escalamiento del tiempo de pared como función del tamaño del problema con el correspondiente para el código secuencial. Realizamos las pruebas utilizando un procesador IBM Power9 y una tarjeta GPU NVIDIA Tesla V100.

La figura 3.10 muestra el tiempo de pared empleado para integrar nuestro halo aislado fiducial hasta dos tiempos dinámicos en función del número de partículas, N , para las versiones serial (líneas azules) y paralela en GPU (líneas verdes) del código utilizando los tres integradores: Leapfrog (línea continua), AKDK (línea punteada) y HHS (línea discontinua). Como era de esperar, el código se ve muy favorecido por la naturaleza paralela. Por ejemplo, para el caso con 32 k partículas y para HHS, el tiempo de ejecución en la GPU V100 (líneas verde discontinua) es alrededor de dos minutos

mientras que para el código serial es de un día. Sin mencionar que una extrapolación a 10^6 partículas para el caso serial, el tiempo de ejecución indica varios meses y para la GPU V100 es alrededor de un día. De la misma figura, también se observa que para $N \sim 2 \times 10^6$ partículas el tiempo de ejecución entre los integradores (para la versión GPU) es más parecido. Sin embargo, es importante señalar que para estas pruebas de escalamiento se mantuvieron fijos los parámetros de integración de cada uno de los tres integradores y sólo se varió el tamaño del problema.

Finalmente, como una prueba adicional, simulamos el halo fiducial con $N = 10^5$ partículas variando el tamaño del bloque empleado en la GPU (es decir, el número de subprocesos por bloque). Como resultado encontramos que no hay dependencia del tiempo de ejecución del programa con el tamaño del bloque usado en la GPU, para cualquiera de los integradores. Concluimos que el tamaño del bloque usado en la GPU no es responsable de la mejora de la eficiencia entre los integradores.

Perfilado y Optimización

Aunque el principal indicador de eficiencia es el tiempo total de pared, utilizamos herramientas de perfilado para cuantificar las operaciones más costosas dentro de cada algoritmo, así como para definir estrategias para posibles mejoras. Comparamos las versiones secuencial y paralela en GPU del código. Debido a que en la versión HHSv3 del código realizamos la recursividad asociada con la construcción de la jerarquía de pasos de tiempo en la CPU, sospechamos que la comunicación continua entre GPU-CPU podría limitar el rendimiento. Para resolver tal problema, inicialmente desarrollamos HHSv4 usando memoria unificada de CUDA, con la intención de obtener beneficios de una comunicación CPU-GPU más rápida. También, realizamos una prueba utilizando la comunicación rápida CPU-GPU de NVLINK. Dicha prueba no tuvo éxito posiblemente por 2 razones: de acuerdo con [112] el tamaño del paquete de información no era lo suficientemente grande para obtener una velocidad significativa sobre la comunicación PCIe4, por lo que pruebas con un tamaño de problema grande pueden ser una vía interesante. La otra posible explicación es que las comunicaciones no son en realidad el principal cuello de botella. Para explorar tal posibilidad intentamos utilizar herramientas de recursividad dentro de CUDA 6.0, sin embargo tales librerías no preservan el orden de las operaciones, lo que produce un conflicto con la causalidad en nuestro código. Por lo anterior, implementamos una versión de HHS iterativa (HHSv6). Sin embargo, la precisión y el tiempo de ejecución de HHSv3, HHSv4 y HHSv6 son comparables, entonces podemos concluir que la comunicación GPU-CPU no domina el tiempo de cálculo en nuestras pruebas. Para investigar si el rendimiento de HHS está dominado por la parte del código que se ejecuta en CPU, analizamos la ejecución de HHSv3 usando un procesador CISC Xeon con una GPU V100, el rendimiento fue el mismo que el obtenido con la CPU RISC Power9 y GPU V100, lo que sugiere que ni la arquitectura CPU ni las comunicaciones CPU-GPU dominan el costo de la simulación.

Para perfilar la versión secuencial del código usamos el perfilador de GNU, gprof⁹ (ver gprof-docs). De manera similar, para perfilar de forma detallada la versión del

⁹gprof calcula la cantidad de tiempo empleado en cada rutina del código y muestra las estadísticas de tiempo del código y sus rutinas.

código en GPU, utilizamos la herramienta de perfilación de NVIDIA nvprof¹⁰ (consulte profiler-users-guide).

La figura 3.11 muestra los resultados de perfilar la versión del código en GPU (utilizando un procesador Power9 y la GPU NVIDIA Tesla V100). Todas las pruebas se ejecutaron para el halo aislado fiducial con 10^5 partículas. Los porcentajes mostrados en la figura son relativos al tiempo total de ejecución de cada integrador y, como en otras pruebas, dependen del tamaño del problema y del hardware. En esta figura vemos que para Leapfrog (panel superior izquierdo) y Hermite (panel superior derecho) la parte más cara del código es el cálculo de aceleración y jerk (solo para Hermite) que representa casi el 100 % del tiempo de ejecución total. Para AKDK (panel central izquierdo) vemos que la búsqueda/selección de las partículas activas representa $\sim 10\%$ del total, mientras que el cálculo de la aceleración representa $\sim 90\%$ del tiempo de ejecución total. En el caso de HHS (HHSv3, panel central derecho) vemos que el cálculo de la jerarquía de pasos de tiempo y el cálculo de la aceleración representan casi el mismo porcentaje del tiempo total ($\sim 45\%$ y $\sim 50\%$, respectivamente), mientras que las comunicaciones CPU-GPU (nombradas como Memcpy) representan un porcentaje menor ($\sim 4\%$), lo cual confirma nuestras especulaciones anteriores basadas en diferentes comparaciones entre las versiones del código HHS en diferentes arquitecturas. Por lo anterior, es claro que reducir el costo de construir la jerarquía de pasos de tiempo es una vía promisoría de mejora en el rendimiento. Cabe mencionar que el perfilado de la versión serial del código arrojó resultados similares.

Con base en la discusión anterior, realizamos una séptima versión (HHSv7) de la implementación de HHS, la cual es una extensión de la versión HHSv3 que permite controlar la frecuencia del cálculo de la estructura de paso de tiempo (TSS, por sus siglas en inglés) a través de una bandera en el código. En esta versión, se requiere guardar el TSS, a diferencia de la versión HHSv3. La figura 3.12 (a) muestra el diagrama de flujo de esta versión, donde *evolve_HHSv3* representa la rutina de evolución HHSv3 (ver fig. 3.3 para el diagrama de flujo), y la rutina *evolve_nTSS* se muestra en 3.12 (b). El panel inferior de la figura 3.11 muestra el perfilado de esta versión, de donde vemos que ahora el cálculo de la aceleración representa el $\sim 65\%$ del tiempo total y el cálculo de la jerarquía de pasos de tiempo representa el $\sim 30\%$.

Dado que en HHSv7 el cálculo de TSS no siempre se realiza, esperamos que esta versión sea menos precisa que la versión HHSv3 a cambio de una mayor eficiencia. Para probar esta versión en términos de precisión y rendimiento, evolucionamos nuestro halo aislado fiducial NFW con $N = 10^5$ partículas usando HHSv7 y comparamos los resultados con Leapfrog y HHSv3. De la figura 3.13, los tres casos conservan el perfil de densidad con diferencias menores al 4% en precisión dentro de r_s durante 20 tiempos dinámicos. Por otro lado, el baricentro del sistema y los momentos lineal y angular se conservan (ver figura 3.14) sin diferencia entre los tres casos. La energía total es mejor conservada por HHSv3, seguido por Leapfrog y finalmente por HHSv7, que tiene un crecimiento sistemático, lo que lo hace menos preciso que Leapfrog después de 8

¹⁰nvprof permite la recopilación y visualización de datos de perfilación desde la línea de comandos. Con nvprof se puede recopilar una línea de tiempo de las actividades relacionadas con CUDA tanto en la CPU como en la GPU, como la ejecución del kernel, las transferencias de memoria y las llamadas a la API de CUDA.

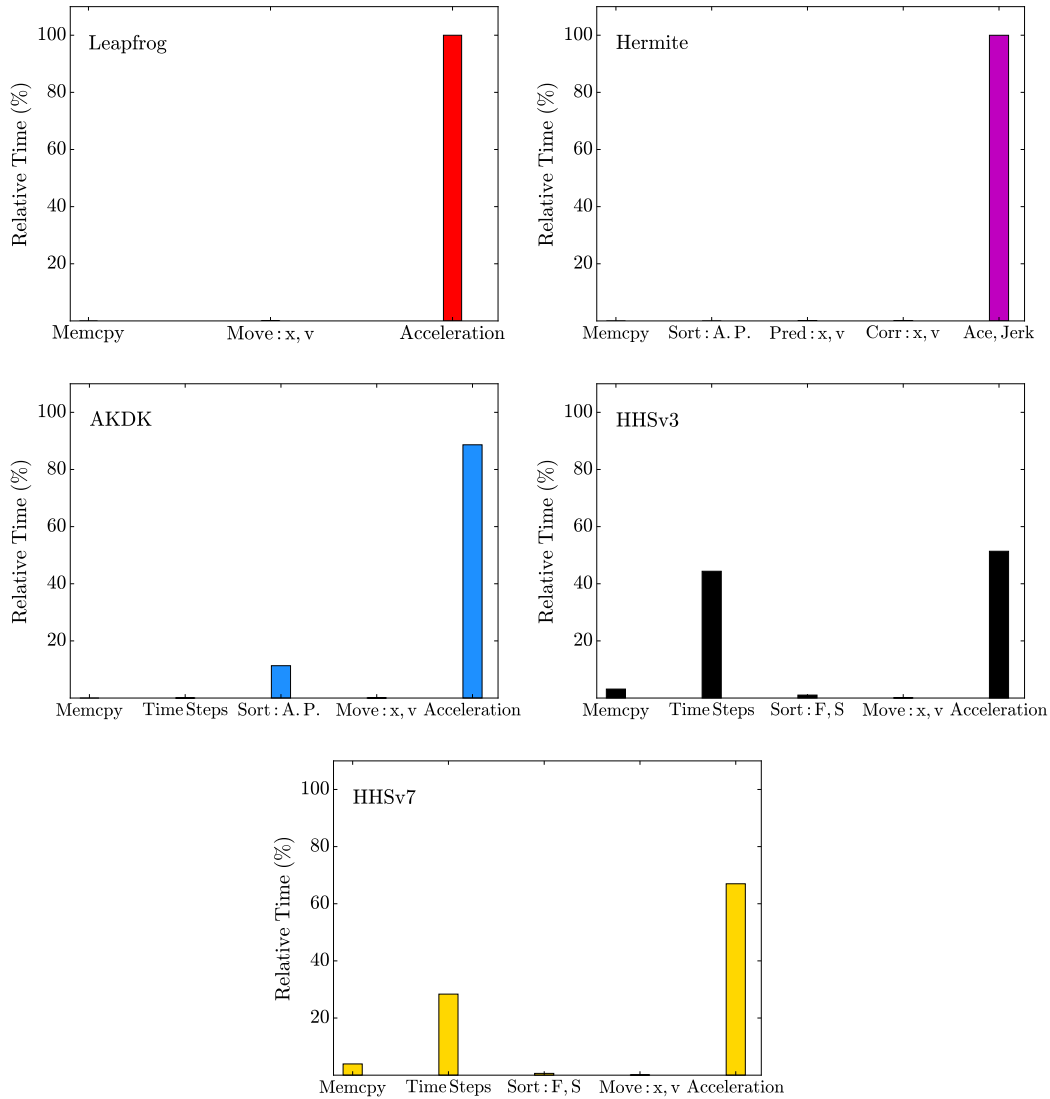


Figura 3.11: Porcentaje de los tiempos de ejecución relativos (al total) de las diferentes partes de los integradores: Leapfrog (panel superior izquierdo), Hermite (panel superior derecho), AKDK (panel central izquierdo), HHS (HHSv3, panel central derecho) y HHS guardando la estructura de pasos de tiempo (HHSv7, panel inferior). Se muestran los resultados para el halo aislado fiducial con 10^5 partículas y usando la versión paralela sobre GPU (se usó la tarjeta Tesla V100).

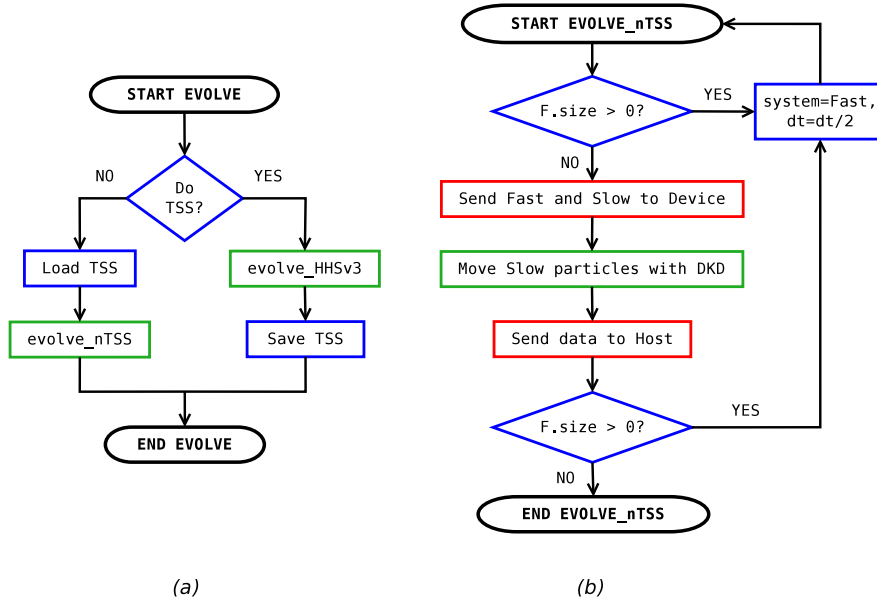


Figura 3.12: (a) Diagrama de flujo de la versión HHSv7 del código, que permite controlar el cálculo de la estructura de pasos de tiempo (TSS, por sus siglas en inglés). (b) Rutina de evolución sin el cálculo de la TSS. Los colores de los rectángulos siguen la misma convención que la figura 3.1.

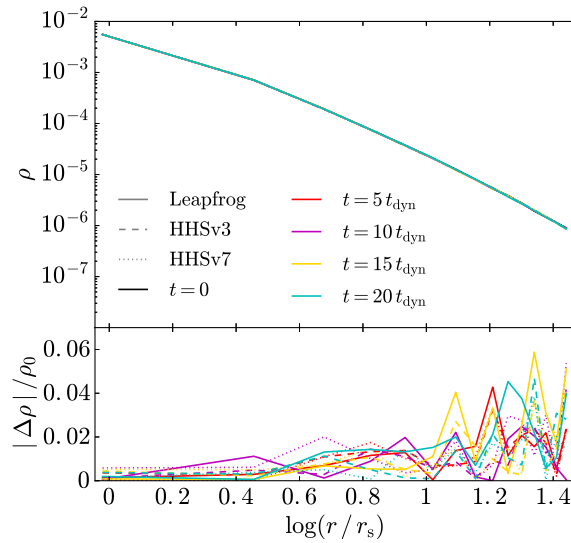


Figura 3.13: Perfil de densidad, en diferentes momentos, del halo NFW aislado con $N = 10^5$ partículas simuladas hasta 20 tiempos dinámicos para Leapfrog (líneas continuas), y las versiones HHSv3 (líneas discontinuas) y HHSv7 (líneas punteadas) del código de HHS. Los tres casos conservan el perfil de densidad con menos del 4% de precisión, dentro de r_s .

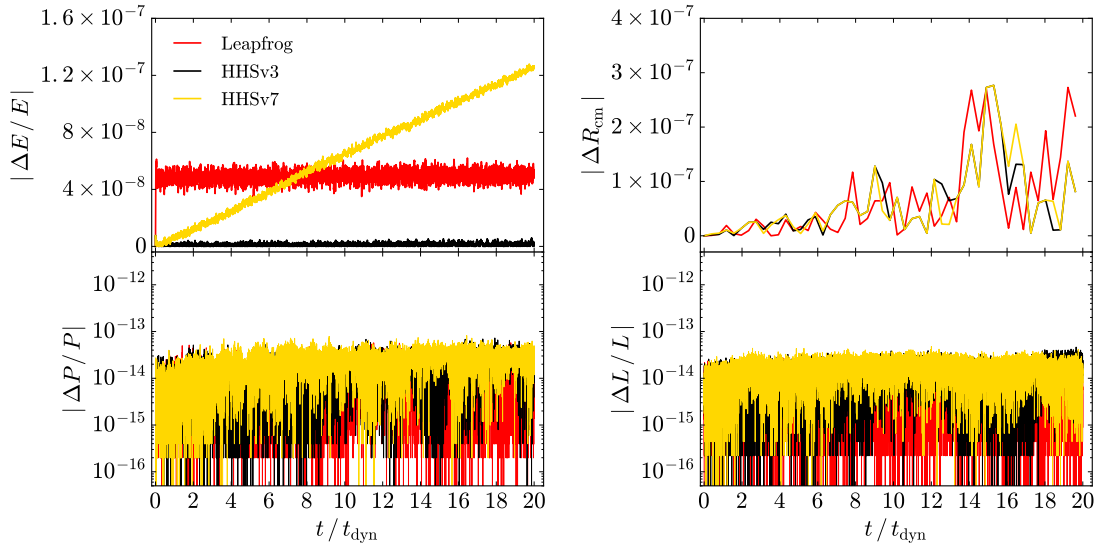


Figura 3.14: Implementación optimizada del código HHS. Error en cantidades conservadas para un halo NFW aislado con $N = 10^5$ partículas simuladas hasta 20 tiempos dinámicos. Se muestran el error de energía (*panel superior izquierdo*), el cambio en la posición del centro de masa (*panel superior derecho*), el momento lineal (*panel inferior izquierdo*) y el cambio en el momento angular (*panel inferior derecho*) para Leapfrog (línea roja) y las versiones del código HHSv3 (línea negra) y HHSv7 (línea amarilla) del integrador HHS. Aunque el código optimizado HHSv7 tiene un crecimiento sistemático de energía, hasta 20 tiempos dinámicos el error de energía sigue siendo comparable a Leapfrog. Para tal caso, HHSv7 es un 20 % más rápido que Leapfrog.

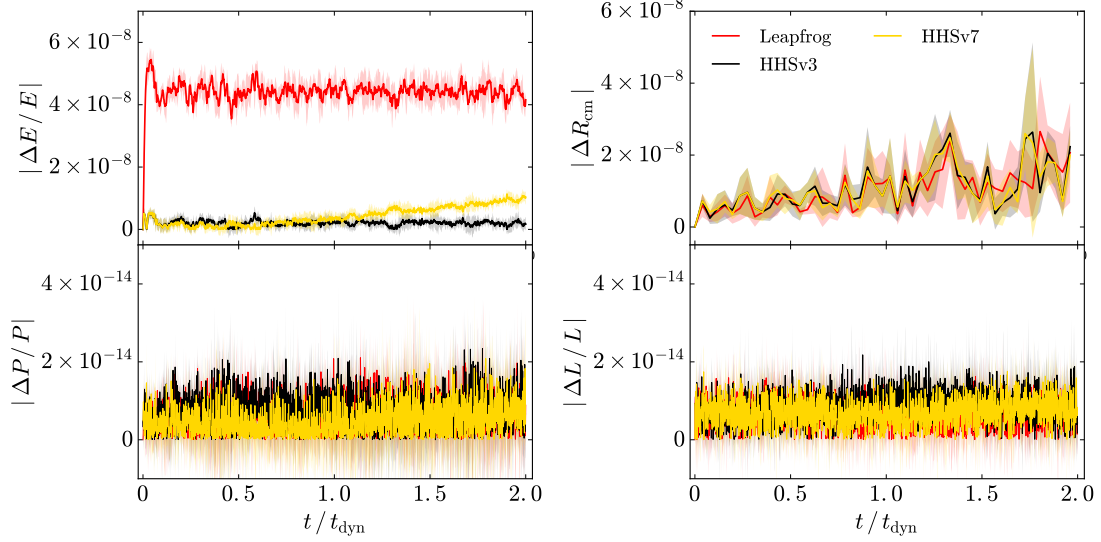


Figura 3.15: Similar a la figura 3.14, pero con el promedio sobre diferentes realizaciones de las condiciones iniciales simuladas hasta dos tiempos dinámicos. Las porciones sombreadas representan una desviación estándar de un sigma propagada sobre las realizaciones. Hay robustez de los resultados frente a diferentes realizaciones de las condiciones iniciales.

tiempos dinámicos. En cuanto al rendimiento, para este caso HHSv7 es un 20% más rápido que Leapfrog, y HHSv3 es un 77% más lento que HHSv7. Por lo tanto, HHSv7 mejora el rendimiento de HHSv3 a cambio de precisión. Este es un muy buen ejemplo de que nuestra implementación del integrador HHS puede ser más eficiente, pero menos precisa (con cierta dependencia con el tiempo de integración como se muestra en [113]), alternativamente, puede ser más precisa, para tiempos de integración lo suficientemente cortos, pero menos eficiente que Leapfrog. En el capítulo 4 de esta tesis mostraremos otros casos con la versión HHSv7.

Para probar si este resultado es robusto frente a diferentes realizaciones de las condiciones iniciales, ejecutamos el mismo halo aislado fiducial NFW hasta 2 tiempos dinámicos para los tres casos utilizando diferentes realizaciones (semillas) de las condiciones iniciales. Como puede verse en la figura 3.15 el comportamiento anterior se mantiene sobre diferentes realizaciones, y también el rendimiento. Por lo tanto, los resultados anteriores son robustos frente a diferentes realizaciones de las condiciones iniciales.

Aunque logramos optimizar el código HHS, aún es deseable poder manejar problemas comparables a 10^6 o más. Para mejorar aún más el rendimiento del código, implementamos una última (octava) versión (HHSv8), que permite el uso de más de una GPU. En esta versión usamos MPI ¹¹ (en español, interfaz de Paso de Mensajes;

¹¹MPI es una especificación de interfaz de biblioteca de paso de mensajes. MPI aborda principalmente el modelo de programación paralela de paso de mensajes, en el que los datos se mueven del espacio de direcciones de un proceso al de otro proceso mediante operaciones cooperativas en cada proceso.

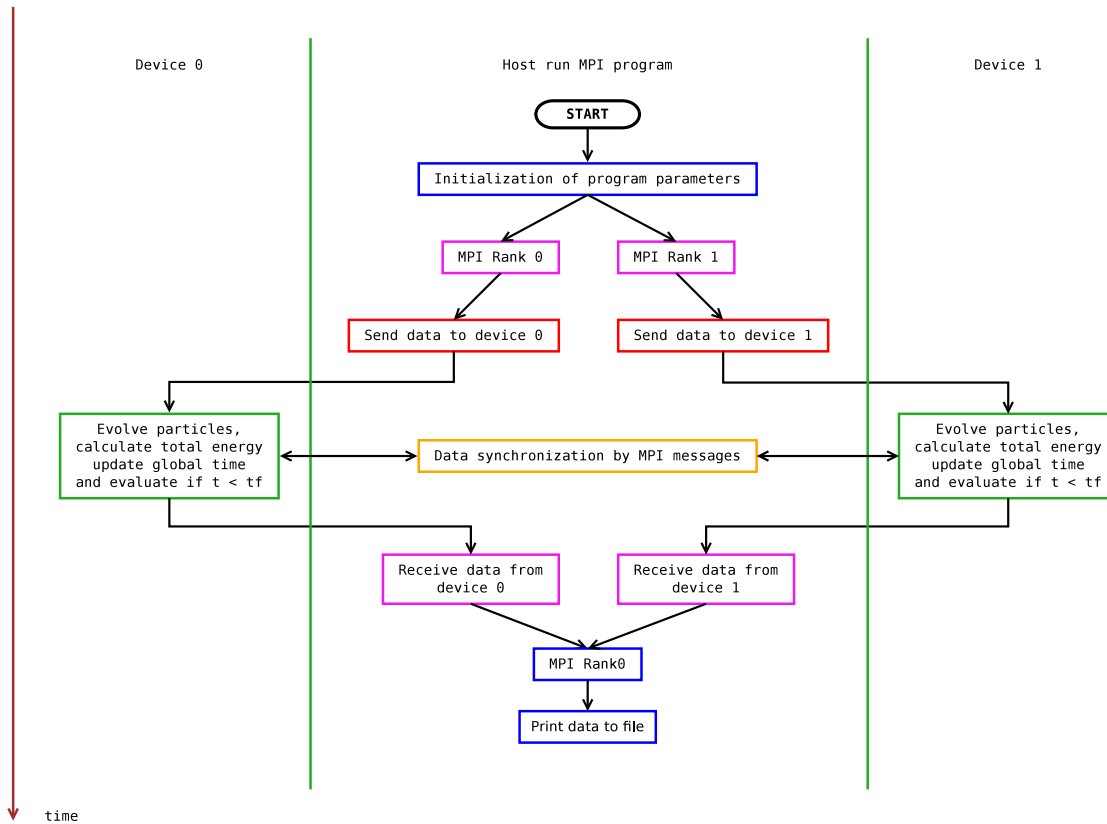


Figura 3.16: Diagrama representativo de la versión HHSv8 del código para el uso de dos GPU's como ejemplo. HHSv8 usa MPI para manejar las comunicaciones de la GPU. Los recuadros morados representan la ejecución paralela de cada proceso MPI y el recuadro naranja la sincronización GPU a través de mensajes MPI. Los otros colores siguen la misma convención que la figura 3.1.

consulte MPI-Forum) para manejar las comunicaciones de la GPU, mientras que la GPU se ocupa, en el caso del integrador HHS, del cálculo del paso de tiempo de las partículas (si es necesario) y de la rutina que mueve/evoluciona a las partículas (posiciones y velocidades), que incluye el cálculo de fuerza por pares. Más detalladamente, el problema de N -cuerpos se divide igualmente entre cada proceso MPI, con uno de los procesos MPI actuando como maestro. Luego, cada proceso MPI se asigna a una GPU, y cada proceso realiza los mismos pasos que la CPU en las figuras 3.1 (a) y 3.12 (cuadros azules), además de las Comunicaciones CPU-GPU y GPU-GPU (consulte la figura 3.16 para ver un diagrama de implementación representativo). Finalmente, cada GPU realiza los mismos pasos que antes (casillas verdes en 3.1 y 3.12, por ejemplo). Para la versión con múltiples GPU's, el tiempo de pared se reduce en $\sim 40\%$ para el integrador HHS cuando se usan dos GPU's en comparación con la versión de una única GPU, para $N = 1.6 \times 10^6$ partículas. Si adicionalmente incluimos la optimización que reduce la frecuencia para construir la estructura de pasos de tiempo, obtenemos 16% adicionales, lo que abre la posibilidad de seguir problemas sin colisiones más grandes.

3.2.6. Discusión y conclusiones

Introducimos NP_{splitt} , un código de N -cuerpos de suma directa que implementa diferentes integradores: el integrador simpléctico Leapfrog (paso de tiempo constante), el integrador de segundo orden AKDK (variante de Leapfrog con paso de tiempo adaptivo), el integrador de cuarto orden Hermite con paso de tiempo adaptivo y el integrador de división hamiltoniana jerárquica (HHS, por sus siglas en inglés), un integrador con pasos de tiempo adaptivos y con buena preservación de cantidades como la energía y el momento.

1. Realizamos pruebas de precisión para sistemas binarios (circulares y excéntricas). Nuestra implementación puede simular binarias de manera competitiva en comparación con los presentados en [100, 101]. En particular para la binaria excéntrica encontramos que HHS es 60% más rápido que Leapfrog debido al paso de tiempo adaptivo. Como era de esperar, los tres integradores siguen los experimentos de N -cuerpos con una buena precisión similar en términos de conservación de energía, momento lineal y angular. Prácticamente, los integradores HHS y Leapfrog tienen la misma precisión y comportamiento, mientras que AKDK es menos exacto para tiempos de integración largos.
2. Las pruebas de rendimiento ejecutadas para la versión secuencial del código en diferentes procesadores, indican que HHS funciona mejor que el Hermite tanto en la arquitectura RISC (Power9) como en la CISC (Xeon, EPYC, Ryzen 5 y 7). Sin embargo, en la arquitectura RISC el rendimiento de HHS es mejor respecto a Hermite que en la arquitectura CISC, lo que sugiere que la recursividad se maneja mejor en RISC; pero el tiempo de ejecución de nuestro código es más corto para la arquitectura CISC que para la RISC. Es justo decir que no optimizamos el código en función de la arquitectura, nuestra comparación es meramente exploratoria pero alentadora.

3. Para algunas de las combinaciones de parámetros que usamos hasta este punto, encontramos que HHS puede ser más eficiente que Leapfrog (con paso de tiempo constante), pero con una precisión similar o menor para la conservación de la energía; alternativamente, HHS puede ser más preciso que Leapfrog, pero con una menor eficiencia. Aunque el número de pasos de integración en los códigos HHS, AKDK y Hermite es menor que en Leapfrog, la construcción de la estructura de pasos de tiempo y su ocupación por partículas pueden compensar y, a veces, dominar el tiempo de cálculo, haciendo que estos integradores sean menos eficientes dependiendo de los parámetros de integración elegidos y si no se realiza ninguna optimización o si el rango dinámico del sistema a evolucionar es pequeño (como veremos en el siguiente capítulo). Por ello, es necesaria una búsqueda en el espacio de parámetros del integrador HHS para lograr una mejor eficiencia en cada caso, para lo cual resulta útil el histograma de pasos de tiempo. Para el integrador no-simpléctico Hermite de cuarto orden, encontramos que es menos eficiente que los demás si imponemos una precisión similar para la conservación de energía.
4. De los resultados del perfilado (figura 3.11) concluimos que en nuestras implementaciones de Leapfrog, Hermite y AKDK casi todo el tiempo de ejecución se dedica al cálculo de la aceleración ($\sim 100\%$ para los primeros dos, y $\sim 90\%$ para el último), mientras que en el integrador HHS solo alrededor de $\sim 50\%$ (para la GPU V100 con $N = 10^5$ partículas) se va en el cálculo de la aceleración, $\sim 45\%$ va en la definición de la jerarquía de pasos de tiempo, y $\sim 4\%$ se destina a las comunicaciones CPU-GPU. Por lo que una forma posible de mejorar el rendimiento del integrador HHS, independientemente de los parámetros elegidos, es mantener la estructura jerárquica de pasos de tiempo durante unos pocos pasos, ahorrando el costoso tiempo de su construcción.
5. Presentamos una versión del código (HHSv7) que permite guardar la jerarquía de pasos de tiempo para algunos pasos de integración, y la probamos, en términos de precisión y rendimiento, contra el integrador Leapfrog y la versión del código HHSv3 (construye la jerarquía de pasos de tiempo en cada paso de integración). Encontramos que esta versión es menos precisa que HHSv3 a cambio de una mayor eficiencia (HHSv3 es un 77% más lenta). Para la combinación de parámetros utilizada en la prueba, encontramos que el código HHSv7 es un 20% más rápido que Leapfrog y más preciso antes de 8 tiempos dinámicos. Sin embargo, la optimización hace más evidente la desviación de la conservación de la energía y se degrada a un factor de dos después de 20 tiempos dinámicos, en contraste, el código original de HHSv3 es más preciso que Leapfrog durante 20 tiempos dinámicos. Encontramos que tal resultado es robusto frente a diferentes realizaciones de las condiciones iniciales. Con esta prueba, mostramos que HHS puede ser o más eficiente pero menos preciso, o más preciso pero menos eficiente, que el integrador Leapfrog. Sin embargo, como veremos en el siguiente capítulo, es posible tener casos en los que HHS es más rápido, y con una precisión similar en la energía, que Leapfrog con paso de tiempo constante, dependiendo del rango dinámico del problema a integrar y de los parámetros de integración elegidos.

6. Finalmente, presentamos una versión del código (HHSv8) que permite el uso de más de una GPU, y usa MPI para manejar las comunicaciones GPU/CUDA. Para una simulación con $N = 1.6 \times 10^6$ partículas usando HHS, el tiempo de pared se reduce en $\sim 40\%$ cuando se usan dos GPU en comparación con la versión de una sola GPU. Si además incluimos la optimización que reduce la frecuencia para construir la estructura de pasos de tiempo, obtenemos un 16% adicional, lo que abre la posibilidad de seguir problemas no colisionales más grandes.

Concluimos que $\text{NP}_{\text{splitt}}$ muestra un muy buen desempeño con un compromiso entre precisión y velocidad. La optimización que controla la frecuencia de la jerarquía de pasos de tiempo mejora el rendimiento del código, haciéndolo más competitivo que Leapfrog, sin embargo, la precisión está sujeta a los parámetros específicos del integrador HHS (número de niveles de pasos de tiempo, etc.) así como del rango dinámico del problema a simular. La versión de dos GPU's mostró una aceleración de 40% con un 16% adicional si optimizamos la frecuencia para construir la estructura de pasos de tiempo, haciendo que el código sea capaz de simular prácticamente problemas con partículas de 10^6 y más en arquitecturas de un solo nodo o incluso en multinodo debido a las capacidades MPI/CUDA. Explorar optimizaciones adecuadas para procesadores RISC (como Power9, ARM) también es una vía interesante para el código HHS, dado que en dicha arquitectura encontramos un mejor rendimiento respecto a Hermite que en la arquitectura CISC.

Capítulo 4

Un nuevo integrador adaptivo reversible para simulaciones no-colisionales: división hamiltoniana jerárquica

En el capítulo anterior presenté el código Np_{splitt} que desarrollé en esta tesis, y mostré su desempeño en distintas arquitecturas computacionales. En este capítulo, presento un estudio del integrador de división hamiltoniana jerárquica (HHS, por sus siglas en inglés) en el contexto de simulaciones no-colisionales y comparo, en términos de precisión y eficiencia, con uno de los integradores típicamente usados en este tipo de simulaciones, el integrador AKDK. Cabe mencionar que este estudio es otro de los productos de esta tesis. HHS ha sido discutido en estudios previos, sin embargo no se ha explicado la razón de su conveniencia, el presente artículo proporciona tal explicación, así como muestra la existencia de combinaciones de parámetros con los que es más eficiente que AKDK.

4.1. Introducción: Evaluando la pertinencia del integrador reversible de división hamiltoniana jerárquica

La integración de las ecuaciones de movimiento en simulaciones de N -cuerpos hace uso de integradores simplécticos los cuales consisten en un mapeo canónico, en el lenguaje de ecuaciones diferenciales se les llama integradores geométricos, el integrador de 2o orden Leapfrog con paso constante es un ejemplo de éstos. El teorema Kolmogorov-Arnold-Moser (KAM) garantiza estabilidad del integrador ante perturbaciones originadas por el orden del método o por el truncamiento o redondeo de la computadora.

El gran rango dinámico en simulaciones cosmológicas requiere del uso de integradores con paso adaptivo e incluso una jerarquía de pasos, sin embargo, los integrado-

res simplécticos pierden sus propiedades cuando utilizan pasos de tiempo adaptivos. A pesar de lo anterior se utiliza ampliamente el integrador Leapfrog en su versión *Kick-Drift-Kick* (KDK) con pasos variables y jerárquicos (AKDK) en simulaciones cosmológicas. Analizamos y caracterizamos la pertinencia del integrador de división hamiltoniana jerárquica (HHS) para el caso de simulaciones no-colisionales. Aunque no es un integrador simpléctico mostramos numéricamente que es aproximadamente reversible respecto a las velocidades aunque no simétrico ante inversiones temporales, en contraste con AKDK el cual no tiene ninguna de las dos simetrías. La extensión del teorema KAM para sistemas reversibles [114] explica la estabilidad de HHS y el por qué arroja mejor conservación de energía respecto al integrador tradicional AKDK (KDK con paso de tiempo adaptivo) y conserva además los momentos lineal y angular a precisión de la computadora. Una situación similar se da en el campo de dinámica molecular, en química, con el código RESPA. Además, mostramos las posibles ventajas en el rendimiento de HHS en códigos de suma directa, P3M, y en GADGET-4 (Árbol y FMM). Como se ilustra en las siguientes secciones del capítulo, este tipo de estrategias puede mejorar el rendimiento de diferentes tipos de códigos. Por todo lo anterior, HHS representa una opción competitiva para usar en simulaciones cosmológicas.

ASSESSING THE HIERARCHICAL HAMILTONIAN SPLITTING INTEGRATOR FOR COLLISIONLESS N -BODY SIMULATIONS

G. Aguilar-Argüello,¹ O. Valenzuela,¹ H. Velázquez,¹ J.C. Clemente,¹ and J. A. Trelles¹

Received ; accepted

RESUMEN

El gran rango dinámico en algunos problemas astrofísicos de N -cuerpos ha llevado al uso de pasos de tiempo múltiples adaptivos, sin embargo, la búsqueda de estrategias óptimas es aún un reto. Estudiamos numéricamente el integrador Hierarchical Hamiltonian Splitting (HHS) utilizando un código de suma directa y lo comparamos con el rendimiento de leapfrog con paso global constante y su versión multi-paso adaptivo (AKDK). Encontramos que HHS es aproximadamente reversible, mientras que AKDK no. Por lo que es posible encontrar una combinación de parámetros tales que el cambio de energía es considerablemente menor para HHS, resultando en una mayor eficiencia. Concluimos que HHS es una alternativa competitiva con ventaja para códigos de suma directa y P3M. También, hallamos ventajas para la implementación de HHS en GADGET4 (Árbol/FMM) que merecen ser investigadas más.

ABSTRACT

The large dynamic range in some astrophysical N -body problems led to the use of adaptive multi-time-steps; however, the search for optimal strategies is still challenging. We numerically quantify the performance of the hierarchical Hamiltonian Splitting (HHS) integrator for collisionless simulations using a direct summation code. We compare HHS with the constant global time-step leapfrog integrator, and with the adaptive one (AKDK). We find that HHS is approximately reversible, whereas AKDK not. Therefore, it is possible to find a combination of parameters where the energy drift is considerably milder for HHS, resulting in a better performance. We conclude that HHS is an attractive alternative to AKDK, and it is certainly advantageous for direct summation and P3M codes. Also, we find advantages with GADGET4 (Tree/FMM) HHS implementation that are worth exploring further.

Key Words: software: simulations — gravitation — celestial mechanics — methods: numerical — galaxies: kinematics and dynamics — cosmology: large scale structure

1. INTRODUCTION

Historically, fully self-consistent realistic astrophysical N -body simulations are a challenging problem (Aarseth 1971; Efstathiou et al. 1985; Stadel 2001; Springel et al. 2001; Dehnen & Read 2011; Klypin 2018). On the purely gravitational case, direct summation N -body codes suffer from a bottleneck due to the computational cost of force calculation on a particle-by-particle basis, which scales as

the square of particle number (N_p^2). For this reason, they are primarily used for simulating dense stellar environments or planetary systems. Such limitations triggered the development of sophisticated approximated hybrid collisionless methods, like the TreePM (Tree-Particle Mesh)/P3M (Particle Particle- Particle Mesh)/P3T (Particle Particle- Particle Tree) codes (Xu 1995; Bode et al. 2000; Bagla 2002; Bode & Ostriker 2003) where the short-range component of the force is carried out either by expensive/accurate direct summation or Tree force solvers (Couchman 1991; Oshino et al. 2011; Habib

¹Instituto de Astronomía, Universidad Nacional Autónoma de México, A.P. 70-264, 04510, México, CDMX, México

et al. 2013). Alternatively, AMR (Adaptive Mesh refinement) methods are used to compute the large dynamical range of the gravitational force (Villumsen 1989; Jessop et al. 1994; Kravtsov et al. 1997; Teyssier 2002), seeking a balance between accuracy and computational efficiency. N -body simulations require both, a fast way to calculate the accelerations and an accurate and efficient integration method to evolve particles in time. The second-order leapfrog symplectic integrator (Verlet 1967) is the most widely used in collisionless N -body simulations (e.g. Klypin 2018; Angulo & Hahn 2021). It is strictly symplectic when a global-constant time-step is adopted; however, this is not suitable for addressing problems with a large dynamical range that are currently studied with modern codes. In the quest to improve efficiency, it is necessary to adopt multiple or adaptive time-steps. The general problem of geometric/symplectic (preserving phase space volume), time-symmetric (recover initial conditions after changing dt for $-dt$) and reversible integrators (recover initial conditions after changing the sign of velocities) has been addressed in the field of differential equations for dynamical systems (Hairer et al. 2002). In such work, they point out the differences between adaptive global time-steps or multi-time-steps (several rates of evolution for different parts of the system) and discuss the constraints required for the integration method and the time-step selection function to preserve the mapping properties. In astronomy, the influential study of Quinn et al. (1997) discusses different operator-based leapfrog implementations using time step blocks plus a time-step selection function in the KDK/DKD leapfrog integrator for massive N -body simulations. They pointed out that particle migration across time-step blocks may break up the symmetry and sometimes involves backward integration, which can be difficult to reconcile with a dissipative component like gas. Current collisionless simulations codes commonly use the KDK leapfrog implementation with adaptive time-steps (e.g. Quinn et al. 1997; Springel 2005; Dehnen & Read 2011; Klypin 2018), which we call hereafter AKDK. Recently, Dehnen (2017) discusses conditions where the Hairer et al. (2002) analysis for time-symmetric integrators can be extended to discrete time-stepping. They conclude that there is not a general solution. Many of the proposed integrators truly preserve the symmetries; however, the specific time-step selection function should also respect symmetrization. In several cases, the computational overhead makes the proposal impractical.

In this paper, we explore and quantify the Hier-

archical Hamiltonian Splitting (HHS) strategy proposed by Pelupessy et al. (2012), which is, as leapfrog, a second-order scheme. This integrator was tested for small number of bodies or collisional simulations, delivering good energy conservation and momentum conservation at machine accuracy. However, no analysis of time-symmetry or reversibility was presented. Some modern N -body codes like AREPO (Weinberger et al. 2020) and GADGET4 (Springel et al. 2021) have implemented versions of HHS with some differences with respect the original proposal, although they do not give details of the performance or other properties that allow the comparison with the commonly used integrators. In this work, we extend the discussion of HHS in the context of collisionless N -body simulations, by numerically investigating the time-symmetry and velocity reversibility. We, also, test some time-step selection functions to explore the potential advantages. In all cases we compare with the global constant time-step leapfrog integrator and the adaptive one (AKDK) in order to asses the conditions under which HHS is a competitive alternative. As we discussed above, in some modern P3M codes running in hybrid architectures, the most expensive calculation is the short-range direct summation force integration, in some cases processed inside GPUs (Habib et al. 2016). Motivated by that we implemented HHS in a direct summation code running in GPUs and complement that with additional tests with the Tree/FMM code GADGET4.

The rest of the paper has been organized as follows: section 2 summarizes the main properties of the integrators used here to carry out their comparison while section 3 introduces the time-step selection functions. Section 4 shows accuracy tests performed with emphasis on the cases on an isolated halo and a minor merger. Sections 5, 6 and 7 contain the results of these tests taking into account the effect of time-step functions, performance and long-term stability, respectively. In section 8, we quantified reversibility and time-symmetry for the different codes. Finally, a discussion and main conclusions are given in section 9.

2. INTEGRATORS

We implemented three different integrators in a direct summation code dubbed as $\text{NP}_{\text{splitt}}$ (Aguilar-Argüello et al. *in prep.*), the leapfrog, Adaptive-KDK (AKDK) and the Hierarchical Hamiltonian Splitting integrators (HHS). Below, we describe each integrator. It is common to express integrators as a composition of operators using the Hamiltonian

splitting technique in potential (*Kick*) and kinetic energy (*Drift*), although there are other possibilities (Oshino et al. 2011).

2.1. Leapfrog

The leapfrog integrator is a second-order widely used integrator. As mentioned previously, this integrator is strictly symplectic only when a global-constant time-step is adopted. Symplectic integrators are designed to numerically preserve the integrals of motion and the phase-space volume of the simulated system.

In the leapfrog method, the evolution of the gravitational system can be written as a sequence of *Kick* (advance of velocities) and *Drift* (advance of positions) operators (e.g. Channell 1993; Quinn et al. 1997), defined by:

$$\begin{aligned} K(dt) : \mathbf{v}(t_n + dt) &= \mathbf{v}(t_n) + dt \mathbf{a}(t_n) \\ D(dt) : \mathbf{x}(t_n + dt) &= \mathbf{x}(t_n) + dt \mathbf{v}(t_n) \end{aligned} \quad (1)$$

where \mathbf{x} , \mathbf{v} and \mathbf{a} are the position, velocity and acceleration of a particle, respectively, and dt is the time step. In this paper, we use the operator sequence called KDK leapfrog (also known as *velocity Verlet*, Swope et al. 1982):

$$KDK : K(dt/2) D(dt) K(dt/2) \quad (2)$$

where we considered that the evolution is for one time step, i.e. from t_n to $t_n + dt$. Through this paper, we will refer to KDK leapfrog with a global-constant time-step as the Leapfrog integrator.

2.2. AKDK

Contemporary codes have extensively used KDK (eq. 2) combined with a block time-step scheme (Hayli 1967; Sellwood 1985; McMillan 1986; Hernquist & Katz 1989; Makino 1991), frequently using rungs which are power of two: $dt_r = dt_0 2^{(-r)}$ and different assigning time-step functions, most frequently an acceleration based one (Springel 2005). We will use it as a reference integration scheme, but it should be noted that it is not symplectic (Hairer et al. 2002) and that the block-step is a multi time-step scheme.

2.3. Hierarchical Hamiltonian Splitting

The hierarchical Hamiltonian Splitting (HHS) method is a second-order integrator that uses individual time steps of the particles (Pelupessy et al. 2012) through recursively splitting the Hamiltonian. It accurately preserves linear and angular momentum and has a good energy conservation.

This integrator consists of adaptively and recursively splitting the Hamiltonian as a function of the current time step, dt , so that the so called *Slow* system (hereinafter S) contains all the particles with a time step larger than dt , and the so called *Fast* system (hereinafter F) contains all the particles with a time step smaller than dt . Thus, the splitting is as follows:

$$\begin{aligned} H_S &= T_S + V_{SS} + V_{SF} \\ H_F &= T_F + V_{FF} \end{aligned} \quad (3)$$

where,

$$\begin{aligned} T_X &\equiv \sum_{i \in X} \frac{p_i^2}{2m_i} \\ V_{XX} &\equiv -G \sum_{i \in X} \sum_{j \in X, j > i} \frac{m_i m_j}{|r_i - r_j|} \\ V_{XY} &\equiv -G \sum_{i \in X} \sum_{j \in Y} \frac{m_i m_j}{|r_i - r_j|} \end{aligned} \quad (4)$$

are, respectively, the kinetic and potential energies, and V_{SF} is the potential energy of the interactions between S and F particles. The previous splitting scheme is known as HOLD (since it ‘‘holds’’ V_{SF} for evaluation at the slow time-step, Pelupessy et al. 2012).

The S system is solved using the DKD scheme (also known as *position Verlet*, Tuckerman et al. 1990), which consists of *drifts* of the particles in this system (due to T_S) and *kicks* on the particles of both systems (due to $V_{SS} + V_{SF}$). For the F system, the same procedure as for the original system is applied but using a halved time-step. Hence, the splitting is applied recursively to the F system with time-step $dt/2^r$. The recursion ends when the system F (of the rung r) has no particles. At the end of the current integration step, the new time-step of a particle is calculated. In this scheme, a particle can change its time step to higher (lower) value if its current integration time is synchronized with the higher (lower) rung.

It is well known that the *Kick* and *Drift* operators are symplectic, however, using multiple or adaptive time-steps may not preserve such properties in a general way (Hairer et al. 2002). Therefore we need to investigate the behaviour of HHS.

3. TIME-STEP SELECTION FUNCTION

Besides the formulation of integrators with individual time steps based on symmetric operators, the choice for each particle time step is made through the so called time-step selection function. There is not a unique choice, arguably the most commonly used

time-step function in contemporary collisionless N -body codes (e.g. GADGET, Springel 2005) is based on the acceleration as:

$$\tau_i = \eta_{\text{accel}} \sqrt{\frac{\epsilon}{a_i}} \quad (5)$$

where a_i is the acceleration acting on the particle i , giving the code the possibility of adapting to high/low accelerations and ϵ is the force plummer softening². Improvements have been recently discussed by using a dynamical time proxy (Zemp et al. 2007) and a tidal force time scale (Dehnen & Read 2011; Grudić & Hopkins 2020), establishing a balance between short and long time steps, which may translate into higher efficiency. Extensive comparisons of AKDK with both choices have been discussed in Zemp et al. (2007) and Grudić & Hopkins (2020).

In our study, in an attempt to preserve the energy stability of the HHS integrator while allowing adaptive multi time-steps, and following Pelupessy et al. (2012), we use an approximated time-symmetrized time-step extrapolation criterion for each particle. To obtain such time-step criterion, we start from the implicit criterion (Hut et al. 1995):

$$\tau_{\text{sym}} = \frac{1}{2} [\tau(t) + \tau(t + \tau_{\text{sym}})] \quad (6)$$

where τ is a time-step function (non-symmetrized), and τ_{sym} is the symmetrized time-step function of τ . To a first-order perturbative expansion:

$$\tau(t + \tau_{\text{sym}}) \approx \tau(t) + \frac{d\tau}{dt} \tau_{\text{sym}} \quad (7)$$

hence, from eq. 6:

$$\tau_{\text{sym}} \approx \tau(t) + \frac{1}{2} \frac{d\tau}{dt} \tau_{\text{sym}} \quad (8)$$

so that, the time-step we will use is given by (Pelupessy et al. 2012),

$$\tau_i = \min_j \left[\frac{\tau_{ij}}{\left(1 - \frac{1}{2} \frac{d\tau_{ij}}{dt}\right)} \right] \quad (9)$$

It is important to state that the minimization indicated above is across the so called *Slow* particles.

²We will adopt the softening as twice the average inter-particle distance at the minimum radius where the density profile is not dominated by Poisson fluctuations, as it is usually adopted in collisionless simulations.

For a time-step proportional to the inter-particle free-fall time:

$$\begin{aligned} \tau_{ij} &= \eta_{\text{FF}} \sqrt{\frac{r_{ij}^3}{G(m_i + m_j)}} \\ \frac{d\tau_{ij}}{dt} &= \frac{3\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}}{2r_{ij}^2} \tau_{ij} \end{aligned} \quad (10)$$

The former option is a two-body-based proxy for the dynamical-time-motivated step function suggested by Zemp et al. (2007).

For completeness with Pelupessy et al. (2012), for a time-step proportional to the inter-particle fly-by time (typically used in collisional problems):

$$\begin{aligned} \tau_{ij} &= \eta_{\text{FB}} \frac{r_{ij}}{v_{ij}} \\ \frac{d\tau_{ij}}{dt} &= \frac{\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}}{r_{ij}^2} \tau_{ij} \left(1 + \frac{G(m_i + m_j)}{v_{ij}^2 r_{ij}} \right) \end{aligned} \quad (11)$$

We will quantify the efficiency of such time-step functions, however, the high acceleration derivatives in the case of collisional problems may require going beyond the first order in the perturbative expansion.

Across the paper, we will mostly use the approximated symmetric free-fall time-step for HHS (defined by eqs. 9 and 10), and only in a few tests we will use the minimum of this and the approximated symmetric fly-by time-step (eqs. 9 and 11). In section 5, we will present a comparison with the GADGET4 implementation of HHS (Springel et al. 2021), this code uses a time-step function similar to eq. 5 but the accuracy parameter, η_{accel} , is included inside the square root. For AKDK, we will use the standard time-step function given by eq. 5.

Table 1 summarizes the combinations of integrators and time-step selection functions used through this work.

4. ACCURACY TESTS

In this section, we present the test results of the HHS algorithm in terms of accuracy by simulating an isolated halo and sinking satellites, and compare them with the global-constant time-step leapfrog and the adaptive one, AKDK. To proceed with the comparison we implemented the different integrators in a direct summation N -body code (Aguilar-Argüello et al. *in prep.*). All the experiments were run in a single GPU. As a sanity check, we performed binary system tests (not reported here) and the results are consistent with those reported in other works (e.g. Dehnen & Read 2011; Pelupessy et al. 2012; Springel 2005).

TABLE 1
COMBINATIONS OF INTEGRATORS AND TIME-STEP FUNCTIONS USED ACROSS THIS PAPER.

Integrator name	Integration scheme	Time-step scheme	Time-step selection function	Reference
Leapfrog	KDK	Global-constant		Verlet (1967)
AKDK	KDK	BLOCK	eq. 5	e.g. Quinn et al. (1997); Springel (2005) Dehnen & Read (2011); Klypin (2018)
HHS	HHS	HOLD	eqs. 9, 10	Pelupessy et al. (2012)
nsHHS	HHS	HOLD	eqs. 9, 10 (without $d\tau_{ij}/dt$)	This work
sAKDK	KDK	BLOCK	eqs. 9, 10	This work

4.1. Isolated Cuspy Halo

We adopted as a reference model an equal particle mass, isolated halo following the NFW cuspy density profile predicted by collisionless dark matter cosmological simulations (Navarro et al. 1997). The large density range and the corresponding different dynamical times make it a suitable system for an adaptive time-step code. Such tests depend on resolution to actually capture the benefit of individual time-steps as compared with a global-constant time-step scheme. We will use as a reference time scale the dynamical time³ at the NFW characteristic radius (r_s), since it has been used to study the stability of the halo in other works (e.g. Klypin et al. 2015).

For the integration of our fiducial model, we adopted $G = 1$ (gravitational constant), $M_{\text{vir}} = 1$ (virial mass) and $r_s = 1$ (scale length, also called characteristic radius), as model units. We will use these model units through the paper.

To investigate and quantify differences in accuracy and performance between the integrators, first, we followed a fiducial halo sampled with 10^5 particles for 40 dynamical times at r_s , t_{dyn} .

Because our implementations of AKDK and HHS have different time-step function, a meaningful comparison is to assume an energy conservation threshold, which implies using distinct accuracy parameters for both integrators. For the first test, we considered a 10^{-7} threshold and accuracy parameters $\eta_{\text{FF}} = 0.003$ and $\eta_{\text{accel}} = 0.01$ for HHS and AKDK, respectively, both constrained to 6 time-step rungs. Figure 1 shows the result of these tests. The upper left panel shows the energy error. HHS (black) stays very close to Leapfrog (red) during the first $20 t_{\text{dyn}}$,

afterwards it shows a small drift. AKDK (blue) drifted almost linearly and after $20 t_{\text{dyn}}$ it slightly flattens. Because the main computational overhead of HHS over AKDK comes from building and update the time-step hierarchy in HHS, we decided to explore experiments where we delayed such an update, and denoted by HHS-sTSS. We observed that such an action results in important savings in computational time (yellow line). The energy accuracy test is lower but acceptable for a collisionless simulation and it is faster; in addition linear and angular momentum are preserved to machine precision.

Regarding the conservation of other dynamical quantities like linear and angular momentum or the system barycenter the situation is different (see figure 1). Leapfrog and HHS preserve almost at machine precision the linear and angular momentum (see bottom panels), whereas AKDK presents smaller accuracy, although it has a slope until $10 t_{\text{dyn}}$, afterwards it flattens. Interestingly, HHS with a delay in updating the time-step hierarchy (HHS-sTSS, yellow) is almost indistinguishable from HHS and leapfrog. The upper right panel shows the accuracy in preserving the halo centroid. Once again, leapfrog and both HHS versions accurately keep the centroid, while AKDK accuracy degraded two orders of magnitude. The wall-clock time for leapfrog, HHS and HSS-sTSS experiments was, respectively, 1.7, 0.8 and 0.6 times the corresponding for AKDK. As a complement, we performed tests using larger time steps reaching lower energy accuracy, the general results are the same. Finally, we emphasize that all integrators accurately preserve the density profile, as we can see in figure 2; with obvious dependence on the number of particles, we decided to show the test with two million particles in order to minimize discreteness effects.

³A dynamical time, also called crossing time, is the time taken for a typical particle to cross the system. In this paper, a dynamical time is defined as $t_{\text{dyn}}(r) = [r^3/GM(r)]^{1/2}$, where r and $M(r)$ are the radius and mass, respectively.

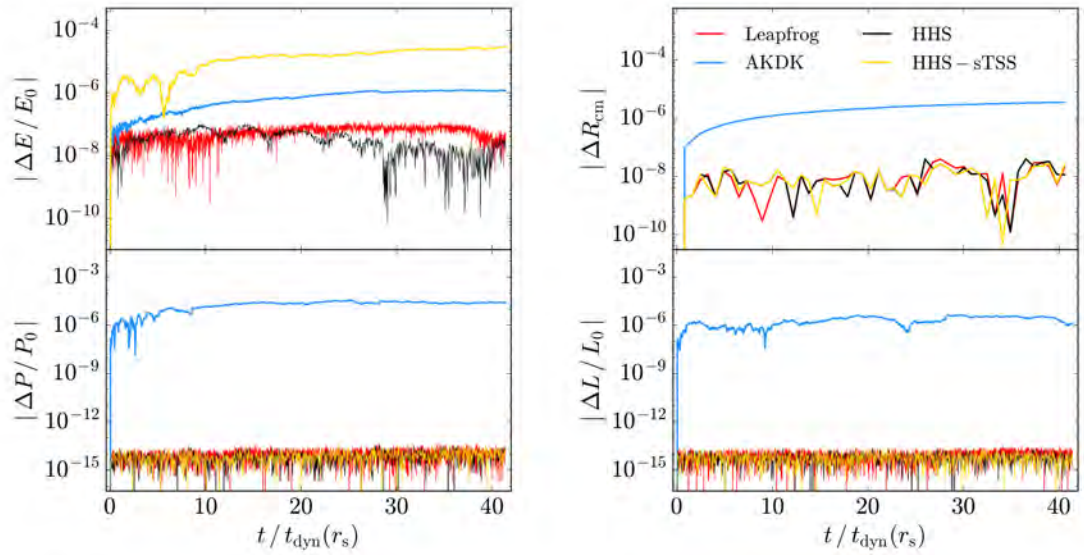


Fig. 1. Error in conserved quantities for an isolated NFW halo with $N = 10^5$ particles, simulated up to 40 dynamical times (at scale radius, r_s). Shown is the energy error (upper left panel), change in center of mass position (upper right panel), error in linear (lower left panel) and angular momentum (lower right panel) for the three integrators: Leapfrog (red), AKDK (blue) and HHS (black). Also, it is shown the HHS with a delay in the time-step hierarchy update (HHS-sTSS, yellow) version.

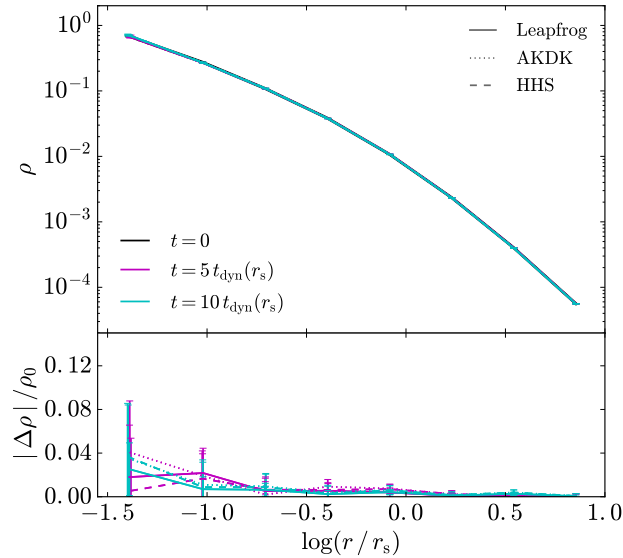


Fig. 2. Density profile, at different times, of the isolated NFW halo with 2×10^6 particles simulated up to 10 dynamical times for the three integrators: Leapfrog (solid lines), AKDK (dotted lines) and HHS (dashed lines). The three integrators accurately preserve the density profile.

4.2. Minor Merger: Sinking Satellites

Satellite accretion onto larger galaxies is an astrophysical problem commonly simulated by both, isolated and cosmological N -body simulations (Miller et al. 2020; Arca-Sedda & Capuzzo-Dolcetta 2016). We simulated a satellite, represented by a softened and massive particle, falling into a spherical system, represented by collisionless softened particles, for seven dynamical times. The spherical system consists of $N = 10^5$ equal mass particles spatially distributed according to the NFW density profile (Navarro et al. 1997). We adopted the same units as the previous fiducial isolated halo case. We used Plummer softening and we chose a softening parameter $\epsilon = 0.026$ (in model units). The satellite's initial separation from the center of the spherical system is $R_{\text{sat}} = 2.6$, and its mass is $m_{\text{sat}} = 0.01$, which is ~ 1300 times bigger than the mass of one collisionless particle.

This test is particularly useful because the sinking process involves orbital angular momentum and energy transfer into the host system. We tracked energy, linear and angular momentum conservation as well as the host center of mass behaviour (see figure 3). Leapfrog (red) stays flat. HHS (black) starts to jump at one dynamical time, afterwards it stays flat. AKDK (blue) shows a lower accuracy in energy conservation and presents a systematic energy growth.

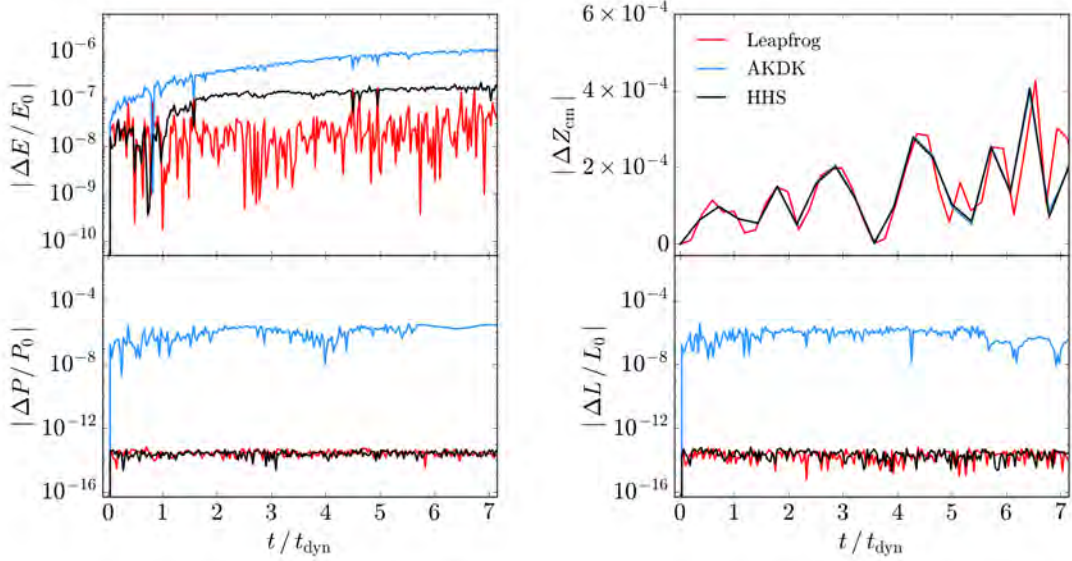


Fig. 3. Error in conserved quantities for a NFW halo with $N = 10^5$ particles plus a satellite simulated up to 7 dynamical times. Shown is the energy error (upper left panel), change in center of mass position (upper right panel), error in linear (lower left panel) and angular momentum (lower right panel) for the three different integrators: Leapfrog (red), AKDK (blue) and HHS (black).

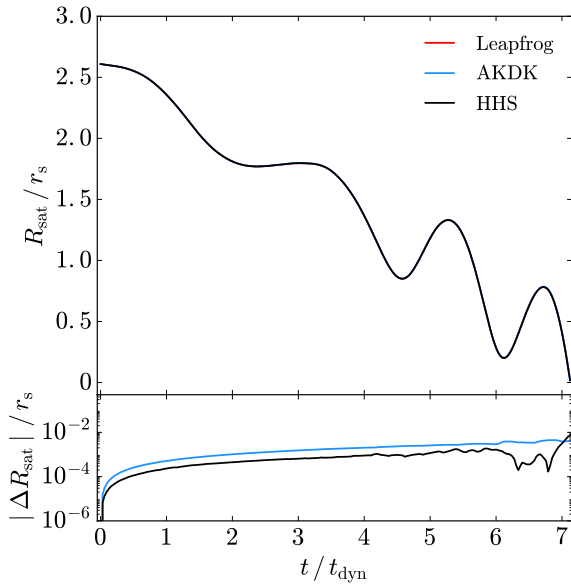


Fig. 4. Evolution of the satellite radial position (upper panel) for the sinking satellite test and for the three integrators: leapfrog (red), AKDK (blue) and HHS (black). When comparing with leapfrog (lower panel), the differences are below 1% between integrators.

As in the case of the isolated halo, HHS energy drift slope is considerably flatter than the corresponding to AKDK. Linear and angular momentum are less accurate for AKDK for several orders of magnitude, while system barycenter behaves essentially the same for all three integrators. These results indicate that HHS is an excellent alternative for dynamical friction studies.

The evolution of the satellite radial position shows differences below the one percent level (see figure 4). We conclude that, for a reduced number of dynamical times, the three integrators can provide an accurate description of the sinking process.

5. TIME-STEP SELECTION FUNCTION TESTS

As it has already been discussed some hierarchical/adaptive time-step integrators, like HHS and AKDK, include a time-step selection function, such a prescription may help restoring the integrator symmetry. We may question if the approximated symmetric time-step selection function based on particle pairs given by eq. 9 is only useful for a particular kind of simulation or code, and if we loose all the convenient properties of HHS, observed at this point, when the problem is not tractable by a direct summation code. To quantify such effect we evolved the fiducial isolated halo switching the time-step selection function and we additionally performed test with the Tree/FMM code GADGET4.

5.1. Direct summation code

First we consider our direct summation code implementation, simulating the fiducial isolated halo considering: HHS and AKDK integrators, as it can be seen in the upper panel of figure 5. We take as a reference the AKDK integrator with an accuracy parameter $\eta_{\text{accel}} = 0.08$ (fiducial, solid blue line). The dotted black line shows HHS (with $\eta_{\text{FF}} = 0.055$), which is almost twice faster but slightly less accurate, and the dashed black line shows a case of HHS (with $\eta_{\text{FF}} = 0.015$) remarkably more accurate but 10% slower than the fiducial AKDK. We include another AKDK case (dashed blue) with a smaller accuracy parameter $\eta_{\text{accel}} = 0.049$, which is 20% less accurate than HHS (dashed black line) but 10% slower, and also is 20% slower and an order of magnitude more accurate than the fiducial AKDK case (solid blue). This means that considering even smaller values for the parameter η will not make AKDK more efficient than HHS. Next, we perform some changes in the time-step selection function, we removed the derivative term from eq. 9 which represents the symmetrizing correction for the HHS integrator, we dubbed such tests nsHHS and they appear as the magenta lines in figure 5, the energy drift is larger as compared with HHS but it is still acceptable for collisionless simulations and it is faster. For completeness, we performed tests using eq. 9 in AKDK (dubbed as sAKDK, green lines) with the same parameters (η_{FF} , rungs, etc.) as in HHS tests. The energy drift is larger for the sAKDK test than the one corresponding to HHS, however sAKDK is faster. Note that it is possible to match HHS accuracy by lowering the η_{FF} parameter in sAKDK tests, but it is slower (we do not show it because it does not add new information). So far, our tests suggest that HHS may benefit hybrid codes that use direct summation force calculation as part of their algorithm, for example in the P3M technique (Hockney & Eastwood 1998) the most expensive part corresponds to direct summation, which runs in accelerators like GPUs (Habib et al. 2013, 2016; Cheng et al. 2020). Even using different time-step selection function, HHS may still obtain considerable performance. We analyze such a situation in the following sections, as well as, HHS performance in codes using approximated force computations (e.g. a tree code).

5.2. Tree Code: GADGET4

Recently, the 4th version of the publicly available code GADGET has implemented the HHS in the so called hierarchical gravity mode (Springel et al. 2021). The time-step selection function is similar to

our equation 5 but the accuracy parameter η_{accel} is inside the square root. Changing the step function is a sensible choice because our equation 9, based on pair interactions, is not practical for very large number of particles. Also, in the GADGET4 HHS implementation, instead of using the DKD representation (as in our implementation), the authors adopted the KDK one (for further details we refer to Springel et al. 2021), in a similar way as Zhu (2017). Hence, this allow us to explore the case of an approximated gravitational acceleration code, such as tree code (e.g. Barnes & Hut 1986), with different time-step selection function. For that purpose, we perform some tests using AKDK and HHS with the Tree version of GADGET4. As in the case of direct summation tests (section 5.1), we simulated the fiducial isolated halo in such a way we can directly compared with the direct summation code tests, including a non-symmetrized time-step function. Middle panel of figure 5 shows the results for the Tree code version of GADGET4. For the corresponding fiducial AKDK (solid blue line), GADGET4 opens three time-step rungs, and for its HHS implementation (dotted black line), it opens only two time-step rungs. Both integrators show an energy drift similar to 10^{-3} , however, HHS (dotted black line) is 66% faster but less accurate. Motivated for such result, we run a new HHS test decreasing the η_{accel} parameter (dashed black line), the energy is considerably flat, therefore energy conservation after $40 t_{\text{dyn}}$ is almost an order of magnitude better than the fiducial AKDK (solid blue line), but it is 7% slower. If we decrease η_{accel} for the AKDK integrator in order to match energy conservation (dashed blue line), the wall-clock time is considerably larger than 50%. Although this is a particular example model, it is consistent with our previous tests. HHS is more stable and at medium and long-term integrations may be more efficient than AKDK, regardless of not using the approximated symmetric free-fall particle pairs time-step function (eqs. 9 and 10).

5.3. Fast Multipole Method Code: GADGET4

Gadget4 has implemented a different gravity solver based on the Fast Multipole Method expansion (FMM, e.g. Dehnen 2000, 2002). We run the same fiducial isolated halo, as in previous section, adopting the FMM scheme truncating the expansion until the quadrupole term. Results are presented in the lower panel of figure 5. Blue solid line shows the fiducial case using AKDK ($\eta_{\text{accel}} = 0.01125$), the corresponding HHS case ($\eta_{\text{accel}} = 0.01125$, dotted black line) is relatively faster (66%), however,

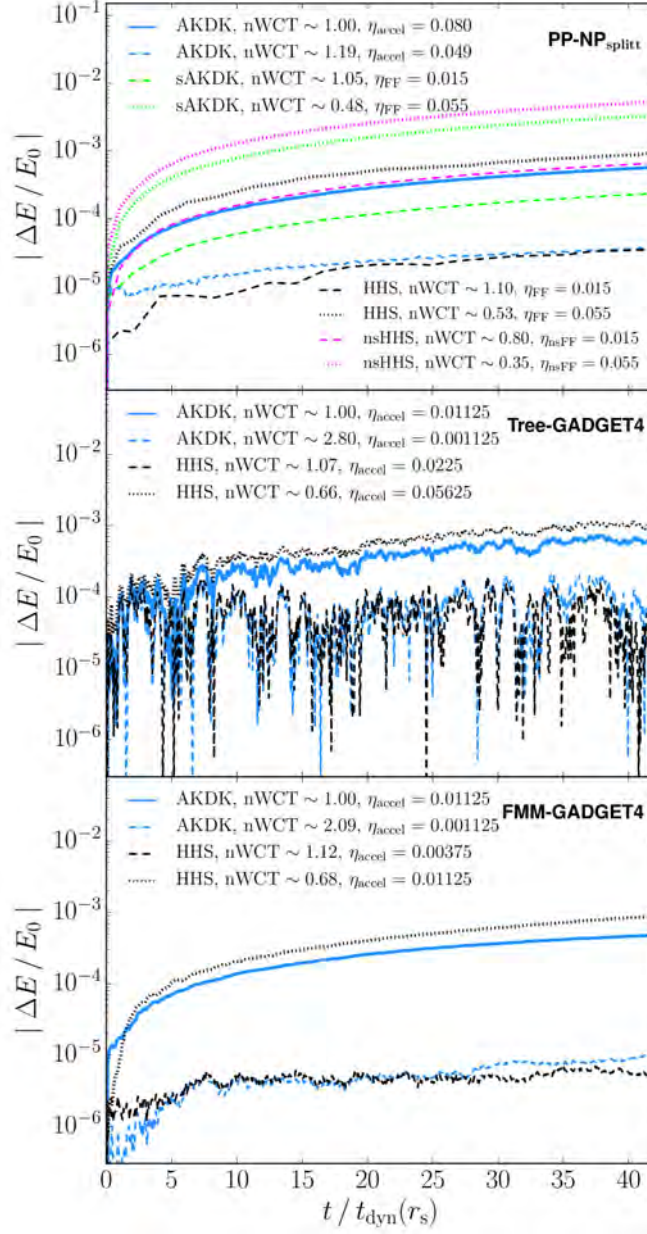


Fig. 5. Effect of time-step selection function and different acceleration codes. nWCT corresponds to the wall-clock time of each test normalized to the fiducial AKDK (solid blue line). Upper panel shows tests with our direct summation code. Solid blue line is the fiducial AKDK test and dotted black line is HHS, almost 50% faster. Dashed lines correspond to experiments with smaller accuracy parameter but reaching a limit where HHS is faster and more accurate than AKDK. Magenta dotted/dashed lines show the same HHS tests but neglecting the symmetrizing derivative term in eq. 9, even going with smaller η_{nsFF} still may be competitive with AKDK. Green dotted/dashed lines show AKDK test with the symmetrized time-step selection function (eqs. 9 and 10) using the same parameters as HHS cases (black dotted/dashed lines, respectively). Middle panel shows experiments with GADGET4 using the Tree version. Results are consistent with the direct summation code. Solid blue line is the fiducial AKDK, dotted black line is HHS 7% slower and slightly less accurate. Dashed lines are AKDK and HHS, with almost flat behaviour, slightly better for HHS, however, AKDK is almost three times slower. Lower panel shows the equivalent tests but now for FMM GADGET4. As before, the AKDK for the flat case (dashed blue) is slightly worse in energy accuracy and almost twice as slow as compared with HHS (dashed black). We conclude that, even for different time-step functions, there is still a regime where HHS is more efficient.

it is less accurate. We experimented lowering the η_{accel} parameter for HHS, the energy evolution is flat (dashed black line). We also decrease η_{accel} for AKDK (dashed blue line), the energy accuracy is 10% worse but the wall-clock time is almost 100% larger; therefore, there is no point trying smaller η_{accel} values. The experiments with GADGET4 FMM version also confirm the conclusions from our direct summation tests.

6. PERFORMANCE

Although the adaptive nature of the HHS and AKDK integrators may imply a higher efficiency compared with the global-constant time-step leapfrog, the benefit of taking adaptive time-steps is evident only when the dynamical range is large. In comparison with AKDK, HHS has a computational overhead due to the recursive splitting of the Hamiltonian needed to build the time-step hierarchy.

We decided to make a short exploration of the simulation parameters (accuracy, rung number, and minimum time-step) with our direct summation code. We present the results in figure 6, which shows a pragmatic diagnostic of the integrator performance: energy conservation vs. wall-clock time. The upper and middle panels correspond to the fiducial isolated halo. In the upper panel we fixed the number of time-step rungs and we varied the minimum time-step; whereas in the middle panel, we fixed the minimum time-step (same for both integrators) and we varied the number of time-step rungs. The lower panel corresponds to the cuspy halo including four live cuspy satellites, a common situation in cosmological simulations that requires a large dynamical range. In many cases HHS outperforms AKDK. We may wonder what is the reason given the extra computations related with the splitting process. One possible suspect is the individual time-step distribution. To investigate that, we built the histogram of time steps for particles in the initial conditions and at later times (see figure 7). For HHS (solid circles) it is clear that only a moderate fraction of particles are found in the deepest time-step rung. For AKDK the distribution is almost the opposite, there is a peak of particles in the three deepest rungs, which translates into many more time-steps than HHS. The difference in performance is partly due to the time-step selection function, in agreement with Zemp et al. (2007); Grudić & Hopkins (2020). Although tweaking parameters we may obtain differences in performance and accuracy, we noticed that for a defined energy conservation threshold in many cases HHS outperforms AKDK,

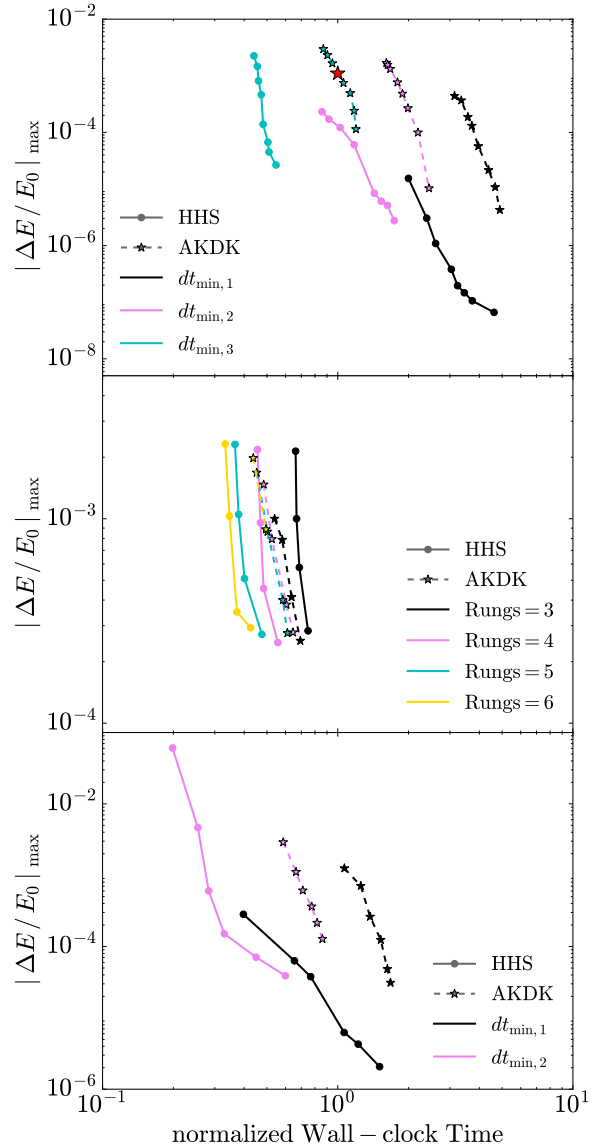


Fig. 6. Energy conservation as a function of wall-clock time (normalized with respect to the red star case in upper panel) for the fiducial isolated halo (upper and middle panels) and for a cuspy halo including four live cuspy satellites (bottom panel), for HHS (solid circles) and AKDK (stars) integrators. In upper panel, we used a plummer softening $\epsilon = 0.007$, and we varied the minimum time step ($dt_{\text{min},1} = 5.4 \times 10^{-3}$, $dt_{\text{min},2} = 1.1 \times 10^{-2}$ and $dt_{\text{min},3} = 2.2 \times 10^{-2}$) but fixing the number of rungs (6 and 5 for HHS and AKDK, respectively). In middle panel, we used $\epsilon = 0.01$, and we varied the number of rungs but fixing the minimum time step ($dt_{\text{min}} = 4.3 \times 10^{-2}$). For the bottom panel, we used $\epsilon = 0.004$, and we varied the minimum time step ($dt_{\text{min},1} = 6.4 \times 10^{-3}$ and $dt_{\text{min},2} = 1.3 \times 10^{-2}$) but fixing the number of rungs (6 and 3 for HHS and AKDK, respectively).

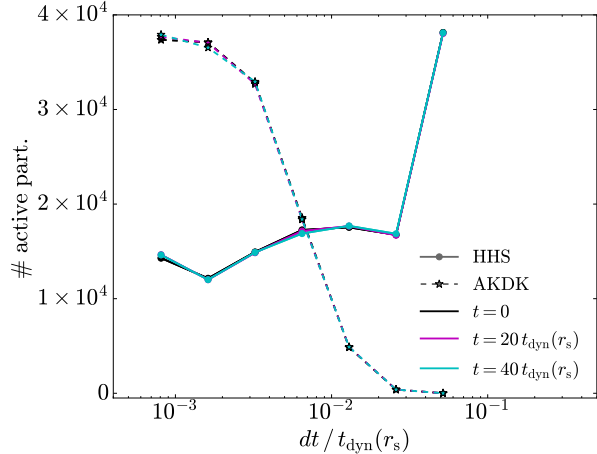


Fig. 7. Particle distribution across time-step rungs for different selection functions: acceleration criteria (eq. 5, stars) and the approximated symmetric free-fall (eqs.9 and 10, solid circles), for the fiducial isolated halo using the same parameters as figure 1. It is notorious that the acceleration criteria has more particles with small time step and that free-fall criteria is almost the opposite, with potential consequences for the performance.

because the energy growth is smaller for HHS than for the standard AKDK implementation, allowing HHS to use larger steps. Nevertheless it is important to mention that if all the particles are similarly distributed in the time-step rungs for both integrators the computational overhead of HHS starts to play a more important role, this may happen for very low resolution runs where the dynamical range is artificially shortened. This is in agreement with the middle panel of figure , showing that only when using more than three step runs HHS is faster than AKDK. The time-step histogram presented in figure 7 is a handy tool to asses the situation.

Our conclusions are consistent with recent studies that published successful results using HHS with GADGET4 and reaching an extremely large dynamical range with multi-million particle numbers (Wang et al. 2020).

7. LONG-TERM STABILITY

At this point we have compared the accuracy and performance of the integrators for some dynamical times. A natural question arises about if HHS advantages are relevant for realistic long-term integration. Dark matter halos survive around 30–200 dynamical times in cosmological simulations depending on the merger/accretion history (Klypin et al. 2015). As before we investigated the stability of energy, linear and angular momentum and density centroid for our fiducial isolated halo model, this time for hundreds

of dynamical times (see figure 8). Energy conservation of HHS (black) is quite close to the global-constant time-step leapfrog (red) behaviour during the first 20 – 40 dynamical times (consistent with our previous tests), however, after that it starts to slowly drift which seems to get smaller at the end of the simulation ($\sim 700 t_{\text{dyn}}$). AKDK (blue) quickly drifts to a considerably larger energy error and keeps systematically growing. The yellow curve represents the HHS-sTSS version that delays the time-step hierarchy updating, as we observed before, it behaves as AKDK but with smaller accuracy, although it is faster ($\sim 40\%$). For linear and angular momentum all integrators preserve such quantities, however, while HHS preserves them almost at machine precision, AKDK preservation is almost 8 orders of magnitude worse. A similar disparity is obtained by following the halo centroid. Seeking for a consequence of this difference in accuracy we tracked the circular velocity in experiments with a smaller number of particles ($N = 2000$) evolving the system for 400 dynamical times at $2.1 r_s$, where circular velocity peaks. Figure 9 shows the circular velocity profile at several times. Despite we find some differences, they are all inside 10%. We conclude that, at the level of high accuracy, we do not expect important differences between integrators and performance is the most relevant difference. As far as N -body simulations reach a larger dynamical range, the smallest structures may live for a larger number of dynamical times, in this context HHS may offer a more stable option.

Recent studies regarding long-term N -body evolution (Hernandez et al. 2020), suggest that long-term integrations may be unstable to small changes in initial conditions realizations, this may be particularly critical for highly non-linear situations like the three-body problem. We generated a small ensemble of realizations for the fiducial isolated halo and for the sinking satellite problem, results are shown in figure 10 only for Leapfrog (red) and HHS (black). Indeed some scatter is found, however, overall the results are robust.

8. REVERSIBILITY AND TIME SYMMETRY

The above numerical experiments show that there are certain parameter combinations where HHS is more accurate than AKDK or, alternatively, it is faster for a given energy accuracy. Because both integrators have different parameters it is natural to ask if there is a true advantage of HHS or if it is a misleading result, dependent on our implementation, accuracy or even particle number.

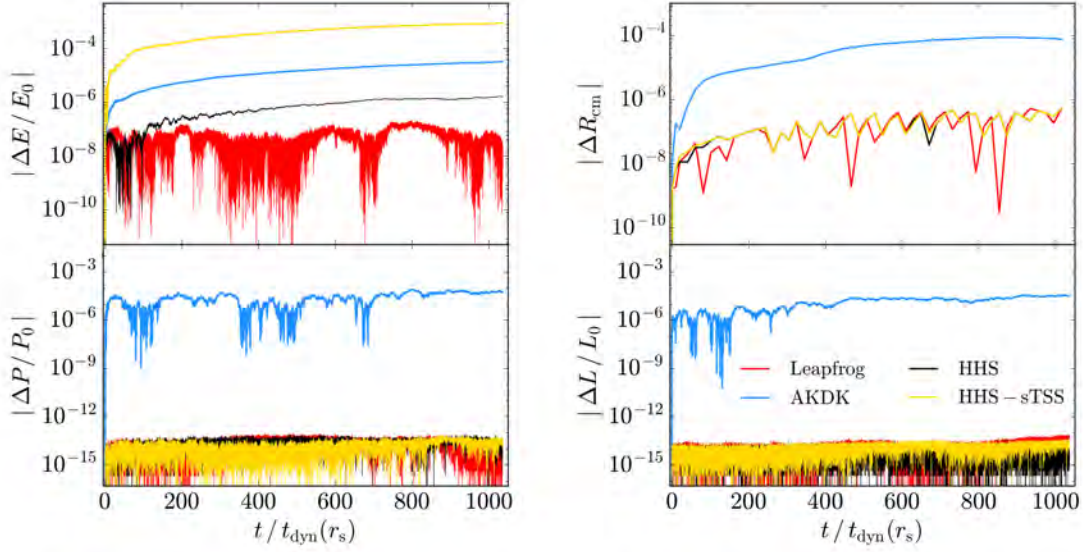


Fig. 8. As figure 1, but following the fiducial isolated halo up to 10^3 dynamical times (at scale radius).

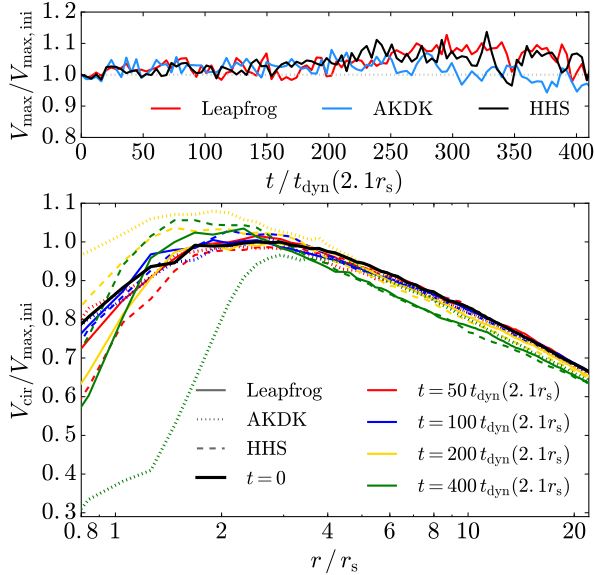


Fig. 9. Circular velocity curves for the fiducial isolated halo with 2000 particles following the system for long integration times. There is not systematic difference between different integrators. The energy conservation level is $\sim 10^{-5}$. The maximum circular velocity is scattered inside 10% at different moments.

As we discussed in the introduction and in agreement with Hairer et al. (2003) we performed time symmetry and velocity reversibility tests using both HHS and AKDK integrators, we used leapfrog with a global-constant time-step as a reference case.

We chose the sinking satellite system as the test bed because the satellite orbit allows easily to track the system response in configuration space as well as in energy. At three different moments, termed BW1, BW2 and BW3, we reversed velocity signs for all particles, and for one case instead of velocities we reversed time sign, after that we continued the integration. Figure 11 shows the global result of forward (FW, solid lines) and backward (BW) evolution (i.e. inverting the sign of velocities) of the satellite distance to the halo center for all integrators either for high (left panels) or low (right panels) energy conservation accuracy ($\sim 10^{-8}$, left panels), the differences in configuration space are in general small, which is consistent with Hernandez & Bertschinger (2018), where they discuss that AKDK preserves quantities like angular and linear momentum (see also figure 3), however, there are still some differences as we can see in the lower panel. The satellite distance change with respect to its own forward evolution (lower panels) is a good diagnostic of reversibility. Clearly, the position difference in HHS (black) regarding the symplectic Leapfrog (red, global constant step) is below the simulation resolution determined by the softening (horizontal grey line), while for AKDK (blue) the calculated position difference is larger but close

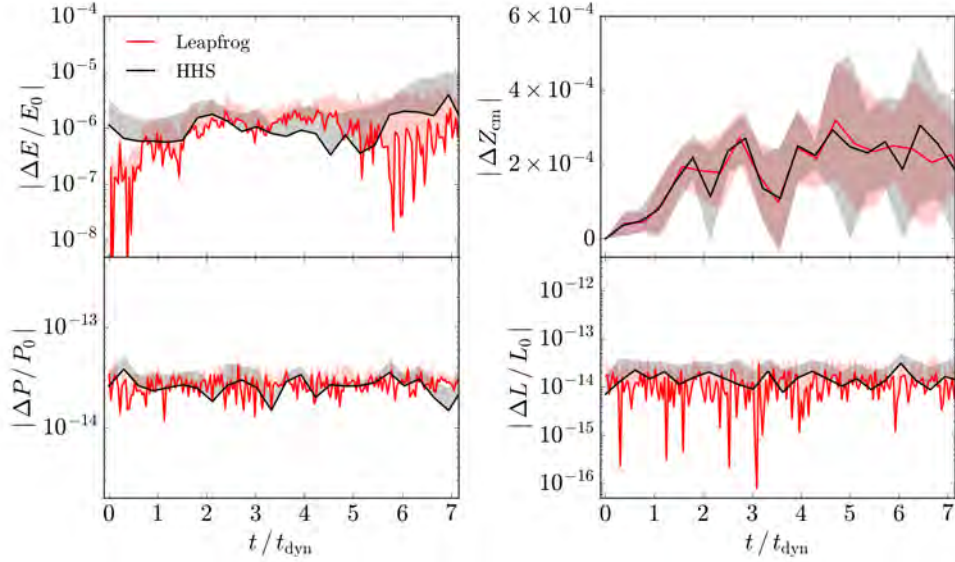


Fig. 10. Error in conserved quantities for a NFW halo with $N = 10^5$ particles plus a satellite simulated up to 7 dynamical times. Shown is the energy error (upper left panel), change in center of mass position (upper right panel), error in linear (lower left panel) and angular momentum (lower right panel) for the Leapfrog (red) and HHS (black) integrators. Shaded portions represent one sigma standard deviation propagated from different realizations run under different random seeds.

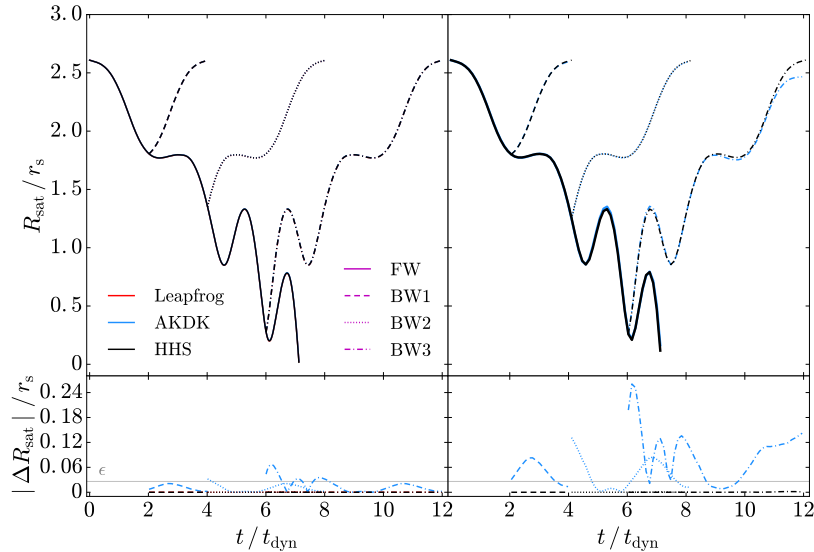


Fig. 11. Reversibility test. For the sinking satellite experiment we reversed the velocity sign at different moments, indicated by BW1 (dashed), BW2(dotted) and BW3 (dash-dot), and continue the integration. Solid lines represents the forward integration. Purple line labels only indicate the line-style corresponding to the start of each FW/BW integration. Upper panels show the normalized satellite distance as a function of time for the truly symplectic leapfrog (red lines), the adaptive one AKDK (blue lines) and HHS (black lines) integrators. Lower panels show the difference in position regarding the corresponding forward solution (solid lines). For high energy conservation accuracy ($\sim 10^{-8}$, left panels), there are some differences between integrators, however, they are inside (Leapfrog and HHS) or very close (AKDK) to the simulation softening (horizontal grey line). For a lower energy conservation accuracy ($\sim 10^{-3}$, right panels), still HHS and Leapfrog position differences are inside the simulation softening, whereas AKDK case is above the simulation resolution.

to the simulation softening. At a less accurate energy conservation but more common in collisionless simulations (10^{-3} , right panels), differences between forward and velocity reversal integration in Leapfrog (global constant step) and HHS are still below the simulation softening, however AKDK has notorious differences that are well above the simulation resolution and they are even detected in configuration space. We also tracked the fractional energy change (figure 12). The first outstanding fact is the truly reversible behaviour of the global-constant time-step leapfrog (upper panel), almost every peak and valley is reproduced after the velocity reversal. The middle panel shows the high accuracy tests for AKDK (blue lines) and HHS (black lines), and the bottom panel the corresponding low accuracy tests. For AKDK, there is a roughly systematic growth for both forward integration (blue solid opaque line) and also after reversing velocities, suggesting that AKDK is non-reversible. Instead, HHS is almost flat, suggesting that it is approximately reversible. We also run a test using HHS where we change the time variable sign. In this case, the energy shows a systematic growth, showing that HHS is not time symmetric, however, for the sake of clarity, it was not included. We performed similar tests for the fiducial isolated halo and obtained similar results. The small slope showed in the energy drift suggests that HHS is approximately reversible, furthermore energy drift is small compared to AKDK that is non-reversible and non time-symmetric. This in agreement with Hairer et al. (2002) who shows that the symplectic Störmer-Verlet-LeapFrog (AKDK) performs poorly for an adaptive time-step integrator, unless the time-step assigning function is properly selected with a possible computational cost.

Figure 5 shows that even GADGET4 with the selection function given by equation 5 and its implementation of HHS allows, under certain conditions, for a faster or more accurate integration than AKDK. Figure 13 investigate the reversibility of HHS and AKDK in the Tree (upper panel) and FMM (lower panel) implementations using GADGET4. The forward (FW) evolution of both integrators is shown with solid opaque lines, and the backward (BW) evolution after changing the velocity sign, at $40 t_{dyn}$, is shown with dashed lines. For the Tree gravity solver using HHS, the energy drift for both forward and backward evolution is similar particularly during the first $10 t_{dyn}$, however, although encouraging, we do not have enough evidence to claim that HHS is reversible, the time-step function adopted by GADGET4 may be the reason

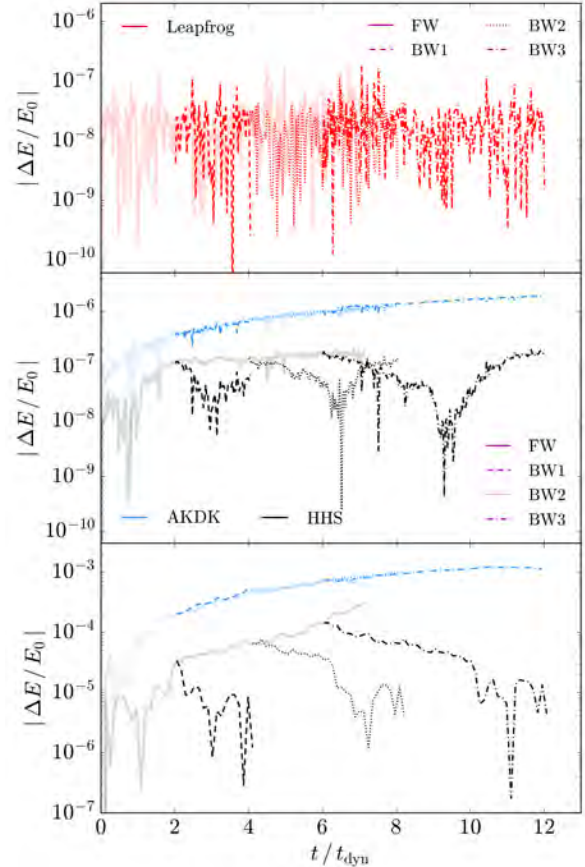


Fig. 12. Fractional energy error before (solid opaque lines) and after reversibility (dashed, dotted and dash-dot lines for BW1, BW2 and BW3 moments, respectively), for the same test of figure 11. Top panel shows the Global-constant time-step leapfrog, middle panel shows the higher accuracy in energy conservation tests, and the bottom panel the lower accuracy tests. Leapfrog is truly reversible. AKDK (blue lines) has a systematic growth of energy, suggesting that it is non-reversible. HHS (black lines) is almost flat with a small drift of energy after $\sim 2 t_{dyn}$ ($4.5 t_{dyn}$) for the high (low) accuracy test of the backward integrations, indicating that it is approximately reversible. Note that the purple line labels only indicate the line-style corresponding to the start of each FW/BW integration.

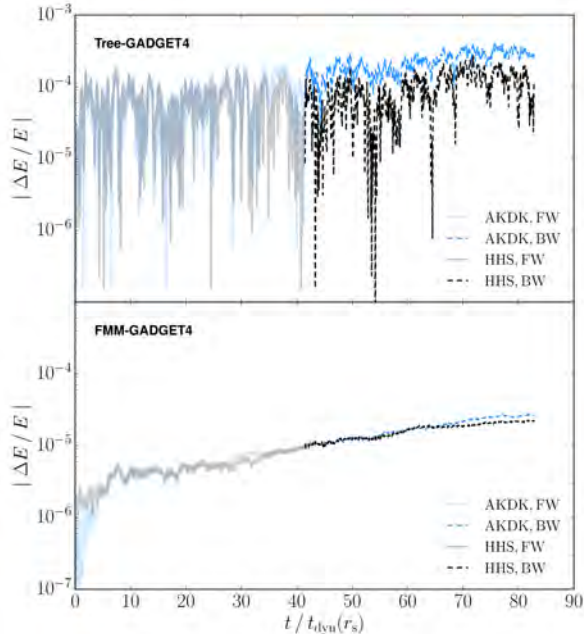


Fig. 13. Reversibility test for GADGET4. We followed the fiducial isolated halo using the GADGET4 code (shown with solid opaque lines) and we reversed the sign of particles velocities at around $40 t_{\text{dyn}}$ (dashed lines). Upper panel (Tree-GADGET4): for HHS with a non-symmetrized step function, remarkably up to $60 t_{\text{dyn}}$ the energy drift stays flat, then it slowly starts to grow. For AKDK the reversed integration just continues the forward systematic growth. Lower panel: For FMM with HHS and AKDK using a non-symmetrized step function, the energy drift grows systematically for both integrators. We can see that Tree-GADGET4 with HHS and the non-symmetrized step function could be approximately reversible or at least more stable, however, more studies is required.

is slightly different to our tests with a direct summation code. On the other hand, the reversed integration of AKDK shows a discontinuous change in its slope, indicating that it is not reversible, we conclude that Tree-HHS is more stable than Tree-AKDK. For the FMM gravity solver the test is not conclusive, because it does not show a significantly smaller slope in energy drift using the HHS scheme (Hierarchical Gravity).

9. DISCUSSION AND CONCLUSIONS

Using a GPU direct summation N -body code, we tested and characterized the Hierarchical Hamiltonian Splitting (HHS) integrator proposed by Pelupessy et al. (2012), but we focused on collisionless simulations. As a reference we compared with the global-constant time-step symplectic Leapfrog inte-

grator and the widely used Adaptive one (AKDK). We also complemented our study using the HHS implementation in GADGET4 (dubbed as hierarchical gravity, Springel et al. 2021), which uses different time-step selection function and approximate force solvers (Tree and FMM).

As recently discussed (e.g. Dehnen 2017), there is no general solution for a symplectic adaptive multi time-step integrator, although there are several proposals (e.g. Huang & Leimkuhler 1997; Hairer 1997; Calvo et al. 1998; Hardy et al. 1999; Farr & Bertschinger 2007). The problem is not exclusive of astrophysics, the field of differential equations of dynamical systems has extensively reviewed the subject (Hairer et al. 2002). In particular it is important to say that adaptive and multi-step techniques are not identical but they are related. Hernandez & Bertschinger (2018) address the important case of adaptive time steps, inspired by Hairer et al. (2002), they discuss what is called the “backwards error analysis” in order to explore commonly used integrators in astrophysics, concluding that symplecticity, time symmetry and reversibility are not the same and they are not always a guarantee of energy conservation. Nevertheless, KAM (Kolmogorov-Arnold-Moser) theorem assures stability for symplectic (geometrical) and reversible integrators (Hairer et al. 2002) in contrast to non-reversible ones, like AKDK with the standard time-step selection function. The case of multi-step integration, requires extra conditions like the so called impulse (splitting) or averaging (less often force evaluation) techniques (Hairer et al. 2002). Similar problems have been discussed in the molecular dynamics field, and the r-RESPA method (Tuckerman et al. 1992) that is an splitting technique has been applied to the long-short range splitting case or multiple mass segregation with considerable advantage in performance. An important feature is that such approaches to the multi-step problem in such cases resulted in a symmetric or reversible method not a symplectic one. The major concrete benefit has been improvement in the performance at a given energy accuracy. A similar situation applies to HHS, the case we study in this paper. HHS is a composition of *Fast* and *Slow* operators (Pelupessy et al. 2012) and based in our numerical experiments it is approximately reversible, which explains the performance advantage shown in section 6. The discussion in Hairer et al. (2002) and Hernandez & Bertschinger (2018) shows that the reversibility depends on the symmetry under a velocity sign change of hamiltonian and the corresponding time-step assigning function, and not on the accu-

racy of gravity calculation or particle number. The particular case of equation 10 is an approximated 1st order version of the one proposed by Hut et al. (1995) that depends only on the module of particle pairs relative velocities, therefore is reversible. A future interesting avenue inspired by the r-RESPA method presented by Tuckerman et al. (1992) is the extension already discussed by Portegies Zwart et al. (2020). We experimented with the strategy motivated by the multi-step averaging technique (Hairer et al. 2002), updating the time-step hierarchy only at a given number of global time-steps (dubbed as HHS-sTSS, figures 1 and 8), there is a gain in CPU time, however, the energy drift increases suggesting that further investigation is required.

Our results are summarized below:

1. Based on reversibility and time symmetry tests we concluded that HHS is not time symmetric but it is approximately reversible, it is also more stable than AKDK for a given energy accuracy. Although the exact correspondence between forward and backwards integration lasts only for few dynamical times, even for ten dynamical times the energy drift growth is small. In contrast, AKDK energy drift grows systematically with time, clearly showing that it is non-reversible using the commonly used time-step selection function. Based in our tests HHS reversibility explains the advantage in performance regarding AKDK. Based on Hairer et al. (2002) discussion and the used time-step selection functions that depend only on velocity module, such properties are independent of accuracy in gravity calculation and particle number.
2. Our findings with the direct summation code may be also relevant for the high accuracy and costly section (PP) of codes using the P3M technique, as it has been also show in molecular dynamics studies (Plimpton et al. 1997).
3. In agreement with our direct summation code tests, changing the time-step selection function for a non-symmetrized one, it is possible to find a combination of parameters in GADGET4 (using both the Tree and FMM code version) where HHS is more efficient than AKDK. We found approximated reversibility for the Tree Gadget4 test, however, for FMM Gadget4 our tests are inconclusive.
4. The population of particle histogram across the time-step rungs is useful to find a convenient

parameters combination for the integrators.

Acknowledgements

GAA thanks V. Springel for nicely clarifying GADGET4 information. GAA and JCC thanks Leonardo Ithehua Rico for the fruitful technical support and discussion. OV and GAA acknowledge support from UNAM PAPIIT grant IN112518. GAA and JAT thanks support from CONACyT graduate fellowships 455336 and 825451, respectively. OV acknowledge support from UNAM PAPIIT grants IG101620 and IG101222. HV acknowledges support from IN101918 PAPIIT-UNAM Grant and from a Megaproyecto DGTIC-LANCAD. The authors acknowledge DGTIC-UNAM for access to the Supercomputer MIZTLI. This research was partially supported through computational and human resources provided by the LAMOD UNAM project through the clusters Atocatl and Tochtli. LAMOD is a collaborative effort between the IA, ICN and IQ institutes at UNAM, and it acknowledges benefit from the MX28274 agreement with IBM.

REFERENCES

- Aarseth, S. J. 1971, *Ap&SS*, 14, 20
- Angulo, R. E. & Hahn, O. 2021, arXiv e-prints, arXiv:2112.05165
- Arca-Sedda, M. & Capuzzo-Dolcetta, R. 2016, *MNRAS*, 461, 4335
- Bagla, J. S. 2002, *Journal of Astrophysics and Astronomy*, 23, 185
- Barnes, J. & Hut, P. 1986, *Nature*, 324, 446
- Bode, P. & Ostriker, J. P. 2003, *ApJS*, 145, 1
- Bode, P., Ostriker, J. P., & Xu, G. 2000, *ApJS*, 128, 561
- Calvo, M., López-Marcos, M., & Sanz-Serna, J. 1998, *Applied Numerical Mathematics*, 28, 1
- Channell, P. J. 1993, in *Proceedings of Symplectic Integrators Workshop*, Waterloo Ontario, Canada
- Cheng, S., Yu, H.-R., Inman, D., Liao, Q., Wu, Q., & Lin, J. 2020, in *Accepted for SCALE 2020*, collocated as part of the proceedings of CCGRID 2020; 2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID); doi:10.1109/CCGrid49817.2020.00-22, arXiv:2003.03931
- Couchman, H. M. P. 1991, *ApJ*, 368, L23
- Dehnen, W. 2000, *ApJ*, 536, L39
- . 2002, *Journal of Computational Physics*, 179, 27
- . 2017, *MNRAS*, 472, 1226
- Dehnen, W. & Read, J. I. 2011, *European Physical Journal Plus*, 126, 55
- Efstathiou, G., Davis, M., White, S. D. M., & Frenk, C. S. 1985, *ApJS*, 57, 241
- Farr, W. M. & Bertschinger, E. 2007, *ApJ*, 663, 1420
- Grudić, M. Y. & Hopkins, P. F. 2020, *MNRAS*, 495, 4306

- Habib, S., Morozov, V., Frontiere, N., Finkel, H., Pope, A., & Heitmann, K. 2013, in SC '13 Proceedings of SC13: International Conference for High Performance Computing, 6
- Habib, S., Pope, A., Finkel, H., Frontiere, N., Heitmann, K., Daniel, D., Fasel, P., Morozov, V., Zagaris, G., Peterka, T., Vishwanath, V., Lukić, Z., Sehrish, S., & Liao, W.-k. 2016, *New A*, 42, 49
- Hairer, E. 1997, *Applied Numerical Mathematics*, 25, 219
- Hairer, E., Lubich, C., & Wanner, G. 2002, *Springer Series in Computational Mathematics*, 12, 399
- . 2003, *Acta Numerica*, 12, 399
- Hardy, D. J., Okunbor, D. I., & Skeel, R. D. 1999, *Applied Numerical Mathematics*, 29, 19
- Hayli, A. 1967, *The N-Body Gravitational Problem and the Simulation of Galactic Clusters*, Vol. 153, 315
- Hernandez, D. M. & Bertschinger, E. 2018, *MNRAS*, 475, 5570
- Hernandez, D. M., Hadden, S., & Makino, J. 2020, *MNRAS*, 493, 1913
- Hernquist, L. & Katz, N. 1989, *ApJS*, 70, 419
- Hockney, R. W. & Eastwood, J. W. 1998, *ISC Workshops*
- Huang, W. & Leimkuhler, B. 1997, *SIAM Journal on Scientific Computing*, 18, 239
- Hut, P., Makino, J., & McMillan, S. 1995, *ApJ*, 443, L93
- Jessop, C., Duncan, M., & Chau, W. Y. 1994, *Journal of Computational Physics*, 115, 339
- Klypin, A. 2018, *Cosmological N-Body Simulations (World Scientific Series in Astrophysics)*, 27–55
- Klypin, A., Prada, F., Yepes, G., Heß, S., & Gottlöber, S. 2015, *MNRAS*, 447, 3693
- Kravtsov, A. V., Klypin, A. A., & Khokhlov, A. M. 1997, *ApJS*, 111, 73
- Makino, J. 1991, *ApJ*, 369, 200
- McMillan, S. L. W. in , *The Use of Supercomputers in Stellar Dynamics*, ed. P. Hut S. L. W. McMillan, Vol. 267, 156
- Miller, T. B., van den Bosch, F. C., Green, S. B., & Ogiya, G. 2020, *MNRAS*, 495, 4496
- Navarro, J. F., Frenk, C. S., & White, S. D. M. 1997, *ApJ*, 490, 493
- Oshino, S., Funato, Y., & Makino, J. 2011, *PASJ*, 63, 881
- Pelupessy, F. I., Jänes, J., & Portegies Zwart, S. 2012, *New A*, 17, 711
- Plimpton, S. J., Pollock, R., & Stevens, M. J. 1997, *PPSC*, 1997
- Portegies Zwart, S., Pelupessy, I., Martínez-Barbosa, C., van Elteren, A., & McMillan, S. 2020, *Communications in Nonlinear Science and Numerical Simulations*, 85, 105240
- Quinn, T., Katz, N., Stadel, J., & Lake, G. 1997, *arXiv e-prints*, astro-ph/9710043, astro
- Sellwood, J. A. 1985, *MNRAS*, 217, 127
- Springel, V. 2005, *MNRAS*, 364, 1105
- Springel, V., Pakmor, R., Zier, O., & Reinecke, M. 2021, *MNRAS*, 506, 2871
- Springel, V., Yoshida, N., & White, S. D. M. 2001, *New A*, 6, 79
- Stadel, J. G. 2001, PhD thesis, UNIVERSITY OF WASHINGTON
- Swope, W. C., Andersen, H. C., Berens, P. H., & Wilson, K. R. 1982, *J. Chem. Phys.*, 76, 637
- Teyssier, R. 2002, *A&A*, 385, 337
- Tuckerman, M., Berne, B. J., & Martyna, G. J. 1992, *J. Chem. Phys.*, 248, 1990
- Tuckerman, M. E., Martyna, G. J., & Berne, B. J. 1990, *J. Chem. Phys.*, 93, 1287
- Verlet, L. 1967, *Physical Review*, 159, 98
- Villumsen, J. V. 1989, *ApJS*, 71, 407
- Wang, J., Bose, S., Frenk, C. S., Gao, L., Jenkins, A., Springel, V., & White, S. D. M. 2020, *Nature*, 585, 39
- Weinberger, R., Springel, V., & Pakmor, R. 2020, *ApJS*, 248, 32
- Xu, G. 1995, *ApJS*, 98, 355
- Zemp, M., Stadel, J., Moore, B., & Carollo, C. M. 2007, *MNRAS*, 376, 273
- Zhu, Q. 2017, *arXiv e-prints*, arXiv:1712.10116

Capítulo 5

Métodos para simular el efecto de neutrinos masivos en el crecimiento de la estructura cósmica

Dada la importancia que tienen las simulaciones en la cosmología, el presente capítulo presenta una breve revisión de algunas de las técnicas/métodos de la literatura actual para realizar simulaciones cosmológicas con una componente explícita de neutrinos masivos, discutiendo sus ventajas y desventajas dentro de la cosmología contemporánea. Motivando de esta forma, el uso de la técnica implementada en el código BPM, otro de los códigos desarrollado en esta tesis.

5.1. Relevancia de las simulaciones con neutrinos masivos

Como se discutió anteriormente, incluir explícitamente el efecto de neutrinos masivos en las simulaciones de formación de estructura cósmica es necesario para la interpretación precisa de las observaciones de BAO, del espectro de fluctuaciones de densidad y de abundancia de cúmulos de galaxias y su consistencia con la cosmología. Además, permitirá acotar la suma de masa de los neutrinos o en su defecto revelar la presencia de otro grado de libertad relativista.

Los efectos de los neutrinos masivos sobre la estructura a gran escala están bien entendidos sólo a escalas grandes y a tiempos tempranos, cuando las fluctuaciones se encuentran en el régimen lineal [54]. Sin embargo, a tiempos tardíos y pequeñas escalas las fluctuaciones de densidad se vuelven altamente no-lineales, haciendo que la teoría de perturbaciones lineal ya no sea válida.

A pesar de que las simulaciones de N -cuerpos han sido exitosas al describir la evolución de cosmologías CDM, no han sido tan buenas para el estudio de ciertas cosmologías, especialmente para universos que contienen partículas masivas con grandes velocidades térmicas en la distribución inicial del espacio fase de las especies. Dichas velocidades generan estructuras espurias en el campo de densidad debido a los efectos de discretización inherente en los métodos de N -cuerpos. Para valores típicos de densidad

numérica promedio de las partículas en la simulación, el espectro de potencias debido al “shot-noise” producido en la simulación puede dominar completamente el espectro de potencias físico, dando como resultado estructuras espurias formadas en todos lados en el volumen de la simulación. Por lo tanto, las mejoras en los métodos numéricos son esenciales para desarrollar un método fiable y consistente para estudiar la formación de estructura de forma precisa en tales cosmologías.

Dada las implicaciones fundamentales que tendría el entender o restringir algunas de las propiedades incógnita de los neutrinos, desde un punto de vista cosmológico, así como la precisión que están alcanzando los nuevos (y futuros) sondeos, ha habido un esfuerzo para mejorar las simulaciones cosmológicas de modelos con una componente de neutrinos masivos. En este capítulo discutiré algunos de los métodos/algoritmos que actualmente existen en la literatura, los cuales pueden ser clasificados como métodos: basados en partículas, basados en una malla, perturbativos+ N -cuerpos, e híbridos.

5.2. Descripción más general

La forma más general para seguir la evolución no-lineal del crecimiento de la estructura de partículas no-colisionales (incluidos los neutrinos) es mediante la ecuación de Boltzmann no-colisional:

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial x^i} \frac{dx^i}{dt} + \frac{\partial f}{\partial p^i} \frac{dp^i}{dt} = 0 \quad (5.1)$$

en donde $f(\mathbf{x}, \mathbf{p})$ es la función de distribución del espacio fase de la especie dinámica a simular.

Sin embargo, esta ecuación es de 6+1 dimensiones (6 dimensiones del espacio fase más una temporal), por lo que seguir su evolución es computacionalmente muy costoso, tanto en tiempo de cómputo como en memoria, haciendo imposible usar más allá de $N_g = 128$ celdas por dimensión [115]. Para hacerse una idea, si quisiéramos resolver la ecuación usando una malla con $N_g = 64$ celdas por cada dimensión del espacio fase (en total serían 64^6 celdas en toda la malla), necesitaríamos alrededor de ~ 270 GB de memoria. Para una simulación con $N_g = 128$ celdas por cada dimensión (128^6 en total) se requerirían ~ 17 TB de memoria, para una con $N_g = 256$ celdas se necesitaría ~ 1 PB, y para una con $N_g = 512$ **requeriríamos $\sim 70PB!$** , lo que equivale a **50 veces Sequoia**, una de las supercomputadoras del top 500 del mundo.

Mientras que para la precisión requerida por la nueva generación de sondeos y datos, se requeriría resolver la ecuación usando una malla de por lo menos $N_g = 1024$ celdas, haciendo imposible realizar estas simulaciones con la tecnología actual. Actualmente se busca tener descripciones híbridas del método completo 6D, acoplada a simulaciones estándar, así como simulaciones adaptivas del solucionador de Boltzmann en 6D ([116]), el código desarrollado en esta tesis puede abordar el problema pero no seguimos esa ruta dada la inviabilidad computacional con los recursos disponibles para nosotros, pero es algo que se puede retomar.

5.3. Métodos basados en partículas

La estrategia más natural para incorporar a neutrinos masivos a los cálculos de la formación de estructura son los llamados métodos basados en partículas, también conocidos como “Cold+Hot”. En este caso los neutrinos son incluidos como un conjunto extra de partículas con una componente de velocidad térmica extra a las de CDM y son evolucionados mediante códigos de N -cuerpos en conjunto con las partículas CDM.

Como se mencionó anteriormente, este tipo de métodos potencialmente sufren de shot-noise, debido a dos efectos: la masa tan pequeña hace imposible que las simulaciones se acerquen siquiera a un número realista de neutrinos, lo cual implica una amplitud de fluctuaciones poissónicas en el campo de densidad de neutrinos que rivaliza con las fluctuaciones físicas incluyendo potencia espuria. El segundo efecto son las grandes velocidades térmicas de los neutrinos no despreciables respecto a la de la luz, a altos corrimientos al rojo, tales velocidades permiten que éstos crucen la caja a simular múltiples veces, ocasionando que las partículas de éstos pierdan rápidamente la memoria sobre sus condiciones iniciales y se distribuyen aleatoriamente en la caja, dando lugar al shot-noise en el espectro de potencias a grandes escalas, afectando también la longitud de flujo libre, existe aún debate sobre la importancia de tales efectos relativistas ([117]). El shot-noise discutido anteriormente, puede llegar a dominar completamente el espectro de potencias si no se usa un alto número de partículas ($\gtrsim 10^{12}$) y volúmenes grandes ($\gtrsim 1$ Gpc), lo cual no es trivial de alcanzar con la tecnología computacional actual, pues una simulación con dichas características resulta muy costosa computacionalmente (tanto en tiempo como en memoria).

Actualmente existen muy pocas simulaciones con el método Cold+Hot, que han alcanzado o están en el límite de dichos requerimientos, un ejemplo de ellas es la simulación Outer Rim [63], la cual simuló un volumen de $\sim (3 h^{-1} \text{Gpc})^3$ con $\sim 1.07 \times 10^{12}$ partículas usando el código HACC [41]. Dicha simulación fue realizada en la supercomputadora Mira y requirió de ~ 1 PTB de RAM, utilizando 32 racks de la supercomputadora, con 1024 nodos cada uno y con 16 núcleos cada nodo. En total generó ~ 5 PTB de datos (snapshots a diferentes z), y por su alto costo computacional sólo fue posible hacer una sola realización.

Otro ejemplo, es la simulación TianNu [118], la cual simuló un volumen cúbico de lado $1.2 h^{-1} \text{Gpc}$ con 6912^3 partículas CDM y 13824^3 partículas neutrino usando el código CUBEP³M [119] optimizado. Esta simulación fue realizada en la supercomputadora Tianhe-2 y requirió 13,824 nodos (~ 86 % de dicha supercomputadora), generando sólo 25 TB de datos ya que únicamente guardaron el snapshot final a $z = 0$.

Otro ejemplo, es el conjunto de 43100 simulaciones Quijote [120] que abarcan más de 7000 modelos cosmológicos en el hiper-plano $\{\Omega_m, \Omega_b, h, n_s, \sigma_8, M_\nu, w\}$ realizadas con el código GADGET-3 (versión mejorada de GADGET-2 [36]). Cada simulación tiene un volumen cosmológico de $1(h^{-1} \text{Gpc})^3$, y para las simulaciones con neutrinos usaron 512^3 partículas CDM y 512^3 partículas neutrino para los valores de masa $\Sigma m_\nu = 0.4, 0.2, 0.1$ eV con 500 realizaciones para cada masa. El conjunto total de simulaciones requirió de un poco más de 35 millones de horas núcleo generando más de 1 PTB de datos (disponibles públicamente) almacenados en servidores en San Diego, New York y Princeton. Cabe mencionar que si bien es un gran esfuerzo referente a la exploración

del espacio de parámetros disponible, no discute en absoluto el efecto del shot-noise. Por lo que seguramente se requerirán nuevas generaciones de simulaciones cuando se vaya a predicciones de alta resolución.

Recientemente, [121] propuso un nuevo método para generar condiciones iniciales para los métodos basados en partículas y para simulaciones con neutrinos masivos, con el propósito de disminuir el shot-noise presente en éstos. El método es una generalización de la estrategia seguida por [122], el cual a su vez tiene elementos en común con la técnica llamada “quite start” [123] para condiciones iniciales en simulaciones de discos cinemáticamente calientes. Ambas estrategias están motivadas por técnicas similares en simulaciones unidimensionales de física de plasmas no-colisionales [124]. [122] incluye a las partículas tipo neutrino considerando pares con velocidades opuestas, esto restringe el shot-noise asociado a una dependencia con número de onda k , proporcional a k^4 , debido a la cancelación aproximada del momento. Lo cual a grandes escalas minimiza el shot-noise, aunque a pequeñas escalas solo el número de partículas amortiguará las fluctuaciones. En lugar de incluir pares opuestos de partículas en el espacio fase, en [121] muestrean todo un volumen esférico en el espacio fase, incluyendo una serie de pares de partículas con orientación regular y velocidades opuestas en el espacio de momentos. Primero cuasi-aleatoriamente la distribución de Fermi-Dirac al generar las velocidades térmicas iniciales. Para ello, primero muestrean la distribución de magnitud de velocidades $p^2 f(p)$ en cascarones, y después subdividen cada uno de estos cascarones en elementos de igual área para asignar la dirección de la velocidad de forma isotrópica de cada cascarón. Finalmente, este procedimiento es replicado en cada punto en el espacio de configuraciones.

La razón por la cual este método ayuda a reducir el shot-noise es que, al fijar las magnitudes y direcciones de las velocidades térmicas de los neutrinos, el número de neutrinos que entran y salen de un volumen dado de la caja a simular no es un número aleatorio, como ocurre cuando la función de distribución es muestreada de forma totalmente aleatoria (es lo que típicamente se hace). En cambio, presumiblemente el flujo de partículas tipo neutrino es cercano a lo que implicaría el espectro de potencias físico de los neutrinos, que establece los desplazamientos iniciales fuera de los puntos de la malla y la evolución gravitacional de las trayectorias, [121] verifican su validez en base al espectro de potencias de materia (MPS) para algunos experimentos numéricos, sin embargo, mayor entendimiento es necesario. Este método de generar condiciones iniciales puede inducir correlaciones espurias entre neutrinos y CDM a escalas pequeñas [125] si se inicializan las partículas de neutrinos a corrimientos al rojo tardíos o si se usa un muestreo diferente para la distribución de Fermi-Dirac, en todos los casos generando potencia espuria en el MPS [91].

Recientemente [91] utiliza un código de malla PM llamado FastPM con un esquema de elección de paso de integración para el AKDK y el esquema de condiciones iniciales de [121], la combinación entre resolución y técnica PM borran parte de la estructura espuria, sin embargo, un estudio más profundo es necesario.

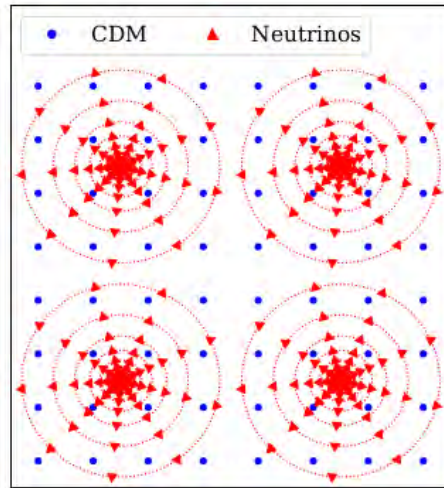


Figura 5.1: Ilustración 2D de la configuración inicial del método propuesto por [121]. Las partículas CDM están representadas con puntos azules y las partículas de neutrinos con triángulos rojos, en donde la dirección de cada triángulo indica la dirección de la velocidad [91].

5.4. Métodos basados en malla

Los métodos basados en una malla incluyen a la componente de neutrinos como un **fluido lineal** (p. ej. [126, 127]) o resuelven las **ecuaciones de Boltzmann linealizadas** (p. ej. [128]) únicamente sobre una malla, la cual se encuentra acoplada a la evolución, no-lineal, de N -cuerpos de las partículas CDM.

Este tipo de métodos funciona bien cuando los efectos no-lineales de los neutrinos son despreciables ($z \gtrsim 2$ y a escalas lineales), por lo que dejan de ser válidos cuando las sobre-densidades de la componente de neutrinos se vuelven no-lineales.

MassiveNuS [129] es un conjunto de 101 simulaciones realizadas usando este tipo de método. En particular, siguen la evolución de la componente de CDM usando una versión modificada de GADGET-2 [36], y la componente de neutrinos es evaluada a partir del fondo de CDM mediante la solución de la ecuación de Boltzmann linealizada respecto a las sobre-densidades de neutrinos. Cada una de estas simulaciones fue realizada cambiando la suma de masa de neutrinos Σm_ν (en el rango de 0 a 0.6 eV), densidad de materia Ω_m y la amplitud del espectro de potencias primordial A_s . Para estas simulaciones se usaron 1024^3 partículas CDM en un volumen cúbico de lado $512 h^{-1} \text{Mpc}$.

5.5. Métodos perturbativos+ N -cuerpos

La llamada técnica “Comoving Lagrangian Acceleration” (COLA) [130, 131] permite incluir correcciones perturbativas a las ecuaciones de movimiento de las partículas. Inicialmente la técnica se propuso para evitar los grandes números de partículas que se requieren para poblar los modos o estructuras de tamaño muy grande y baja amplitud,

resolviendo estructuras pequeñas como halos y realizar simulaciones rápidas, en este caso el desplazamiento a escalas grandes en lugar de depender de una solución ruidosa del campo de densidad, es una aproximación perturbativa típicamente de segundo orden. La solución perturbativa incluyendo materia oscura y otra física como un campo escalar puede abordarse, en nuestro caso partículas de tal campo representaría los neutrinos masivos. MG-PICOLA [89, 90] es un código de malla (PM) que a través de la técnica COLA incluye el efecto de los neutrinos, tales soluciones son muy eficientes computacionalmente pero están limitadas a la precisión de la solución perturbativa la cual a su vez debe calibrarse utilizando simulaciones completas. Son útiles como una primera exploración o para la generación de simulaciones rápidas y con precisión controlada. Cabe mencionar que en este enfoque sólo se sigue a las partículas de CDM y al efecto de los neutrinos sobre éstas, similar a [126]. Aún se requiere un estudio cuantitativo comparando este tipo de técnicas con otras. Recientemente [132] introdujeron una técnica similar motivada por herramientas perturbativas en física de plasmas, acopladas a simulaciones tradicionales de N -cuerpos para la componente CDM (GADGET-3). En ese enfoque ambas especies pueden seguirse por separado. La implementación que ilustra el artículo de presentación es lineal, pero es posible extenderla. Es una técnica prometedora que debe cuantificarse.

5.6. Método de momentos de la ecuación de Boltzmann

Con el objetivo de tener simulaciones más precisas que puedan capturar tanto el régimen lineal como el cuasi-lineal (o no-lineal) se han comenzado a desarrollar nuevos métodos para la inclusión de una componente de neutrinos masivos en las simulaciones de crecimiento de estructura. Uno de estos métodos, propone incluir a la componente de neutrinos como un medio continuo, el cual está descrito a través de momentos de la ecuación de Boltzmann (completa, ver ec. 5.1).

Ahora bien, para poder seguir la evolución de un fluido no-colisional mediante el método antes descrito, se requiere de cerrar la jerarquía de Boltzmann. Para fluidos colisionales, puede usarse una ecuación de estado que relacione la densidad de energía con la presión para cerrar el sistema de ecuaciones (y con ello la jerarquía de Boltzmann). Sin embargo, para algunos de los casos no-colisionales, como lo es para los neutrinos, no existe una ecuación de estado simple, por lo que se requiere de una alternativa para cerrar dicha jerarquía.

En [133] proponen cerrar la jerarquía de Boltzmann mediante momentos de la función de distribución estimados a partir de partículas prueba, por lo que en su propuesta, representan a los neutrinos simultáneamente como medio continuo sobre una malla y así como usando partículas pruebas.

Por otro lado, en ν CONCEPT [134] proponen cerrar la jerarquía de Boltzmann usando la forma linealizada de la presión y del esfuerzo anisotrópico, los cuales son calculados usando CLASS [135, 136]. Sin embargo, no está claro si al hacer dicha elección existirá algún efecto en las escalas cuasi-lineales o en las no-lineales. Es importante mencionar que este código no garantiza que la solución que calcula sea TVD, por lo

que puede generar inestabilidades.

5.7. Limitantes y su impacto en las restricciones cosmológicas

La figura 5.2 muestra los resultados obtenidos de una simulación realizada con GADGET-3 para las componentes CDM, bariones y neutrinos masivos [137]. En la figura se muestra el campo de densidad en el plano XY en $z = 0$, para las tres componentes y para diferentes masas de neutrinos: $\Sigma m_\nu = 0, 0.1, 0.4, 0.8$ eV.

De esta figura se puede observar que para masas de neutrinos pequeñas el efecto de éstos es consistente con el crecimiento lineal de las fluctuaciones afectadas por el flujo libre, y conforme la masa de éstos aumenta, los efectos del agrupamiento de materia son más notables, haciendo incluso que algunas de las sobre-densidades de materia oscura se diluyan.

Entonces, un método para incluir una componente explícita de neutrinos que no pueda capturar el régimen no-lineal en simulaciones cosmológicas nos sesgará artificialmente a masas pequeñas de neutrinos o a Λ CDM. Por lo tanto, para generar simulaciones con la precisión requerida por los actuales y futuros sondeos de galaxias es indispensable que el método para incluir neutrinos en las simulaciones pueda capturar el régimen causi-lineal y no lineal de forma precisa.

En la tabla 5.1 se muestra de forma resumida los pros y los contras que tienen cada uno de los métodos discutidos en las secciones previas.

Podemos ver que la promesa de los censos de galaxias y la gran precisión que alcanzarán y que podrían lograr detectar el efecto de los neutrinos en el agrupamiento de galaxias, ha estimulado una gran actividad por representar tanto fielmente como eficientemente el efecto de los mismos sobre el crecimiento de estructura cósmica en simulaciones cosmológicas. No existe una solución final. Si bien los métodos basados en la ecuación de Boltzmann tanto en 6D como a partir de momentos tienen una gran potencial teórico, su costo computacional no es nada despreciable aunque podrá ser manejado por la nueva generación de computadoras peta y exascale. Los otros métodos linealizados o perturbativos son muy eficientes pero suelen requerir calibración con la ayuda de simulaciones completas, su capacidad predictiva se limita si se busca dilucidar la no-linealidad o física extra como auto-interacción o decaimiento. En esta tesis elegimos el método de momentos de la ecuación de Boltzmann tanto por su eficiencia respecto al caso 6D como por su susceptibilidad de ser extendido al caso más general.

¹Supercomputadora japonesa que ha liderado, por ahora, tres veces consecutivas (Junio 2020, Noviembre 2020 y Junio 2021) la lista TOP500 de las mejores 500 supercomputadoras de alto rendimiento del mundo. Para más información sobre Fugaku visitar: <https://www.r-ccs.riken.jp/en/fugaku/>

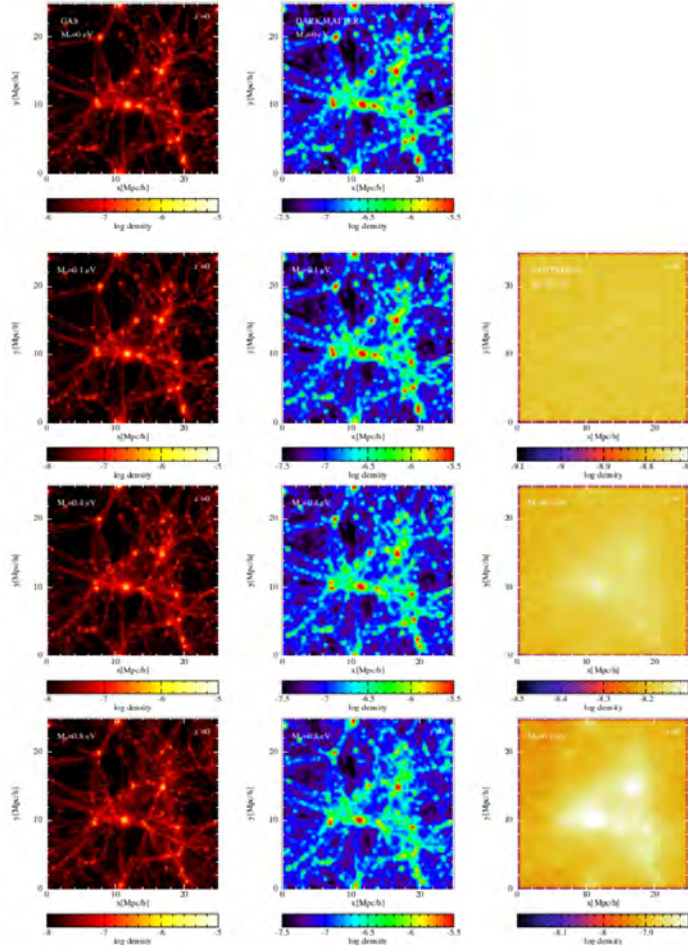


Figura 5.2: Proyecciones del campo de densidad sobre el plano XY a $z = 0$ para una caja de $25 h^{-1}\text{Mpc}$ con $N_p = 192^3$ partículas de cada tipo (DM, bariones y neutrinos) obtenido con GADGET-3 [137] considerando una componente de neutrinos. De arriba hacia abajo, la suma de masas de los neutrinos es $\Sigma m_\nu = 0, 0.1, 0.4$ y 0.8 eV, y de izquierda a derecha se muestra las componentes: Gas (bariones), materia oscura (DM) y neutrinos. Entre mayor es la masa de los neutrinos los efectos del agrupamiento son más notables, por lo tanto un método que no pueda capturar el régimen no-lineal sesgará hacia menores Σm_ν y/o ΛCDM .

Método	Pros	Contras
Boltzmann Completo	<ul style="list-style-type: none"> - Es la descripción más completa, incluyendo efectos no-lineales 	<ul style="list-style-type: none"> - Altamente costoso, memoria, apenas abordable en Fugaku¹ - Resolución requerida no es alcanzable aún, solo en esquema híbrido
Basado en Partículas	<ul style="list-style-type: none"> - Incluye efectos no-lineales 	<ul style="list-style-type: none"> - Alto costo computacional - Shot-noise en escalas pequeñas
Basado en una Malla	<ul style="list-style-type: none"> - Menos costosas que N-cuerpos - Componente ν no tiene shot-noise 	<ul style="list-style-type: none"> - Deja de ser válido en régimen no-lineal, calibración - Componente CDM puede tener shot-noise
Perturbativos + N -cuerpos	<ul style="list-style-type: none"> - Bajo costo computacional - Componente ν no tiene shot-noise 	<ul style="list-style-type: none"> - Precisión limitada por parte perturbativa, requiere calibración - Puede haber shot-noise en componente CDM
Momentos de Boltzmann	<ul style="list-style-type: none"> - No tienen shot-noise - Válidos en escalas cuasi-lineales* - Mallas menos finas en simulaciones, caso intermedio entre N-body y Boltzmann completo 	<ul style="list-style-type: none"> - Mayor consumo de CPU que PM - Truncan jerarquía de Boltzmann

Cuadro 5.1: Pros y contras de los métodos que actualmente hay para incluir una componente de neutrinos masivos en las simulaciones de crecimiento de estructura.

Capítulo 6

Código Boltzmann-PM o BPM

Este capítulo presenta al código Boltzmann-PM o BPM desarrollado como parte de esta tesis y que es uno de los principales productos de ésta. Aquí se describe el algoritmo usado y su implementación en el código, que como veremos en las siguientes secciones, está constituido por dos códigos con técnicas distintas. Posteriormente, se presentan una serie de pruebas numéricas realizadas a los códigos desarrollados (y explicados) en este capítulo.

Como ya hemos discutido, para poder estudiar cosmologías con neutrinos masivos con la precisión necesaria, se requiere que las simulaciones resuelvan de forma precisa la estructura a gran escala, es decir, que puedan resolver varias escalas de BAO, pero también deben resolver las escalas pequeñas, es decir, deben poder resolver las escalas de los trazadores observacionales, de lo contrario puede haber un sesgo en la interpretación de los resultados.

Para lograr lo anterior, es necesario que la aproximación a usarse en las simulaciones con neutrinos naturalmente pueda resolver tanto escalas lineales como escalas no-lineales, y que no esté dominada por el shot-noise mencionado anteriormente, es por ello que en este proyecto doctoral utilizaremos el método de momentos de la ecuación de Boltzmann, y en particular usaremos la propuesta de [133] para cerrar la jerarquía de Boltzmann. Es importante mencionar que nuestra implementación permite seguir múltiples especies simultáneamente a diferencia de [133]. Por ejemplo, tanto la materia oscura como los neutrinos podría ser representada por un medio continuo, esto puede tener ventajas para amortiguar el ruido a gran escala, aunque requiere un estudio cuantitativo.

6.1. Algoritmo

En la sección 5.6 se discutió la idea general de la propuesta de momentos de la ecuación de Boltzmann no-colisional (ver ec. 5.1). Aquí, discutiré de forma más detallada dicho método.

Para construir este algoritmo, partimos de la ecuación de Boltzmann no-colisional

(ec. 5.1), en donde, asumiendo una métrica Newtoniana [138], se tiene [139]:

$$\frac{dx^i}{dt} = \frac{p^i}{aE} (1 - \Phi + \Psi) \quad (6.1)$$

y,

$$\frac{dp^i}{dt} = -p^i \left(\frac{\partial \Phi}{\partial t} + \frac{\dot{a}}{a} \right) - \frac{E}{a} \frac{\partial \Psi}{\partial x^i} \quad (6.2)$$

con Φ y Ψ los potenciales Newtonianos, a el factor de escala y $E = \sqrt{p^i p^i + m^2}$. \dot{a} se refiere a una derivada respecto al tiempo, y no respecto a un tiempo conforme.

Los potenciales métricos Φ y Ψ están conectados con el contenido de la materia a través de la ecuación de Poisson:

$$\nabla^2 \Psi = 4\pi G a^2 \left[\bar{\rho} \delta - 3 \frac{\dot{a}}{a} (1 + w) \partial_i v^i \right] \quad (6.3)$$

donde v^i es la velocidad peculiar local, $\delta = \delta\rho/\bar{\rho}$ es la sobre-densidad local, y $w = \bar{P}/\bar{\rho}$ es la ecuación de estado del fondo. En esta ecuación se ha asumido que $\Psi = -\Phi$, lo cual es válido para épocas de dominio de materia ($z \lesssim 300$), cuando el esfuerzo anisotrópico general es pequeño. En simulaciones donde el volumen es mucho más pequeño que el volumen de Hubble, el segundo término puede ser despreciado.

Ahora bien, para obtener las ecuaciones de fluido con las que se describirán a las partículas no-colisionales (e.g. neutrinos), obtenemos momentos de la ecuación de Boltzmann (ec. 5.1) sustituyendo, previamente, las ecuaciones 6.1 y 6.2 en ésta. Luego, el momento cero (ec. tipo la de continuidad) se obtiene multiplicando por E a la ecuación resultante, e integrando sobre \mathbf{p} (para más detalle ver apéndice A):

$$\dot{\delta} + \frac{1}{a} \frac{\partial}{\partial x^i} [(1 + 2\Psi) \Pi^i] = 0 \quad (6.4)$$

en donde,

$$\Pi^i = \frac{\int p^i f(\mathbf{x}, \mathbf{p}) d^3 p}{\bar{\rho}} \quad (6.5)$$

De forma similar, para obtener el primer momento (ec. tipo Euler), multiplicamos por p^i a la ec. 5.1 e integramos (para ver detalle A):

$$\dot{\Pi}^i = - (1 - 3\omega) \frac{\dot{a}}{a} \Pi^i - \frac{1}{a} \frac{\partial}{\partial x^j} [(1 + \delta) W^{ij}] - \frac{1 + \delta}{a} \frac{\partial \Psi}{\partial x^i} \quad (6.6)$$

con

$$W^{ij} = \frac{\int \frac{p^i p^j}{E} f d^3 p}{\rho} \quad (6.7)$$

En principio, se puede continuar este proceso de generación de ecuaciones utilizando momentos cada vez más altos de la ecuación de Boltzmann, en donde cada ecuación obtenida dependería de la siguiente, por ejemplo, la ecuación 6.4 depende de Π^i , y la ecuación 6.6 depende de W^{ij} . Por lo anterior, es requerido cerrar esta jerarquía infinita de ecuaciones, conocida como jerarquía de Boltzmann.

Como mencioné al inicio de este capítulo, el método que utilizamos para este proyecto es el propuesto por [133]: usar partículas prueba para estimar momentos de la función de distribución (DF, por sus siglas en inglés) y con ellos truncan la jerarquía.

Dicho lo anterior, el algoritmo de este método es el siguiente:

- La componente de neutrinos masivos es representada como un fluido no-colisional que está descrito por la ecuación de Boltzmann (ec. 5.1).
- El fluido no-colisional es representado simultáneamente usando cantidades de fluido en una malla (mediante las ecs. 6.4 y 6.6), y usando partículas prueba.
- La malla es evolucionada mediante los momentos de Boltzmann (ecs. 6.4 y 6.6).
- Truncar la jerarquía usando momentos de la DF (ecs. 6.5 y 6.7) estimados a partir de las partículas prueba.
- Las partículas prueba se evolucionan bajo el campo gravitacional estimado del fluido, usando la ec. 6.3.

6.2. Implementación

Como vimos, en el método de momentos de Boltzmann la componente de neutrinos es representada tanto como medio continuo en una malla y como partículas prueba, y cada una de éstas es evolucionada con distintas ecuaciones cuya solución requiere de métodos numéricos distintos. Entonces, la implementación del método se puede separar en dos partes. La primera consiste en resolver la representación de medio continuo, usando técnicas hidrodinámicas de advección, para lo cual desarrollé un código al cual me referiré como “Hidro-Boltzmann”. La segunda parte consiste en resolver la representación de partículas prueba, mediante N -cuerpos usando la técnica PM [140, 35], con este propósito desarrollé un código al cual me referiré como “PM”.

Finalmente, para la implementación del método completo de Boltzmann desarrollé un tercer código, el cual acopla a los códigos antes mencionados (Hidro-Boltzmann y PM). A este código me referiré como “Boltzmann-PM”.

6.2.1. Código PM

Este código resuelve el problema de N -cuerpos usando la técnica “Partícula-Malla” (PM) [140, 35, 141, 142]. En esta técnica se construye un campo de densidad sobre una malla a partir de la distribución de partículas, y se sigue la evolución de esta mediante la ecuación de Poisson asociada al problema (para mayor detalle ver el apéndice B).

En particular, el código que desarrollé tiene las siguientes características numéricas:

- ▷ Usa la interpolación *Cloud-In-Cell* (CIC, [143]) para la construcción del campo de densidad sobre la malla.
- ▷ Usa la biblioteca FFTW3 [144] para calcular las FFT’s necesarias en el código.

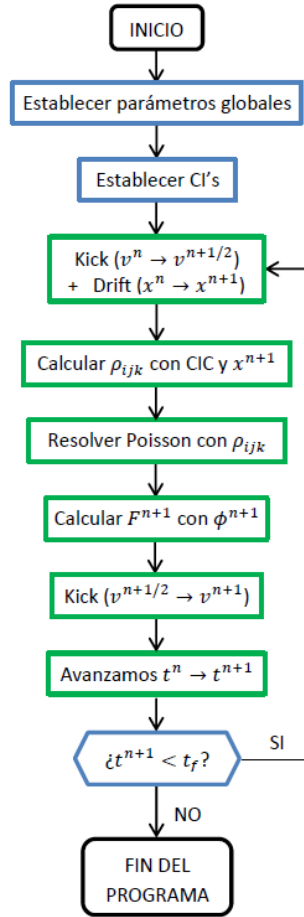


Figura 6.1: Diagrama de flujo código PM.

- ▷ Usa funciones de Green de 1er. o 2o. orden (p. ej. [145]) para calcular el potencial en el espacio de Fourier.
- ▷ Usa condiciones de frontera periódicas.
- ▷ Usa diferencias finitas de 2o. o 4o. orden para el cálculo de las fuerzas.
- ▷ Usa el integrador *Kick-Drift-Kick* (KDK) para evolucionar a las partículas, mediante las ecs. 6.1 y 6.2.
- ▷ Las unidades internas del código son adimensionales.

En la figura 6.1 se muestra el diagrama de flujo del código.

6.2.2. Código Hidro-Boltzmann

Este código evoluciona a un medio continuo, descrito a través de las ecuaciones 6.4 y 6.6, sobre una malla 3D y usando técnicas hidrodinámicas de advección. Para ello, utiliza el método llamado “División del operador” [146], el cual permite dividir

términos no-lineales (o complejos de resolver en conjunto) de un sistema de ecuaciones parciales y simplificar los métodos de solución.

En particular, en el código usamos dicha técnica para separar cualquier término fuente de la parte advectiva de las ecuaciones 6.4 y 6.6, de esta forma hacemos por separado la solución de los términos fuente y la de un sistema de dos ecuaciones advectivo en 3D. Posteriormente, para el sistema advectivo (en 3D) obtenido previamente, aplicamos nuevamente la misma técnica pero ahora para dividir dicho sistema en seis ecuaciones 1D (a esto se le conoce como “División direccional”, [146]).

Entonces, después de este procedimiento convertimos el problema original (ecs. 6.4 y 6.6) en dos problemas más sencillos de resolver: un problema que sólo tiene a los términos fuente y otro que tiene un sistema de seis ecuaciones de advección en 1D.

Ahora bien, para resolver el problema de advección 1D sobre la malla se usa un esquema de volumen finito que es “piecewise linear”, y por lo tanto de segundo orden en el espacio, en conjunto del limitador de flujo Superbee [147], el cual es requerido para que el método sea “Total Variation Diminishing” (TVD, [148]), y con ello sea estable. Entonces, dada la ecuación de advección 1D

$$\frac{\partial q}{\partial t} = -\frac{\partial (qu)}{\partial x} \quad (6.8)$$

su solución numérica está dada por:

$$\frac{q_i^{n+1} - q_i^n}{\Delta t} = -\frac{f_{i+\frac{1}{2}}^{n+\frac{1}{2}} - f_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} \quad (6.9)$$

donde q_i^n representa el valor de q en el centro de la celda i en el paso de tiempo n , y $f_{i\pm\frac{1}{2}}^{n+\frac{1}{2}}$ representan los flujos en los bordes de la celda i en el paso de tiempo $n + \frac{1}{2}$ y son calculados como sigue:

$$\begin{aligned} f_{i\pm\frac{1}{2}}^{n+\frac{1}{2}} &= 0.5 u_{i\pm\frac{1}{2}}^n \left[\left(1 \pm \theta_{i\pm\frac{1}{2}}^n\right) q_i^n + \left(1 \mp \theta_{i\pm\frac{1}{2}}^n\right) q_{i\pm 1}^n \right] \\ &\pm 0.5 \left| u_{i\pm\frac{1}{2}}^n \right| \left(1 - \left| c_{i\pm\frac{1}{2}}^n \right| \right) \phi \left(r_{i\pm\frac{1}{2}}^n \right) \left(q_{i\pm 1}^n - q_i^n \right) \end{aligned} \quad (6.10)$$

donde $c_i^n = u_i^n \Delta t / \Delta x$, $\theta_i^n = |u_i^n| / u_i^n$, r_i^n está definido mediante

$$r_i^n = \begin{cases} \frac{q_{i-\frac{1}{2}}^n - q_{i-\frac{3}{2}}^n}{q_{i+\frac{1}{2}}^n - q_{i-\frac{1}{2}}^n}, & \text{si } u_i^n > 0 \\ \frac{q_{i+\frac{3}{2}}^n - q_{i+\frac{1}{2}}^n}{q_{i+\frac{1}{2}}^n - q_{i-\frac{1}{2}}^n}, & \text{si } u_i^n < 0 \end{cases} \quad (6.11)$$

finalmente, $\phi(r)$ es el limitador de flujo, que para nuestro caso usaremos el limitador de flujo Superbee [147], dado por

$$\phi(r) = \begin{cases} 0 & \text{si } r < 0 \\ \min(2r, 1) & \text{si } 0 < r < 1 \\ \min(2, 1) & \text{si } r > 1 \end{cases} \quad (6.12)$$

Para evolucionar las ecuaciones se usa el integrador TVD Runge-Kutta de segundo orden explícito (TVD-RK2, [149]), que para la ecuación de advección 6.8, está dado por:

$$\begin{aligned} q_i^* &= q_i^n + \Delta t F(q_i^n) \\ q_i^{n+1} &= \frac{1}{2}q_i^n + \frac{1}{2}q_i^* + \frac{1}{2}\Delta t F(q_i^*) \end{aligned} \tag{6.13}$$

Finalmente, para resolver los términos fuente se usan diferencias finitas centradas de segundo orden, y, al igual que el problema de advección, el integrador TVD-RK2.

6.2.3. Código Boltzmann-PM

Este código acopla a los códigos Hidro-Boltzmann (resuelve descripción de medio continuo) y PM (resuelve la descripción de partículas prueba) mediante la ec. de Poisson (ec. 6.3), y con ello implementa el método de momentos de la ecuación de Boltzmann.

En este código, para evolucionar al fluido no-colisional (neutrinos) por un paso de tiempo se siguen los siguientes pasos:

1. Se resuelven las ecuaciones 6.4 y 6.6 usando el código Hidro-Boltzmann.
2. Se evoluciona a las partículas prueba con el código PM usando la densidad calculada en el paso previo para resolver la ec. de Poisson (ec. 6.3).
3. Se calcula el tensor de dispersión de velocidades (ec. 6.7) con las partículas prueba.
4. Actualizamos tiempo global de integración.

En la figura 6.2 se muestra el diagrama de flujo del código.

6.2.4. Código para dos especies

En los códigos anteriores, sólo se está considerando que hay una sola especie de partícula en la simulación, por lo que para las simulaciones de un modelo ν CDM se requiere que el código pueda evolucionar a dos especies de partículas diferentes.

Por lo anterior, extendí el código Boltzmann-PM para que pueda manejar más de una especie en la simulación, asumiendo que la única interacción entre ambas especies es gravitacional.

Ahora bien, para hacer la extensión del código Boltzmann-PM se tienen dos casos (cada uno de ellos representa una versión del código extendido). El primer caso es aplicar el método de momentos de Boltzmann para ambas especies, es decir, representar a cada una de las especies como un fluido no-colisional, por lo que la evolución de cada una de las especies es seguida, independientemente, con el código Boltzmann-PM y únicamente quedan acopladas mediante la ecuación de Poisson modificada para dos especies (se agrega el término de densidad de cada especie). Me referiré a esta versión del código como “BPM-BPM”.

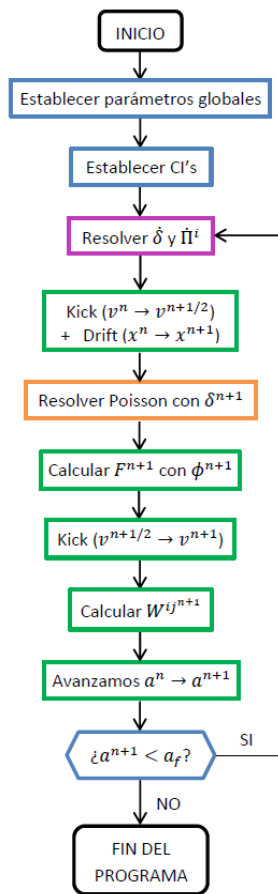


Figura 6.2: Diagrama de flujo código Boltzmann-PM.

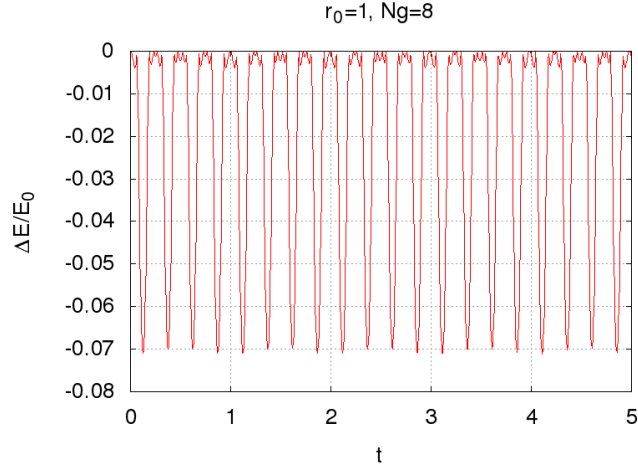


Figura 6.3: Error fraccional de la energía para 2 cuerpos con $r_0 = 1$ celda

El otro caso es aplicar el método de momentos únicamente para una de las especies (p. ej. neutrinos) y para la otra especie (p. ej. CDM) usar N -cuerpos, es decir, la evolución de la primer especie es seguida con el código Boltzmann-PM y la evolución de la segunda sólo con el código PM, y, al igual que en BPM-BPM, ambas evoluciones quedan acopladas únicamente mediante la ecuación de Poisson. Me referiré a este caso como “BPM-PM”.

6.3. Pruebas numéricas de los códigos

En esta sección mostraré algunas de las pruebas numéricas realizadas para cada uno de los códigos desarrollados.

6.3.1. Pruebas numéricas para PM

Las pruebas numéricas realizadas para este código son las siguientes:

- * **2-cuerpos:** la solución al problema de 2-cuerpos representa una prueba natural para los códigos de N -cuerpos, ya que se conoce la solución analítica del problema. Para esta consideré a dos cuerpos de igual masa en órbita circular. En las figuras 6.3 y 6.4 se muestra el error fraccional de la energía para una separación inicial $r_0 = 1$ y 2 celdas, respectivamente, el número de celdas en la malla es 8^3 y 16^3 celdas, respectivamente. Para esta prueba consideré un paso de integración $dt = 0.002$ y se siguió la evolución de los dos cuerpos hasta $t_f = 5.0$, ambos en unidades adimensionales (en las que $dL = 1$, $t_{\text{dyn}}(r_0) = 1$, y $\rho_T = 1$). De estas figuras podemos observar que hay conservación de la energía total del sistema, lo cual es de esperarse, y que el error fraccional de ésta disminuye para el segundo caso (figura 6.4). Esto último se entiende ya que este segundo caso correspondería a aumentar la resolución del primer caso (figura 6.3) por un factor 2, y por ello el error disminuye a aproximadamente la mitad.

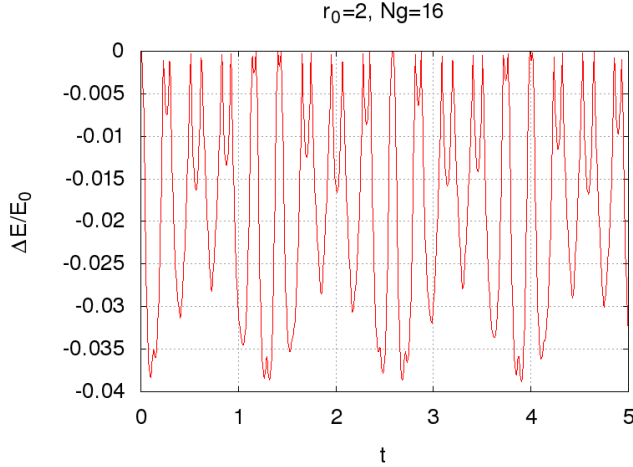


Figura 6.4: Error fraccional de la energía para 2 cuerpos con $r_0 = 2$ celdas.

- * **Plummer aislado:** esta prueba consiste en seguir la evolución de un sistema de N -cuerpos aislado con condición inicial dada por una esfera de Plummer [150]. Dado que el sistema es aislado y se encuentra en equilibrio, entonces el perfil de densidad del sistema, así como la energía total se deben conservar en el tiempo. Para esta prueba, consideré un sistema con $N = 16^3$ partículas de igual masa, el radio de escala del perfil de Plummer es $r_s = 3$, el radio del sistema es $R_{\max} = 11.1$, el paso de integración es $dt = t_{\text{dyn}}(r_s)/100$, y la evolución del sistema se siguió hasta $t_f = 5.0$, todo en unidades adimensionales (en las que $dL = 1$, $t_{\text{dyn}}(r_h) = 1$, y $\bar{\rho}(r_h) = 1$, con r_h el radio de masa media). En la figura 6.5 se muestra la evolución del perfil de densidad así como para el error fraccional de la energía total del sistema considerando 32^3 (columna izquierda) y 64^3 (columna derecha) celdas para la malla. De esta figura podemos observar que tanto el perfil de densidad como la energía total del sistema se conservan en el tiempo, siendo menor el error para el caso 64^3 .
- * **Pancake de Zeldovich:** esta prueba se refiere al colapso en 1D de una onda plana en un universo en expansión, la cual representa una de las pruebas tradicionales para los códigos de N -cuerpos en cosmología [30, 32]. En esta prueba, la solución analítica [85] es usada para verificar la precisión del código. Si se conoce la condición inicial, la solución analítica de las posiciones, para cualquier otro momento de tiempo es:

$$x_i(a) = q_i + D_+(a)A \sin(2\pi q_i/\lambda) \quad (6.14)$$

En donde q_i son las posiciones iniciales (sin perturbar) de las partículas, λ es la longitud de onda, $A = \lambda/(2\pi D_+(a_c))$ es la amplitud, con a_c el factor de escala al tiempo de cruce y $D_+(a)$ es la función de crecimiento lineal. Para un universo Einstein-de-Sitter ($\Omega_m = 1$), se tiene $D_+(a) = a$. Las velocidades de las partículas se obtienen diferenciando la expresión anterior. Para esta prueba usé $N = 64^3$ partículas en una malla de 32^3 celdas. El sistema se evolucionó desde $a = 0.150$

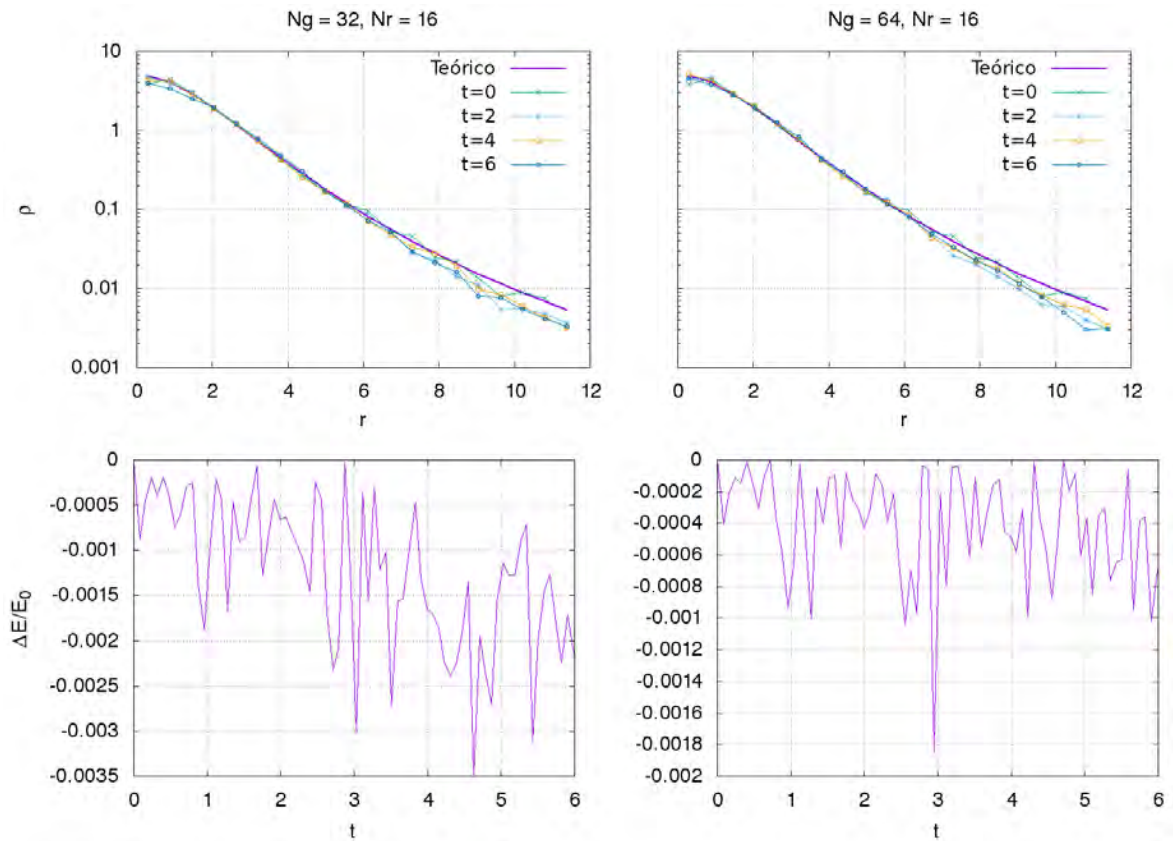


Figura 6.5: Evolución del perfil de densidad (primer renglón) y del error de la energía para la esfera de Plummer con $N = 16^3$ partículas y 32^3 (columna izquierda) y 64^3 (columna derecha) celdas.

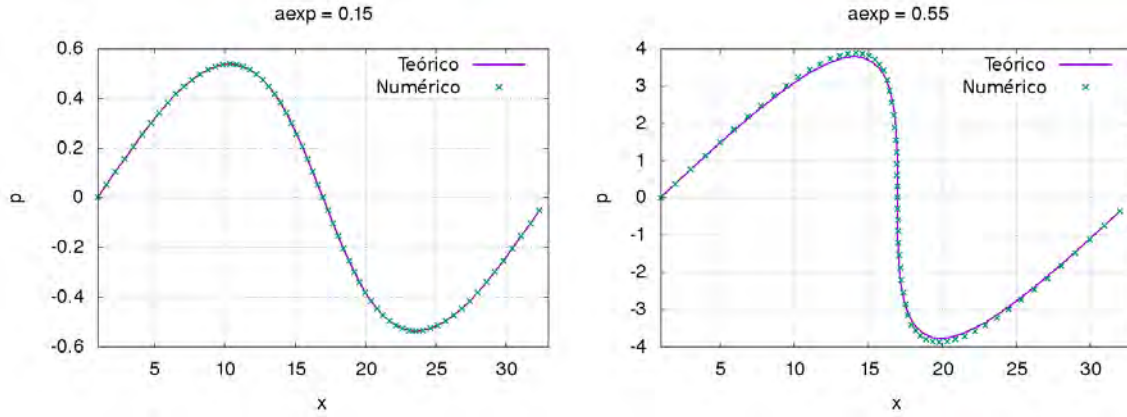


Figura 6.6: Espacio fase de la condición inicial (izquierda) y el snapshot final (derecha) para la solución numérica (cruces) y analítica (línea) del “pancake de Zeldovich”.

($z \approx 5.7$) hasta $a_c = 0.550$ ($z \approx 0.8$) con un paso de integración $da = 0.008$. Los parámetros cosmológicos de la prueba son los de un universo Einstein-de-Sitter: $\Omega_m = 1$, $h_s = 0.7$. En la figura 6.6 se muestran el espacio fase de la condición inicial (en $a = 0.150$, a la izquierda) y el snapshot final (en $a_c = 0.550$, a la derecha) para la solución numérica (cruces) y analítica (línea) del “pancake de Zeldovich”. En esta figura podemos observar que la solución numérica es bastante buena pues se asemeja bastante a la solución analítica.

6.3.2. Pruebas numéricas para Hidro-Boltzmann

Las pruebas numéricas realizadas para este código son las siguientes:

- * **Advección 1D:** esta prueba considera el desplazamiento temporal de una función $q(x)$ a lo largo del eje X con una velocidad constante $u = 1$, y cuya condición inicial está dada por:

$$q(x, t = 0) = 1, \text{ si } x \leq 30; 0 \text{ si } x > 30 \quad (6.15)$$

Y con condiciones de frontera periódicas. La evolución de la función $q(x, t)$ está dada por la ecuación de advección 1D. Por lo que esta prueba es una prueba muy sencilla para probar que los esquemas numéricos para la solución de la ec. de advección (sin fuente) funcionan correctamente. Para esta prueba consideré $x \in [0, 100]$, una malla 1D con 100 celdas, por lo que el tamaño de una celda es $dL = 1$. Seguí la evolución de la función desde $t = 0$ hasta $t_f = 70$, con un paso de tiempo $dt = 0.2$. En la figura 6.7 se muestran tres snapshots de la evolución de la función $q(x, t)$: uno en $t = 0$ (al inicio de la evolución), uno en $t = 35$ (a la mitad de la evolución), y el otro en $t = 70$ (al final de la evolución). De esta gráfica podemos ver que la amplitud de la función se mantiene constante en el tiempo, lo cual es de esperarse, pues la ecuación de advección (sin términos fuente) describe únicamente el desplazamiento de la función en la dirección de la

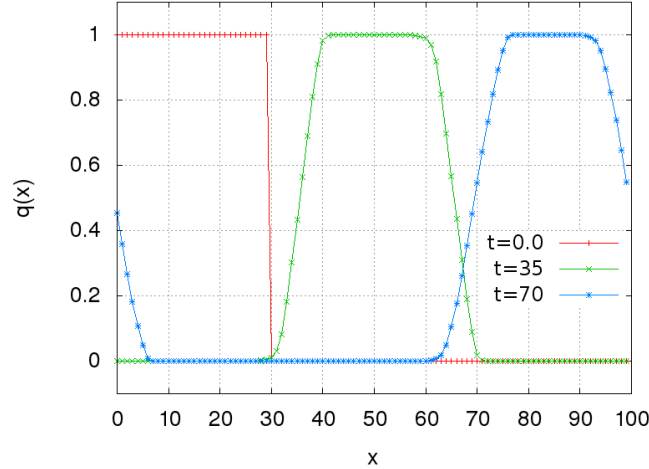


Figura 6.7: Evolución advectiva de la función $q(x, t)$.

velocidad, y por tanto se debe conservar la amplitud de la función inicial, pues no hay ningún término difusivo.

- * **Blob gaussiano 1D:** en esta prueba consideré una perturbación gaussiana en la densidad en 1D, cuya evolución está dada por las ecuaciones de continuidad y Euler (sin expansión del Universo) en 1D. La condición inicial está dada por:

$$\rho(x, t = 0) = 1 + 0.3 \exp\left(-\frac{(x - 50)^2}{100}\right), \quad x \in [0, 100] \quad (6.16)$$

Y con condiciones de frontera periódicas. Dado que en esta prueba se resuelven las ecuaciones de continuidad y Euler (o conservación de momento) en 1D, entonces permite verificar que el método de “división de operadores” se está implementando correctamente en el código “Hidro-Boltzmann”, ya que estas ecuaciones son de advección (en 1D) pero con término fuente (en la ecuación de Euler). Para esta prueba consideré una malla 1D con 64 celdas, por lo que el tamaño de una celda es $dL \approx 1.6$. Seguí la evolución de la función desde $t = 0$ hasta $t_f = 400$, con un paso de tiempo $dt = 0.4$. En la figura 6.8 se muestran diferentes snapshots de la evolución de la perturbación: el primero es en $t = 0$ y corresponde a la condición inicial. Los siguientes son $t = 1.6, 4.0, 6.4$ en y 7.6 , y corresponden a puntos intermedios en la evolución. De estos snapshots podemos ver que la evolución de la perturbación es como sigue: el “blob” gaussiano en la densidad se divide en dos “blobs” iguales entre sí, uno que se mueve hacia la izquierda y otro que se mueve hacia la derecha. Continúan moviéndose de esta forma hasta que llegan a las paredes del dominio (es decir, a la frontera) en $t = 7.6$. A partir de aquí, ambos “blobs” se reflejan y regresan hacia el centro del dominio juntándose nuevamente en el “blob” inicial (esto último no se muestra en la figura 6.8 por claridad). Este patrón de onda se repite muchas veces.

- * **Plummer advectivo:** consiste en seguir la evolución temporal de un perfil de Plummer, el cual está en equilibrio dinámico y aislado, y se le da una velocidad

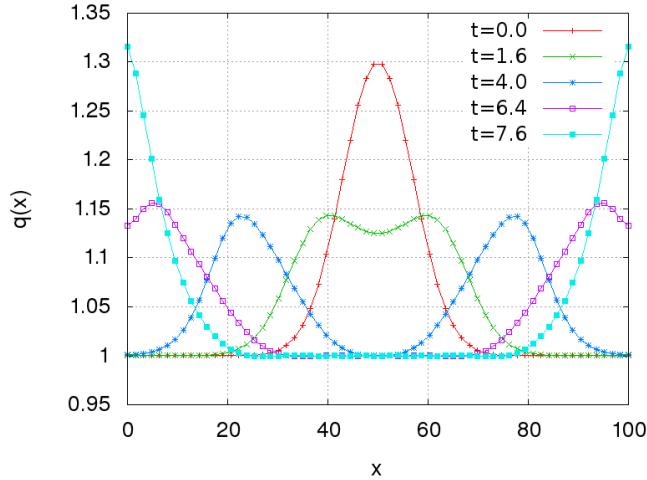


Figura 6.8: Evolución de la perturbación gaussiana 1D.

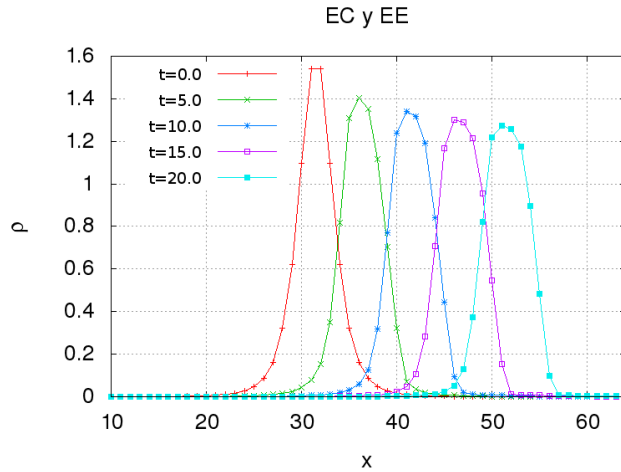


Figura 6.9: Evolución del perfil de densidad para un halo Plummer advectivo.

de advección extra a las partículas, lo cual va a tener el efecto de mover el perfil de densidad en la dirección de esta velocidad. En la figura 6.9 se muestra el perfil de densidad para distintos momentos de la evolución. De aquí podemos ver que hay difusión en la solución numérica, pero la magnitud y dirección con la que se mueve el perfil de Plummer es consistente con la velocidad de advección inicialmente asignada.

6.3.3. Pruebas numéricas para Boltzmann-PM

Las pruebas numéricas realizadas para este código son las siguientes:

- * **Convergencia:** en esta prueba evolucionamos un modelo Λ CDM, desde $z_0 = 20$ hasta $z = 0$, utilizando una caja de $32h^{-1}\text{Mpc}$ con $N = 64^3$ partículas y con una malla de 64^3 celdas. En esta prueba, evolucionamos varias veces al modelo, pero

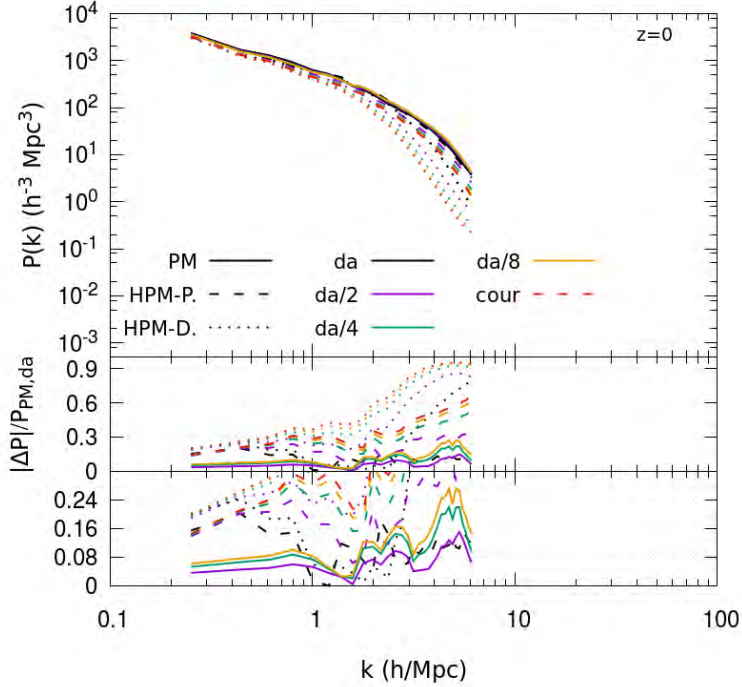


Figura 6.10: Prueba de convergencia temporal del código Boltzmann-PM, considerando un modelo Λ CDM.

cada vez cambiando el paso de integración con el propósito de verificar que el código si tuviera convergencia “temporal”. En la figura 6.10 se muestra el MPS, a $z = 0$, de los resultados, y se compara con el código PM. De la figura vemos que el código si converge.

- * **Efecto de la malla elegida:** para esta prueba también evolucionamos un modelo Λ CDM, desde $z_0 = 20$ hasta $z = 0$, utilizando una caja de $32h^{-1}\text{Mpc}$ con $N = 64^3$ partículas. En esta prueba, evolucionamos varias veces al modelo, pero cada vez cambiando el número de celdas en la malla con el propósito de verificar el código si tuviera convergencia “espacial”. En la figura 6.11 se muestra el MPS, a $z = 0$, de los resultados, y se compara con el código PM. De la figura vemos que para un número mayor de celdas que de partículas en la caja Boltzmann-PM se ve más afectado, lo cual tiene sentido, pues en este caso hay un mayor número de celdas vacías, y las ecuaciones usadas para éste código son más sensibles a esto, que el código PM.
- * **Supresión de shot-noise:** en esta prueba evolucionamos un modelo Λ CDM, desde $z_0 = 49$ hasta $z = 0$, utilizando una caja de $256h^{-1}\text{Mpc}$ con $N = 256^3$ partículas y con una malla de 256^3 celdas. El propósito de esta prueba fue ver como se compara el código Boltzmann-PM respecto al PM, en simulaciones más realistas. En la figura 6.12 se muestra el MPS, a $z = 0$, de los resultados. De la figura vemos que hay diferencias entre el 10% y 15% entre ambos códigos para

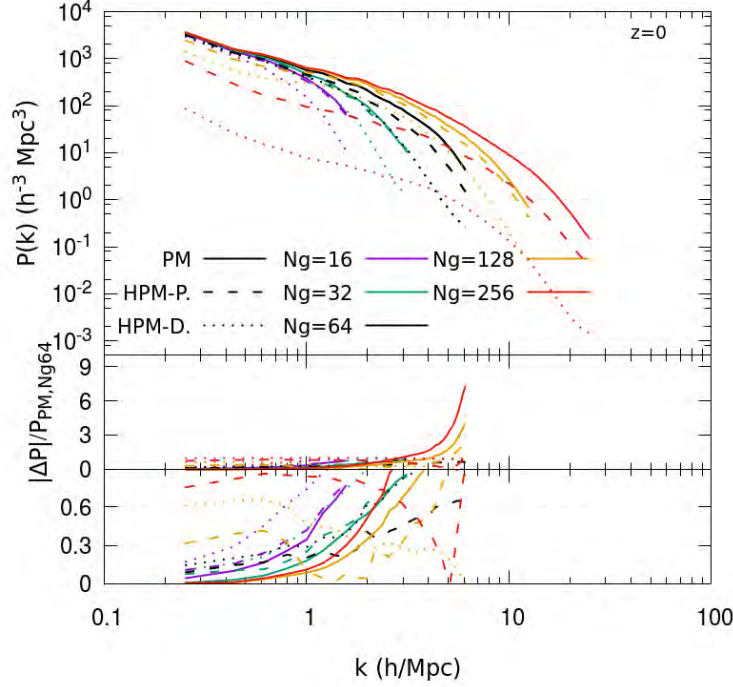


Figura 6.11: Prueba de convergencia espacial del código Boltzmann-PM, considerando un modelo Λ CDM.

escalas pequeñas, y como el Boltzmann-PM en principio no está afectado por shot-noise, no hay un aumento de potencia para escalas pequeñas a diferencia del PM.

6.3.4. Pruebas numéricas para 2-especies

Las pruebas numéricas realizadas para este código son las siguientes:

- * Λ CDM: para esta prueba evolucionamos un modelo Λ CDM con el código, separando aleatoriamente a las partículas en dos grupos, para tratar a cada grupo como si fueran especies diferentes al nivel del código. Dado que no hay ninguna diferencia física entre las partículas, entonces, al evolucionar el sistema con el código deberíamos obtener los mismos resultados que al correr el sistema con el código de una sola especie. En la figura 6.13 se muestra el MPS a $z = 0$ con los resultados obtenidos, y se compara con la versión de una sola especie. De la figura vemos que los resultados son consistentes entre sí, como era de esperarse.
- * ν CDM: para esta prueba consideramos un modelo $\nu\Lambda$ CDM con 512^3 partículas de cada especie en una caja de 1 Gpc de lado, e iniciamos la simulación en $z = 127$. Los parámetros cosmológicos utilizados son: $\{\Omega_m, \Omega_b, h, n_s, \sigma_8, M_\nu, w\} = \{0.3175, 0.049, 0.6711, 0.9624, 0.834, 0.4, -1\}$. Para esta prueba utilizamos la versión del código BPM-PM, en donde el BPM se usa para los neutrinos y el PM para

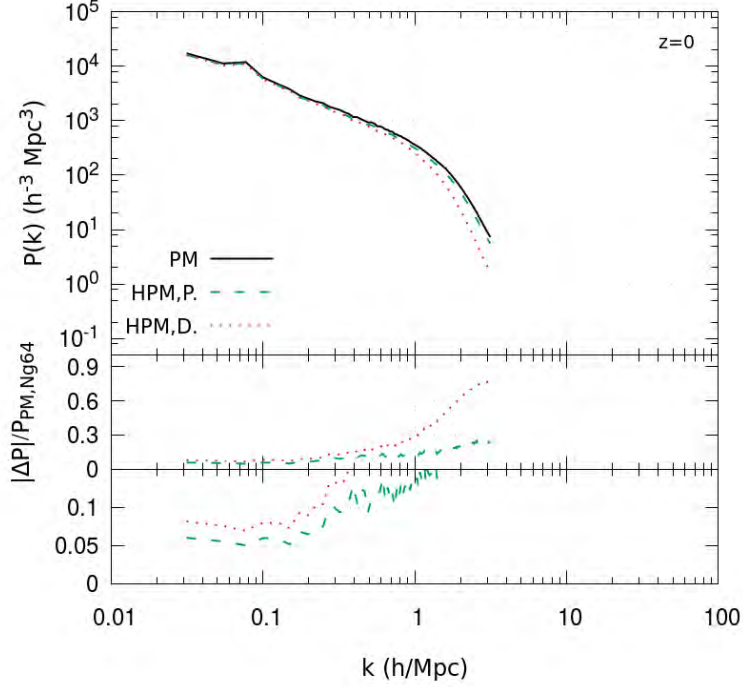


Figura 6.12: Prueba de que Boltzmann-PM suprime el shot-noise presente en los códigos de N -cuerpos como el PM.

CDM. Para fines comparativos, realizamos tres simulaciones diferentes (usando las mismas condiciones iniciales). La primera considera una malla de 256 celdas por lado y se usa únicamente la ecuación tipo continuidad (ec. 6.4) en la jerarquía de Boltzmann, llamaremos a esta prueba como EC-256. En la segunda simulación, se usó la misma malla que en la primera, pero ahora se usaron tanto la ec. tipo continuidad como la tipo Euler (ecs. 6.4 y 6.6) en la jerarquía de Boltzmann, nos referiremos a esta prueba como ECyEE-256. Finalmente, en la tercera simulación consideramos una malla de 512 celdas por lado y, al igual que en la primera simulación, se usa únicamente la ecuación tipo continuidad, nombraremos a esta prueba como EC-512. La figura 6.14 muestra una misma rebanada (espacialmente) del campo de densidad de los neutrinos (escala de color en tonos rojos) y de la componente CDM (escala de color en tonos azules) para las tres simulaciones descritas previamente: EC-256 (primeros dos renglones), ECyEE-256 (dos renglones de en medio) y EC-512 (últimos dos renglones). La figura 6.15 muestra el espectro de potencias de materia (MPS) obtenido para las tres simulaciones realizadas a distintos corrimientos al rojo, z . De la figura se aprecia que la diferencia más notoria para el MPS total ($P_{\nu+\text{cdm}}$) es por efectos de resolución a escalas pequeñas, pues el tener una malla más fina (simulación EC-512) permite resolver escalas más pequeñas. Para el MPS de la componente CDM (P_{cdm}) se tiene un resultado similar al anterior. En contraste, el MPS de la componente de neutrinos (P_{ν}) presenta pequeñas diferencias para las distintas

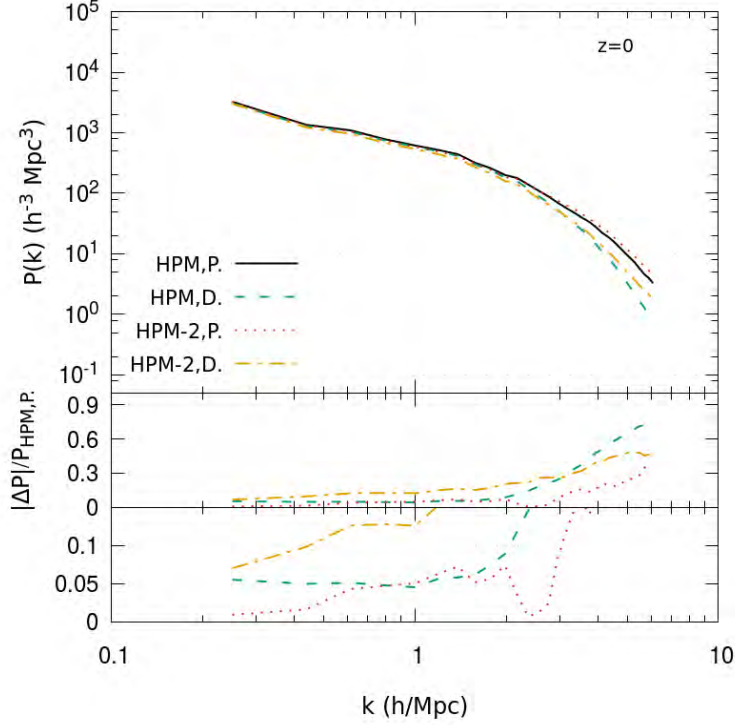


Figura 6.13: Comparación del MPS a $z = 0$ para el modelo Λ CDM comparando los códigos BPM-BPM y Boltzmann-PM.

simulaciones en escalas grandes (k 's pequeñas). En particular, hay una diferencia notoria entre la simulación EC-256 y ECyEE-256 (a diferencia de los otros dos MPS), lo cual es de esperarse pues con la ecuación tipo Euler se captura con mejor precisión la dinámica de los neutrinos. En la figura 6.16 se muestra una comparación entre las simulaciones EC-256 y ECyEE-256 para los mismos MPS y los mismos instantes que la figura 6.15. De esta figura podemos apreciar mejor que sí existe una diferencia entre las dos simulaciones para las tres componentes ($P_{\nu+\text{cdm}}$, P_{cdm} , P_{ν}), la cual es mayor en escalas pequeñas (k 's grandes), y además crece con el tiempo. Como notamos anteriormente, para $P_{\nu+\text{cdm}}$ y P_{cdm} la diferencia entre las simulaciones es pequeña pero existente, mientras que para P_{ν} es más notoria. El hecho de que P_{cdm} no sea tan sensible (como P_{ν}) a si se usa sólo la ecuación tipo continuidad o si se usan tanto la tipo continuidad como la tipo Euler, es debido a que la componente CDM se resuelve con un PM, además de que el efecto en P_{ν} no es tan dominante. Lo mismo ocurre con $P_{\nu+\text{cdm}}$ pues P_{cdm} domina dicho MPS.

6.4. Potencial del código Boltzmann-PM

Es importante hacer notar que las técnicas usadas en el código Boltzmann-PM pueden aplicarse para estudiar modelos cosmológicos diferentes al ν CDM, desde un

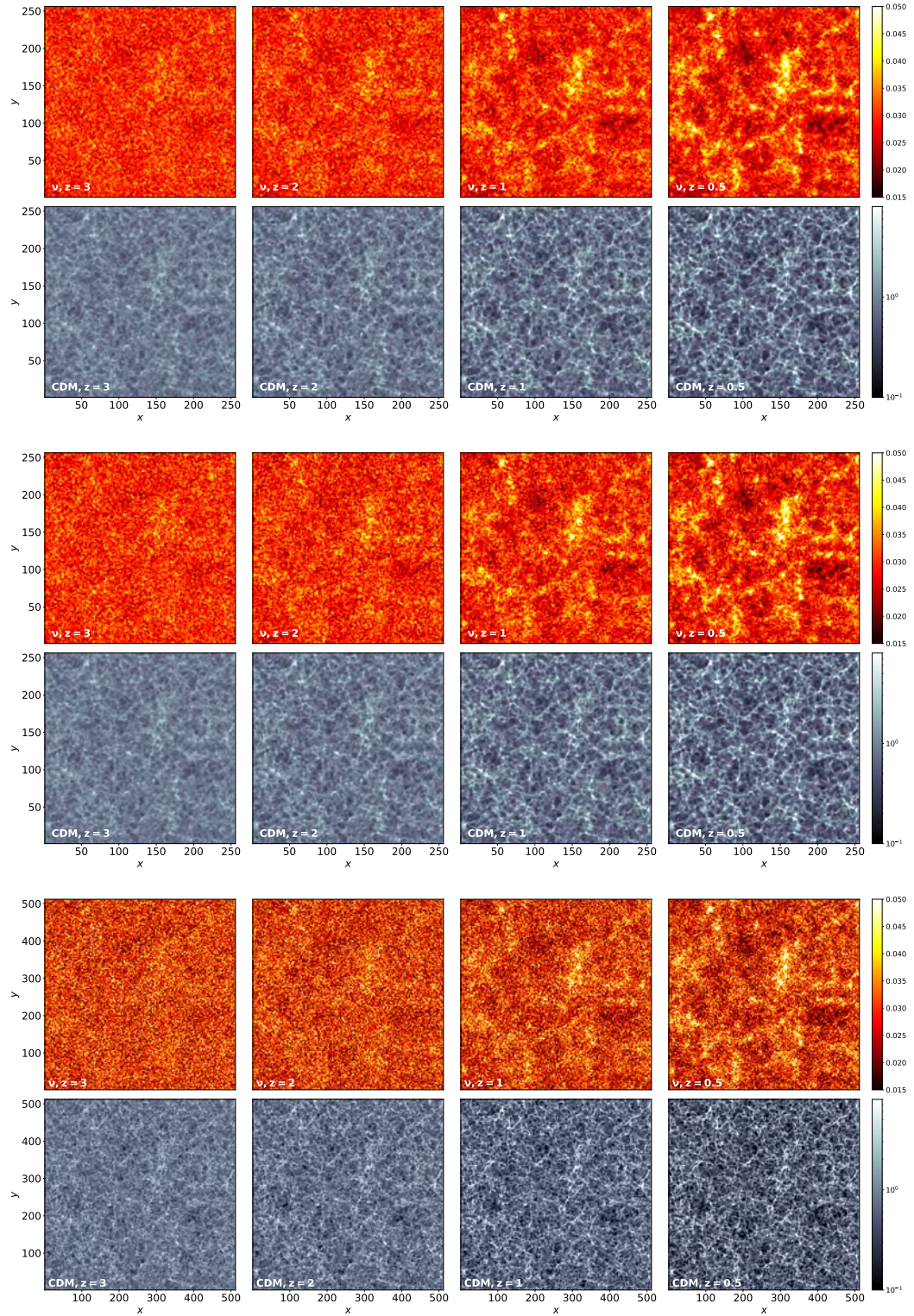


Figura 6.14: Estructura en simulaciones $\nu LCDM$ con BPM. Rebanada (en el plano XY) del campo de densidad de neutrinos (escala de color roja) y CDM (escala de color azul) para cuatro corrimientos al rojo: $z = 3, 2, 1$ y 0.5 . Las primeras dos filas corresponden a la simulación EC-256, las siguientes dos a ECyEE-256, y las últimas dos a EC-512.

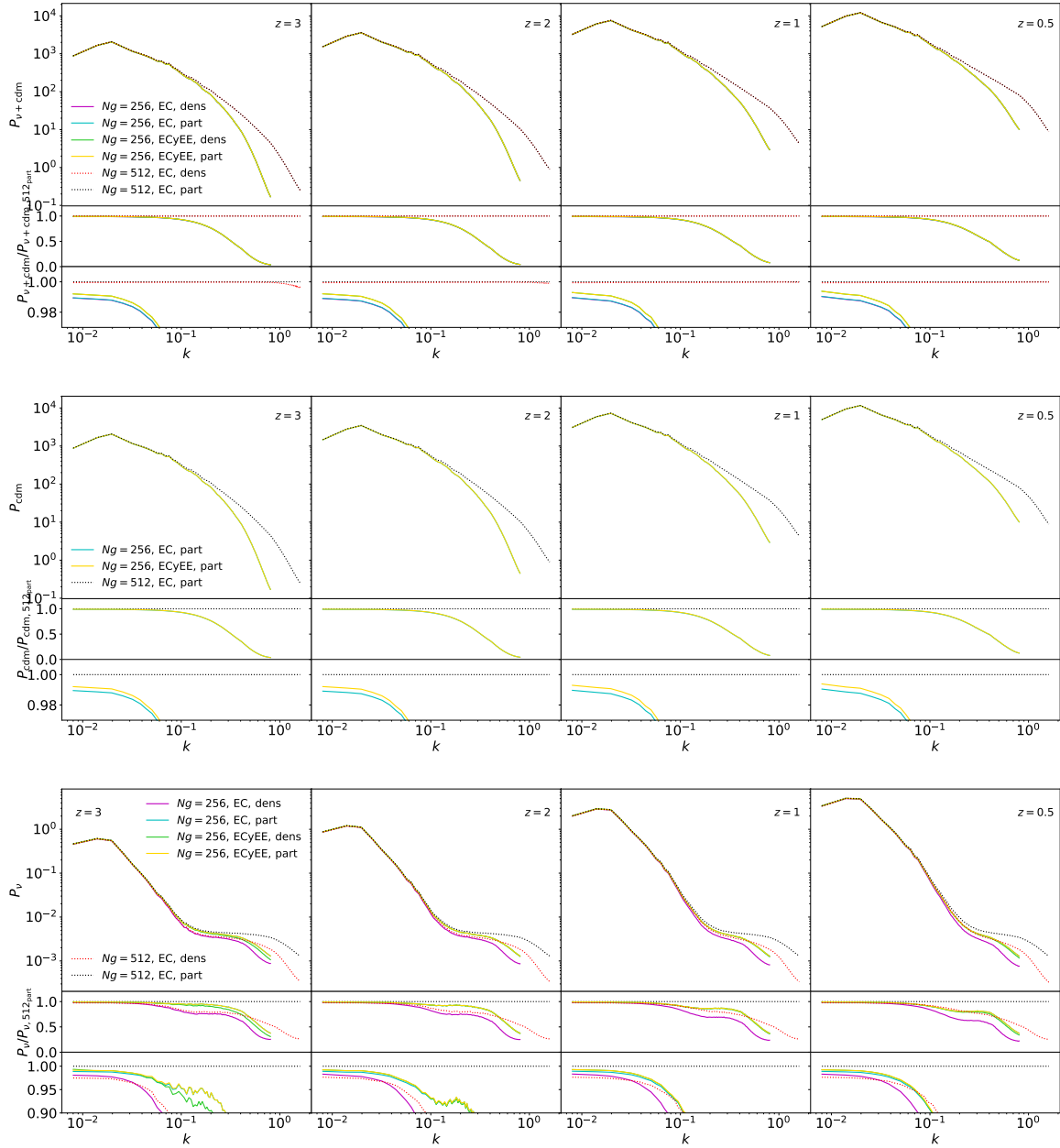


Figura 6.15: MPS de las componentes total (ν +CDM, primera fila), CDM (fila central), y neutrinos (fila de abajo) en cuatro corrimientos al rojo, $z = 3, 2, 1$ y 0.5 , para las tres simulaciones EC-256, ECyEE-256 y EC-512. La diferencia más notoria para $P_{\nu+\text{cdm}}$ y P_{cdm} se da por efecto de resolución en escalas pequeñas. P_{ν} es un poco más sensible a las diferencias entre las simulaciones.

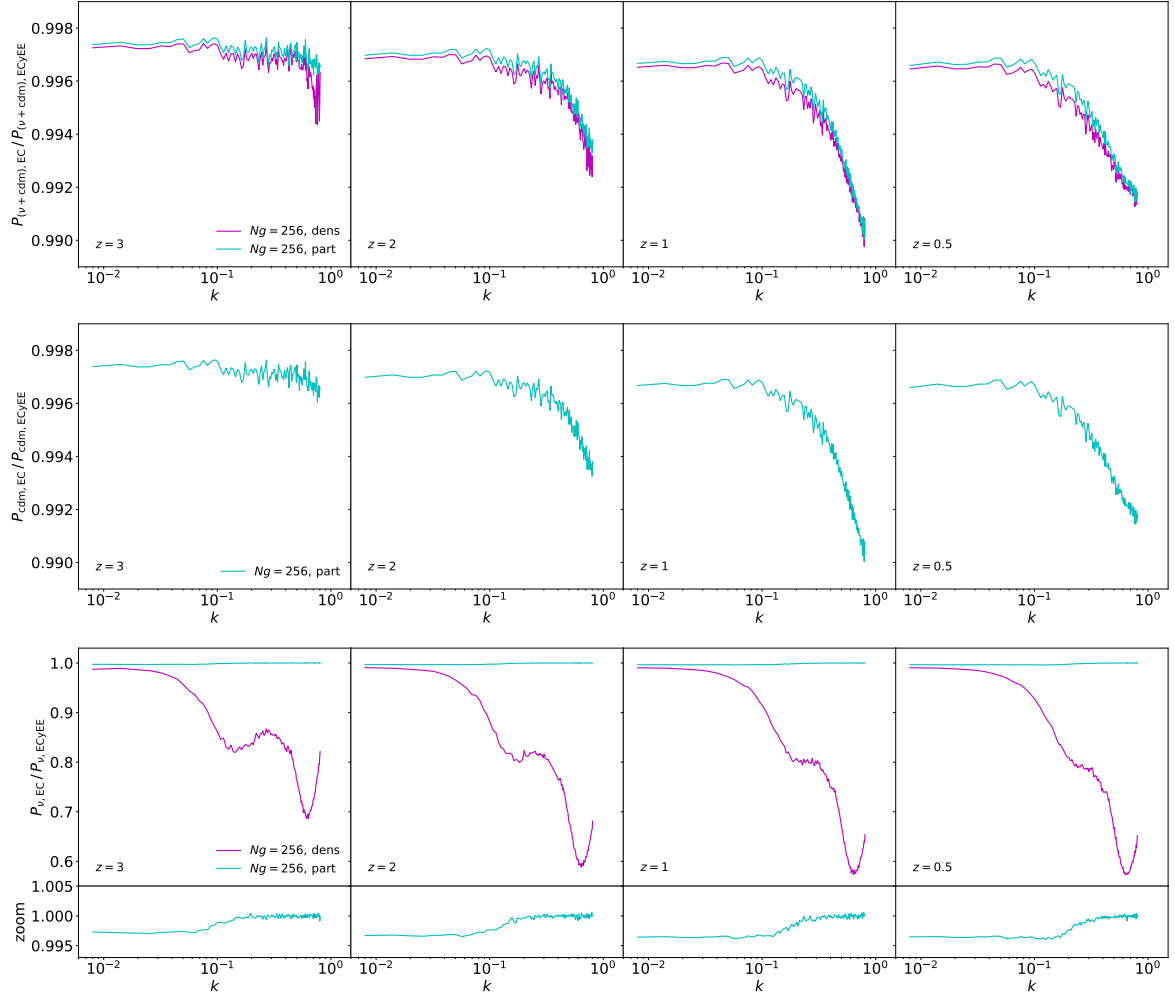


Figura 6.16: Comparación del MPS entre las simulaciones EC-256 y ECyEE-256 para las tres componentes: total (ν +CDM, primera fila), CDM (fila central), y neutrinos (fila de abajo) en para los mismos z 's que las figura 6.15. La diferencia entre EC-256 y ECyEE para $P_{\nu+cdm}$ y P_{cdm} es pequeña pero existe. P_{ν} es más sensible a las ecuaciones que se resuelven.

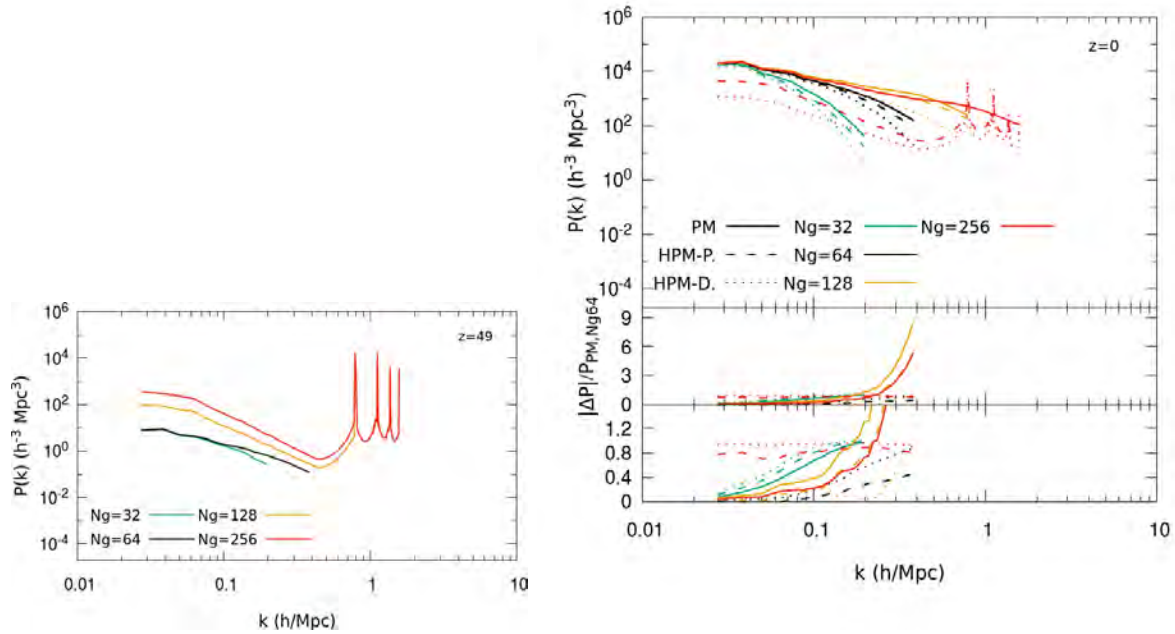


Figura 6.17: Boltzmann-PM preserva la estructura característica del MPS, mientras que PM no.

punto de vista teórico, como WDM o energía oscura dinámica, pero con ecuaciones similares. Por lo tanto, el código Boltzmann-PM es una buena herramienta que puede ser usada para estudiar modelos con supresión en el MPS, lo que resulta difícil para códigos de N -cuerpos debido al shot-noise.

En la figura 6.17 se muestra un ejemplo de la ventaja que tiene el código Boltzmann-PM sobre el PM. Para esta figura evolucionamos un modelo ficticio que inicialmente tiene estructura en su MPS y comparamos el MPS obtenido por ambos códigos al final de la evolución. Vemos que Boltzmann-PM preserva las características iniciales del MPS, mientras que el PM las borra. Este modelo ficticio, fue inspirado en los modelos de materia oscura ultraligera.

El código también puede utilizarse para varios problemas dinámicos en astrofísica, como discutiremos en los siguientes capítulos.

Capítulo 7

Simulando la respuesta de halos oscuros a la acreción de galaxias satélites con BPM

Los modos de densidad inducidos en el halo estelar de galaxias y que reflejan los de la respuesta del halo de materia oscura, comparten el ser de muy baja amplitud de forma análoga con la estructura en simulaciones cosmológicas con neutrinos masivos. Una demandante y excelente prueba independiente de la cosmología a gran escala, es la respuesta en densidad y velocidad en sistemas que han sufrido la caída de un satélite masivo como la nube mayor de Magallanes (LMC, por sus siglas en inglés). Es por esto que en el presente capítulo estudiamos dichos sistemas dinámicos usando el código BPM. Este estudio es otro aporte de la presente tesis, pues la técnica de momentos de Boltzmann implementada en BPM no había sido usada con este propósito.

7.1. Introducción: Modos de respuesta en halos de galaxias, fricción dinámica.

Es bien conocido que el proceso de acreción de galaxias satélites a galaxias como la Vía Láctea (MW, por sus siglas en inglés) estimula modos de respuesta en los halos de galaxias, esta respuesta depende de las propiedades del medio como dispersión de velocidades o de la velocidad del sonido en un hipotético halo oscuro. Recientemente se anunció la detección de la estela (o “wake”, en inglés) local y de un modo dipolar (respuesta global) en el halo estelar de MW, presumiblemente resultado de la interacción con la LMC. Capturar tales respuestas en simulaciones de N -cuerpos suele requerir cientos [151] o miles de millones de partículas [152]. Nuestro código BPM, en su versión aislada, lo puede hacer a un costo computacional modesto ($N = 128^3$ o 256^3 , ver figura 7.1). En el presente capítulo ilustramos la dependencia de la respuesta en halos de materia oscura que acretan una galaxia satélite del 10% y que comparten similar curva de velocidad circular pero tienen diferente perfil de densidad, lo cual no se había realizado aún en la comunidad debido al gran costo computacional. La respuesta es sensible a las diferencias de los modelos, ofreciendo información adicional que combi-

nada a futuro con otras restricciones podrá constreñir la naturaleza de los halos oscuros o de su componente, la materia oscura.

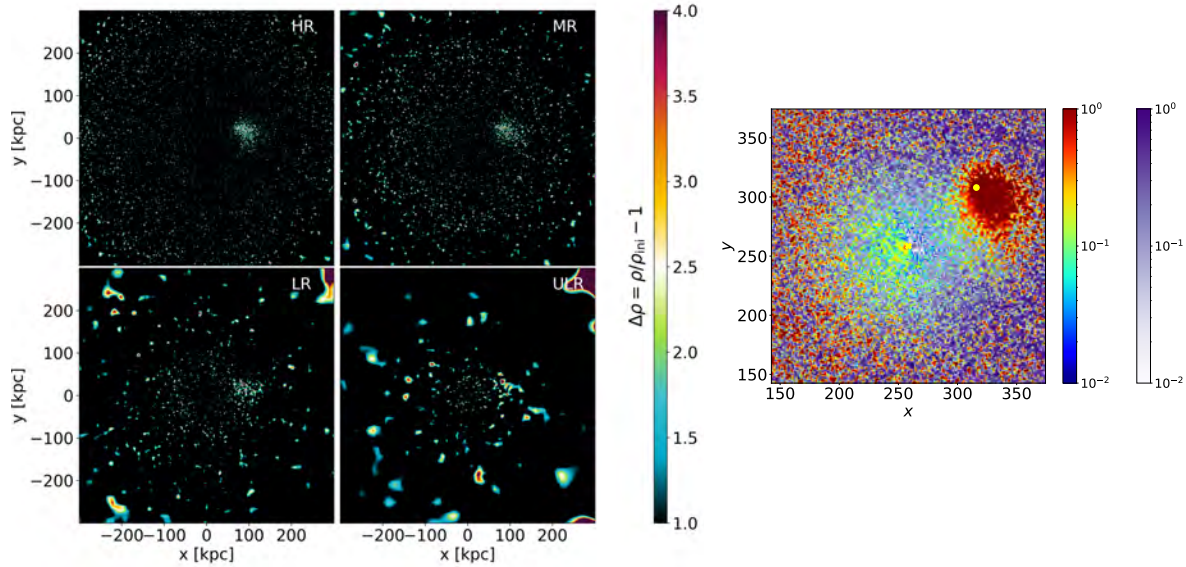


Figura 7.1: *Izquierda*: partículas coloreadas por la sobre(sub)-densidad en un halo de materia oscura de tamaño galáctico utilizando un código de árbol, los modos son inducidos por un satélite cuyas partículas no se muestran [152]. El caso de alta resolución (HR) incluye 300 millones de partículas, el de media (MR) 30 millones, el de baja (LR) 4 millones y el de ultra baja (ULR) 300 mil. Todas las mediciones se realizaron en el plano orbital del satélite con espesor de 3 kpc, coloreando solo partículas con sobre-densidad mayor a dos para resaltar el contraste. *Derecha*: mapa de sobre(sub)-densidad para un sistema comparable realizado con BPM y 256^3 partículas en el plano orbital con una celda de espesor (~ 1 kpc). La figura ilustra la eficiencia de BPM para señales tenues como estos modos o la estructura en colas estelares de marea en galaxias enanas. Cabe mencionar que la representación gráfica no realiza ningún filtrado como en el panel superior, y además la escala de color esta en escala logarítmica.

A Boltzmann Poisson like Method to Simulate Galactic Halos Response to Satellite Accretion

Dependence on halo density profile

G. Aguilar-Argüello¹, O. Valenzuela¹, and A. Trelles¹

Universidad Nacional Autónoma de México, Instituto de Astronomía, AP 70-264, CDMX 04510, México
e-mail: gaguilar@astro.unam.mx, octavio@astro.unam.mx

May 8, 2022

ABSTRACT

Context. Recent studies reported the detection of the galactic stellar halo wake and dipole triggered by the Large Magellanic Cloud (LMC), mirroring the Dark Matter (DM) corresponding response. These studies open the possibility of adding constraints to the global mass distribution of the Milky Way (MW) and even the DM nature with current and upcoming stellar surveys, reigniting the discussion about the response modes in dynamical friction. However, the simulation of such features is computationally challenging.

Aims. Using a continuous medium approach, we investigate the density and velocity response modes in simulations of Galactic-type DM halos accreting LMC-sized satellites, including the dependence on the halo density profile.

Methods. Using for the first time in the context of galactic dynamics a Collisionless Boltzmann Equation (CBE)+Poisson solver based on the method of Banerjee & Dalal (2016), we study the dynamical density and velocity response of halos to sinking perturbers.

Results. We successfully capture both the local wake and the global over/under-density induced in the host halo. We also capture the velocity response. Consistent with previous studies, the code can reproduce the core formation in the cuspy profile and the satellite core stalling. The Angular Power Spectrum (APS) response shows to be sensitive to each density profile. The cored Plummer density profile seems the most responsive, displaying a richness of modes. At the end of the simulation, the central halo acquired cylindrical rotation. If present, a stellar component will behave similarly.

Conclusions. The CBE description makes it reachable to capture the response modes with better handling of noise in comparison to traditional N -body simulations. Hence, given a noise level, BPM has a lower computational cost than N -body simulations, making it feasible to explore large parameter sets. We anticipate that stellar spheroids in the MW or external galaxies could show central cylindrical rotation if they underwent a massive accretion event. The code can be adjusted to include different dark matter physics.

Key words. Galaxy: kinematics and dynamics – (Cosmology:) dark matter – Methods: numerical

1. Introduction

The current paradigm for cosmic structure formation is the Λ Cold Dark Matter (Λ CDM) scenario. This scenario considers collisionless cold Dark Matter (DM) and a cosmological constant (Planck Collaboration et al. 2020). Galaxies' dark halo structure has been extensively studied in Λ CDM-related models, most frequently with rotation curves, disk dynamics, or galaxy scaling relations (Rodríguez-Puebla et al. 2016; Aquino-Ortíz et al. 2018). The dynamics of satellite galaxies are also critical tests (Wang et al. 2018), particularly for nearby galaxies (Cautun et al. 2020). Recent and upcoming surveys like GAIA (Gaia Collaboration et al. 2021), APOGEE (Majewski et al. 2017; Fernández-Trincado et al. 2020), DESI (Cooper 2021), WEAVE (Dalton et al. 2012), and LSST (Rich 2018) will produce an exquisite mapping of the Milky Way's (MW) stellar halo. In addition, they will open up the opportunity to track the subtle stellar response that is presumably paired with the DM halo response, constraining processes like dynamical friction and halo or even DM properties, as was recently shown observationally by Conroy et al. (2021). Using high-resolution N -body simulations, Garavito-Camargo et al. (2021) investigated the halo response modes to satellite accretion following pioneer studies like Weinberg (1989), Ogiya & Burkert (2016); Tamfal et al. (2021)

used super high-resolution simulations to study the relative importance of global mode and local wake in dynamical friction.

The recent studies mentioned above (Ogiya & Burkert 2016; Garavito-Camargo et al. 2021; Tamfal et al. 2021) revealed a rich response mode population in fully self-consistent calculations reaching particle numbers of around $10^8 - 10^9$, using large computational resources. However, exploring a large parameter space using N -body simulations, although possible, may be challenging due to the large amount of computational resources required, therefore an alternative technique may be useful. In the field of numerical simulations, testing results using different techniques always contributes to robustness. Additionally, a continuum medium approach to dynamical problems is always interesting due to shot-noise handling, which is critical when the main mechanism to capture is low-amplitude over-densities, as presented in the previous discussion.

In this work, we propose to use a flexible alternative/complementary strategy based on a continuous medium description for matter density. Our code is based on recent Collisionless Boltzmann Equation (CBE) solver implementations aimed to study cosmic neutrinos (Banerjee & Dalal 2016, hereafter BD16). In cosmological simulations with massive neutrinos, the shot-noise density fluctuations are comparable to the model's actual initial inhomogeneities, triggering artificial

power. Traditionally this is alleviated with a large number of particles, but a continuous medium description like BD16 is an interesting choice. The BD16 method solves the Boltzmann moment equations on a grid and uses tracer particles to estimate the higher-order moments of the Boltzmann hierarchy (instead of the density). We illustrate the code performance by tracking isolated halo and satellite sinking simulations.

In section 2, we describe general aspects of the BPM code. Section 3 presents the simulations following the wake, the global density, and the kinematics response in sinking satellite experiments exploring the dependence of different host halo density profiles. Results are presented in section 4, including comparisons to N -body codes (subsection 4.2). Finally, discussion and conclusions are included in section 5.

2. BPM code

Solving directly CBE+Poisson equations allows one to address many important aspects of the dynamics of self-gravitating systems by keeping the advantage of a continuous media description while avoiding particle shot-noise. However, such an approach is quite expensive due to the high dimensionality of the phase-space but possible on the world's top supercomputers, as, for instance, illustrated by recent works of Yoshikawa et al. (2013, 2020). To reduce the cost compared to the direct approach, we propose instead, following BD16, to use a moment hierarchy approach. Such a technique allows a three-dimensional description that is more manageable from the computational point of view because it reduces dimensionality compared with a full Vlasov one. We calculate the first two CBE moments or the Continuity and Euler-like equations coupled with the Poisson equation on a fixed 3D Cartesian grid. The order of the moment expansion may be questioned, in particular, crossed terms are not included. However, in contrast to Self-consistent field codes (e.g., Weinberg 1989), our approximation is not directly performed on density or gravitational potential. As a healthy measure, we present in section 4.2 a quantitative comparison to full N -body codes like a Fast Multipole Method (FMM, Greengard & Rokhlin 1987; Dehnen 2000) and a Particle Mesh (PM, Hockney & Eastwood 1988) ones.

The set of equations that we solve are listed below:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v^i)}{\partial x^i} = 0 \quad (1)$$

$$\frac{\partial(\rho v^i)}{\partial t} + \frac{\partial(\rho v^i v^j)}{\partial x^j} = -\rho \frac{\partial \Psi}{\partial x^i} - \frac{\partial(\rho \sigma^{ij})}{\partial x^j} \quad (2)$$

$$\nabla^2 \Psi = 4\pi G \rho \quad (3)$$

They are equivalent to the cosmological case presented by BD16. A critical point when using a moment expansion approach is the closure condition to truncate the expansion. Following BD16, we close the CBE hierarchy by using test particles to estimate moments of the distribution function, such as the velocity dispersion tensor and the bulk velocity. We use, at all times, the continuous density from the CBE moments solution to solve the Poisson equation (Eq. 3). Additionally, we evolved in time the test particles using the leap-frog integrator (Kick-Drift-Kick, KDK) and the gravitational potential calculated from the Poisson equation. Then, we use the trajectories of the test particles to calculate the velocity dispersion tensor and the bulk velocity to close the CBE hierarchy. It is important to say that such kinematics estimation is the principal noise source; all the other quantities are computed at the continuous medium regime.

Table 1: Mass and scale radius of the Halo Host Models.

Host	Mass ($10^{12} M_{\odot}$)	R_s (kpc)
Plummer	1.00	30
NFW1	0.98	52
NFW2	0.80	30

Notes. The quoted mass is the one inside 100 kpc. To generate the NFW models, we included a *sech* (hyperbolic secant) weighting function that produced a smoother cut-off at large radii, r_{trunc} , as in Dehnen & McLaughlin (2005) models.

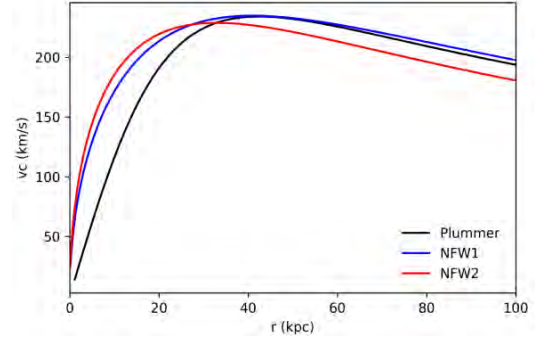


Fig. 1: Dark halos' circular velocity curve. NFW1 (blue) has a similar mass as the Plummer (black) model and also the same circular velocity out of 30 kpc, and NFW2 (red) is 20% less massive but more concentrated and with a similar V_{max} .

To solve the coupled moment equations (Eqs. 1 and 2), following the BD16 implementation, we first use the directional splitting technique (Toro 2012). We then use a 1D piecewise linear advection solver, which is non-linear in space. In addition, we use the Superbee flux limiter (Harten 1983), which has minimum diffusivity. The CBE moments are evolved in time by a Total Variation Diminishing (TVD) Runge-Kutta of order two (RK2) integrator, which is known to preserve the TVD properties (Trac & Pen 2003). Therefore, our code (to which we will refer as BPM) is stable to artificial oscillations since it is TVD. Based on the previous discussion, the number of operations in BPM is similar to that of the PM code, exchanging the CIC for the advection part, which makes BPM more expensive than a PM.

Our implementation adds the possibility of following two separated species, either both of the species followed by the CBE moments (dubbed as CBE+CBE) or one species followed by CBE and the other by PM¹ (hereafter CBE+PM). In the CBE+PM case, the species are coupled only through the Poisson equation. This ability may be useful to follow a stellar component in addition to the DM density; however, in this paper, we consider only the DM component. The CBE+PM version is also useful because it allows savings in computational cost compared to the CBE+CBE one.

3. Simulations

To study the response of the underlying DM halo to a satellite accretion, we used spherical halos without any loss of generality. Since the typical particle speed is small compared with the speed of light, we only solve the continuity-like equation (1st

¹ Instead of the PM code, another N -body code may be used

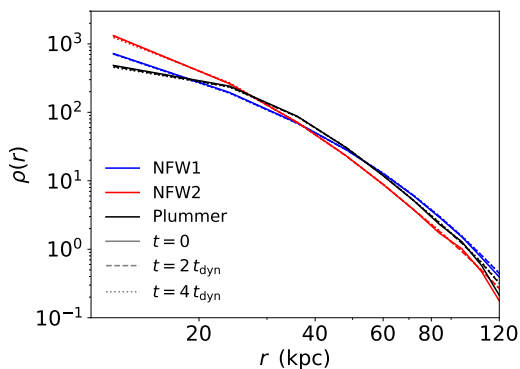


Fig. 2: Stability test for BPM. We present the density profile for each of the three halo models at different times up to $4.0 t_{\text{dyn}}$. The density profile inside 100 kpc barely changes, thus confirming that the BPM accurately calculates its evolution.

CBE moment). BD16 has been illustrated this in their isolated Plummer halo case and the cosmological CDM experiments. For analysis reasons, it is convenient to simulate the host halo and satellite as different species. As mentioned previously, BPM has two versions for simulations with two species: CBE+CBE and CBE+PM. We performed tests using both versions of the code; in particular, for the CBE+PM version, we evolved the host halo using CBE and the satellite using PM. We found that both CBE+CBE and CBE+PM gave similar results regarding the satellite sinking history; however, the CBE+PM is less computationally expensive. For this reason, the results that we will present adopted the CBE+PM version of the code.

3.1. Models and initial conditions

The host halo configuration is either a Plummer (Plummer 1911) density profile or a cuspy NFW (Navarro et al. 1997) model, with spherical symmetry in both cases. For the satellite, we consider only a Plummer density profile. The satellite and the halo have the same mass resolution. The initial conditions were generated by sampling the Plummer distribution function and the NFW model using the publicly available MakeHalo code (McMillan & Dehnen 2007). The mass ratio satellite/Host is 10%. We placed the satellite at a distance of 90 kpc with an orbital circularity of 0.8. Table 1 presents the parameters of the models. We generated the initial conditions for the BPM code directly using the analytical density solution from the equations in McMillan & Dehnen (2007). It is important to state that we did not pretend to simulate a specific galaxy model. We used three distinct halos with V_{max} close to 220 km/s but with different density profiles; a Plummer model, an NFW with a similar circular velocity out of 30 kpc (NFW1), and a 20% less massive NFW halo with a higher concentration but with the similar maximum circular velocity (NFW2), see figure 1. The satellite is represented by a Plummer model with a core radius of 3.6 kpc. Before starting the merger simulation, we evolved all the halos in isolation for 2.0 dynamical times, t_{dyn} .

3.2. Numerical experiments

We performed all calculations on a grid of 512^3 cells and, unless explicitly stated, 256^3 test particles. The time step was selected according to particles velocity dispersion or sound speed for con-

tinuous medium and stability analysis (BD16). We adopted 1 kpc as the spatial resolution for the BPM and PM codes, and 0.3 kpc for the Tree code. The first set of tests followed the stability of the isolated halo for each of the halo models with the BPM code. Figure 2 shows that BPM held the density profile of all the three halos for $4.0 t_{\text{dyn}}$. Next, we performed the sinking satellite experiments. As expected, the satellite sinks due to dynamical friction. As is well known, the cored Plummer profile shows the satellite core stalling (see Petts et al. 2015, and references therein), and the NFW1 and NFW2 models develop a core formation at the end of the run. More details of such a process will be presented in the following sections.

4. Results

4.1. Results discussion

It is critical to accurately find the host and satellite centers since a defect in such estimation would trigger a false density response. We estimate such positions using the shrinking-sphere method (Power et al. 2003). The case for the Plummer halo was challenging. We tested our centroid estimation against the Rockstar halo finder (Behroozi et al. 2013), and the results of both estimations are compatible. Once the initial and the current snapshot share the same centroid, we proceed to calculate the over-density maps. The BPM density is a direct code solution. The results are presented in figure 3, where we show a slice of the over-density maps in the orbital plane with one cell thickness (1 kpc). From top to bottom: Plummer, NFW1, and NFW2 over-density maps are presented for (left to right) $t = 0.3, 1.0,$ and $2.0 t_{\text{dyn}}$. At $t = 0.3 t_{\text{dyn}}$, the local wake behind the satellite is clearly seen in all models. It is also evident an over/under-density dipole (blue and green, respectively). The amplitude of such a dipole is larger for NFW2, followed by NFW1, and finally Plummer. For $t = 1.0 t_{\text{dyn}}$, the wake is still visible for all three models, but the relative amplitude regarding the global dipole is getting even for the NFW models. At $t = 2.0 t_{\text{dyn}}$, it is hard to see the wake, but the global mode is still clearly detected. Figure 4 shows a zoom in the over-density map inside the Plummer core radius at $t = 2.0 t_{\text{dyn}}$; an over/under-density around the satellite is observed, showing that there are modes exchanging angular momentum besides the large scale one. A similar zoom in the NFW models did not show the same structure.

To further study the response of the host halo to the sinking satellite, we calculated the Angular Power Spectrum (APS) of each halo. To that end, we first projected the halo’s density map (previously re-centered on the initial halo centroid) onto a sphere of 100 kpc radius. We then calculated the spherical harmonics coefficients (using the python package HEALPY, Górski et al. 2005; Zonca et al. 2019) $c_{\ell,m}$ of the projected density and summed over all m values, for a given ℓ : $C_{\ell} := \sum_m |c_{\ell,m}|^2$. We will refer to C_{ℓ} as a “mode”. Figure 5 shows the APS, normalized to the initial one, for each host halo and for the three different times. At $t = 0.3 t_{\text{dyn}}$ (solid lines), the cuspy models are similar, however, the compact NFW2 one (red) seems less responsive, and the modes at $\ell = 3^2$ and $\ell = 5$ are similar in amplitude. In contrast, for the NFW1 halo (blue), the $\ell = 3$ mode is dominant over the others. The Plummer model (black) seems

² The analysis method used in our experiments projects the grid cells onto a spherical surface at 100 kpc. The projections of the coordinate planes redistribute the power, shifting the main mode to $\ell = 3$; however, if we instead project only particles, the dominant mode is $\ell = 1$ in agreement with previous studies like Tamfal et al. (2021).

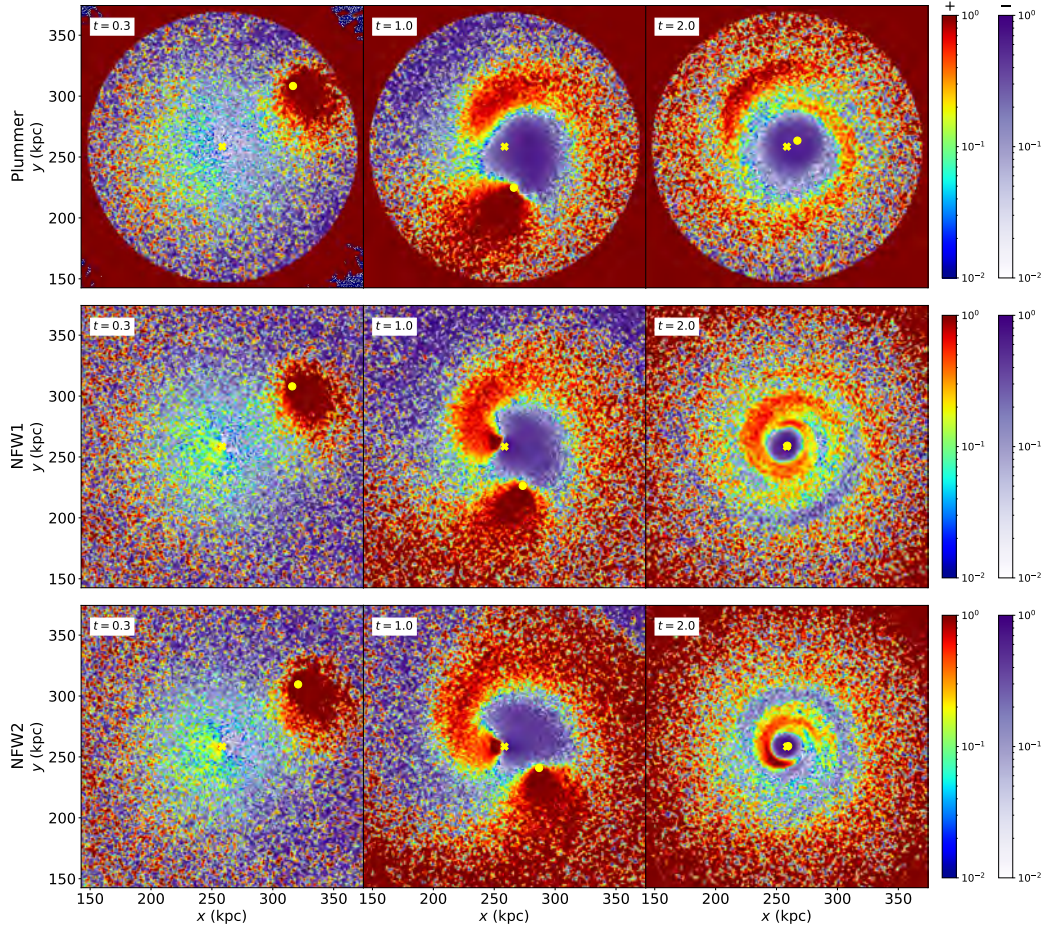


Fig. 3: Evolution of halo over-density ($\rho_i/\rho_0 - 1$) maps for the three host halos shown in figure 1. Purple colors are for negative over-density values and red for positive. The upper row corresponds to the Plummer halo model, the middle row to NFW1 (same V_c as the Plummer model out of 30 kpc), and the bottom row to the smaller NFW2 mass model with higher concentration. Each column corresponds to 0.3 (left), 1.0 (center), and 2.0 (right) dynamical times (t_{dyn}), respectively. The host center and the satellite positions are marked by a yellow cross and solid circle, respectively. The wake and the global response are evident in all panels. The Plummer cored model starts to deviate significantly once the core radius is reached by the satellite. There are small quantitative differences in the wake and dipole for each model. At $2.0 t_{\text{dyn}}$, the global anisotropy is smaller, but other modes are visible.

the most responsive of the three halos based on the APS amplitude. The differences are more dramatic for the following times. At $t = 1.0$ (dashed lines) and $2.0 t_{\text{dyn}}$ (dotted lines), the APS amplitude of NFW2 grows considerably less than the other models, and the shape becomes smoother, suggesting that the compact and smaller NFW2 halo is less responsive than the NFW1 and the Plummer halos. The latter seems the most responsive because its APS amplitude considerably grows. The APS behavior is in agreement with the over-density maps. There is a peak at $\ell = 3$ and $\ell = 5$ for all models, and particularly the Plummer one additionally captures peaks at $\ell = 7$, $\ell = 9$, and possibly more. At the end of the simulation ($t = 2.0 t_{\text{dyn}}$), both the NFW1 and the Plummer APS are heavily dominated by the $\ell = 3$ mode, while the NFW2 one is less bumpy.

It is also interesting to see the evolution of the bulk velocity map in the orbital plane. Figure 6 shows the maps of the X (upper figure) and Y (lower figure) bulk velocity components for the Plummer, NFW1, and NFW2 halos (top to bottom, respectively), and from $t = 0.3, 1.0$, and $2.0 t_{\text{dyn}}$ (left to right, respectively). The left and central column panels show a nega-

tive/positive (blue/red) velocity gradient centered on the satellite illustrating the local wake formation. There is also a global velocity gradient (blue/red). At the rightmost panels, in agreement with the over-density map, there is no clear signal of the local wake. We may ask if the velocity gradient indicates rotation. In figure 7, we show a zoom into the core region of the bulk velocity components maps (at $t = 2.0 t_{\text{dyn}}$), now including the Z component. It is evident that the rotation and all angular momentum transfer occur inside the orbital plane. All density and velocity maps show dependence on the halo velocity profile. Although the results are encouraging, we need to test the robustness of numerical code dependence.

4.2. Comparison with N-body codes

Besides the convenient properties of BPM handling the shot-noise effects, it is also important to test the reliability of BPM in comparison with standard N -body codes. For this reason, we additionally run some of our numerical experiments with our own PM code and also with a Tree code (GyrFalcon, Dehnen

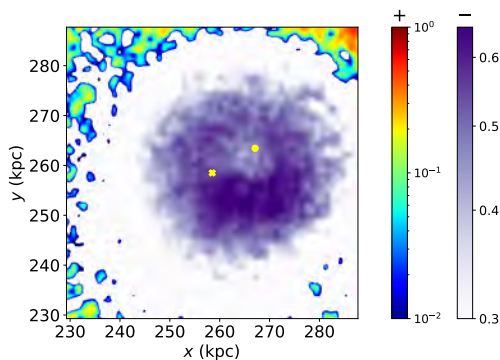


Fig. 4: Modes in over/under-density zoom map inside the core radius for the Plummer model. We can observe a small over-density leading the satellite (cross) and also a trailing under-density.

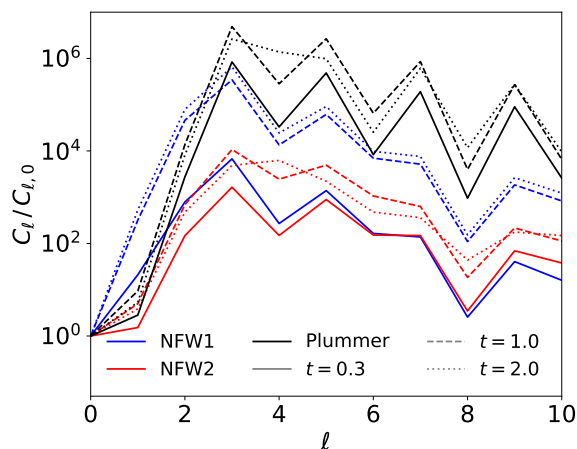


Fig. 5: Evolution of halo density angular modes for the three halo models calculated with the BPM code. The peak at $\ell = 3$ (see footnote 2 for consistency with previous works) has the largest amplitude for Plummer (black) and NFW1 (blue) models at all times. However, NFW2 (red) model shows an increasingly flatter spectrum. This is consistent with maps shown in figure 3, where the dipole has a comparable amplitude with the wake.

2000). Figure 8 presents the sinking satellite history for the three halo models and the three codes. In general, the three codes capture the same sinking satellite history with differences of around 10%. After $1.5 t_{\text{dyn}}$, the sinking stalls for the Plummer model, and with all codes as it has been reported in previous studies (Petts et al. 2015). For the NFW models, the sinking rate also decreases; however, it is close to the resolution.

In addition to testing the consistency of the sinking satellite history across codes, we also tracked the momentum evolution in experiments with the different codes. Figure 9 shows the linear (upper panel) and angular (lower panel) momentum during the $2.0 t_{\text{dyn}}$. Indeed, the BPM (solid line) case has lower momentum conservation, likely because it only solves the continuity equation. The rate of change of momentum conservation is significant for BPM but does not grow systematically over time. Hence, the main dynamical effects are consistent with traditional full N -body codes, at least for the number of simulated dynamical times, as we can see in figure 8.

Finally, we may ask if BPM truly has an advantage in tracing the over-density signal in the simulations. Figure 10 shows the over-density maps (from top to bottom) built based on: BPM, PM, and Tree codes (with two different Cloud-In-Cell, CIC, resolutions, 1.0 and 0.3 kpc, respectively). For the N -body simulations, we used the CIC interpolation to build the density and over-density maps, while, for BPM, we used the numerical solution of the continuity+Poisson equations. Density, in BPM code, inherits some noise from the velocity estimation; however, it is less noisy than the N -body counterparts at a similar grid resolution. The fourth row shows the analysis of the Tree simulation using a CIC grid with equal spatial resolution as the simulation (i.e., 0.3 kpc, $N_g=1720$). In this case, the satellite or the global mode is hardly seen. However, the structures start to appear when a lower resolution (1.0 kpc) CIC grid is used (third row). The above outlines that the analysis of traditional N -body simulations requires special treatment, for example, looking for convergence when smoothing the grid or using adaptive smoothing procedures. The PM case is similar to the Tree smoothed case. Finally, the BPM case directly shows the density/over-density field out of the continuity equation with no smoothing required. All three codes show similar features (see first three rows); however, BPM makes it easy to identify small changes in the density field without a special analysis of the simulations.

Figure 11 shows the evolution of the APS estimated in simulations of the NFW1 halo using three different codes: BPM, PM, and Tree (solid, dashed, and dotted lines, respectively) with spatial resolution ~ 1.0 kpc in the density field. The behavior of the APS is similar between codes: the $\ell = 3$ mode has the largest amplitude, while the other odd modes decrease in amplitude as ℓ increases. This behavior remains similar for three snapshots. The BPM and PM codes capture the dominant $\ell = 3$ mode with the same amplitude, and the Tree code captures the mode with a lower one. For the $\ell = 5$ mode, the codes differ in amplitude, from highest to lowest, BPM, PM, and Tree, respectively. The relative amplitude between codes of the $\ell = 7$ mode varies in time. For example, at $t = 0.3$, BPM has a higher amplitude, but PM and Tree have a lower similar one, while at $t = 2.0$, BPM and Tree have equal amplitude, and PM has a lower one.

We may wonder if the PM code already captures the response modes. For that reason, we performed a test with BPM and PM codes using $N_p = 128^3$ particles and a grid of $N_g^3 = 512^3$ cells (1 kpc resolution). Figure 12 presents the results for the Plummer halo at $t = 1.0 t_{\text{dyn}}$. The top and middle panels show the over-density field (built similarly to figure 10) for the BPM and PM simulations, respectively. The bottom panel shows the corresponding normalized APS (green lines). We clearly see a significant amount of damping in the PM case (green dashed line); even the relative satellite-halo-center distance is shorter at the scale of 15%, suggesting that unresolved modes in the PM code slow down the satellite sinking in the BPM. Increasing the particle number to 256^3 is not enough (magenta line in figure 12), and possibly 512^3 may approach BPM; however, the memory cost increment would be a factor of 64 larger for PM in comparison with BPM, considering that the memory cost of both codes scales like $N_p + N_g^3$.

Regarding execution time, we made a comparison using a single core. Table 2 summarizes the setup of the simulations and their performance. It is relevant to state that we did not make an exhaustive comparison of code performance. We found that the PM code is $\sim 40\%$ faster than the BPM one (for a similar simulation setup). Tree and BPM run at similar times, both codes using the same number of particles. However, BPM easily captures subtle fluctuations in the density field (see figure 10).

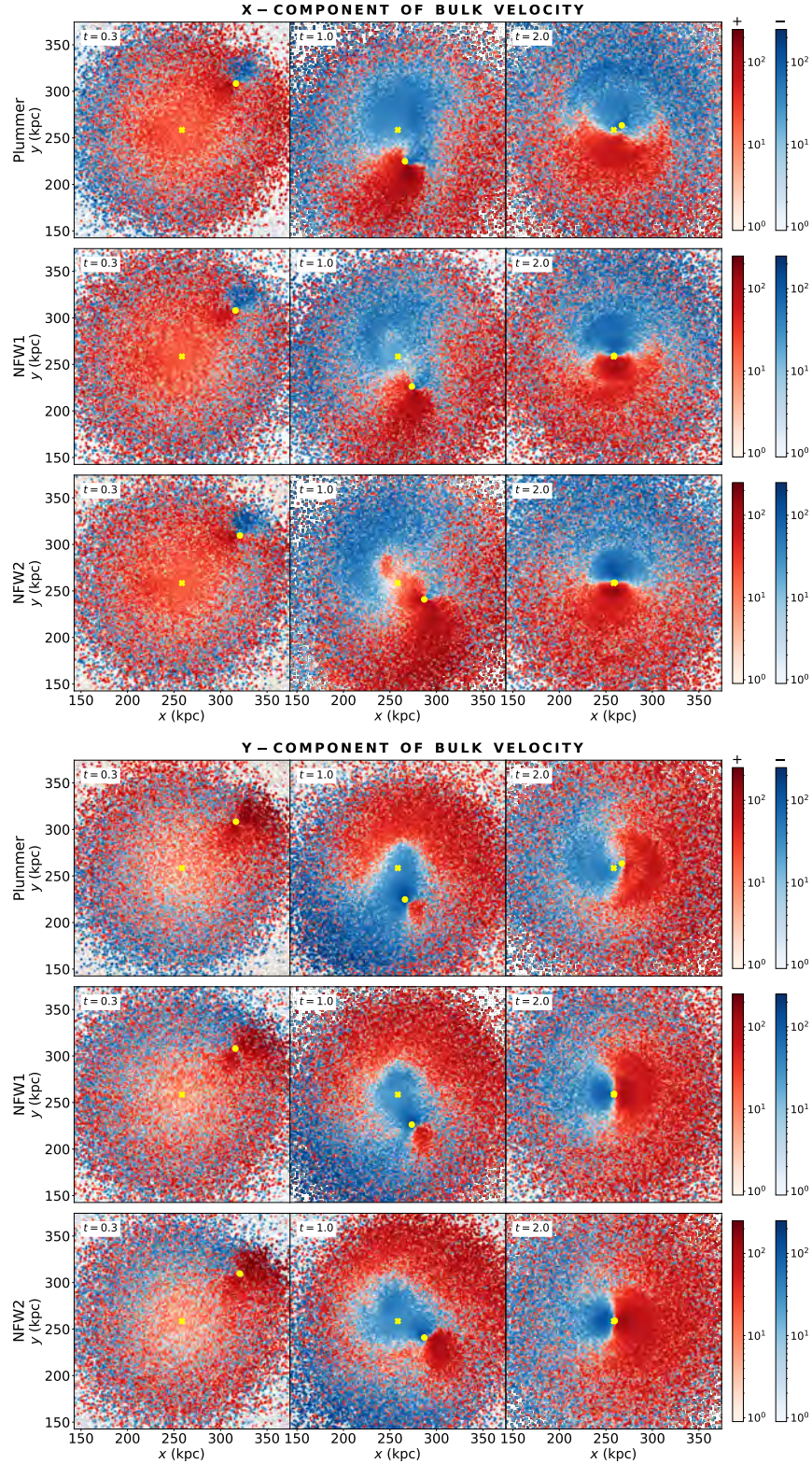


Fig. 6: Velocity maps. The map shows the X (upper) and Y (lower) bulk velocity components (positive red, negative blue). As in figure 3, the columns correspond to, from left to right, $t = 0.3$, 1.0 , and $2.0 t_{\text{dyn}}$, respectively; and the rows to, from top to bottom, the Plummer, NFW1, and NFW2 halos, respectively. Satellite and Host centers are pointed by a yellow circle and a cross, respectively. The halo response behind the satellite is clearly seen in the first two columns. The map for NFW2 starts to deviate from the other two models at $t = 1.0$, suggesting that the momentum redistribution is different. The differences between the middle and right panels, for all the models, suggest that angular momentum transfers from the satellite to the background halo.

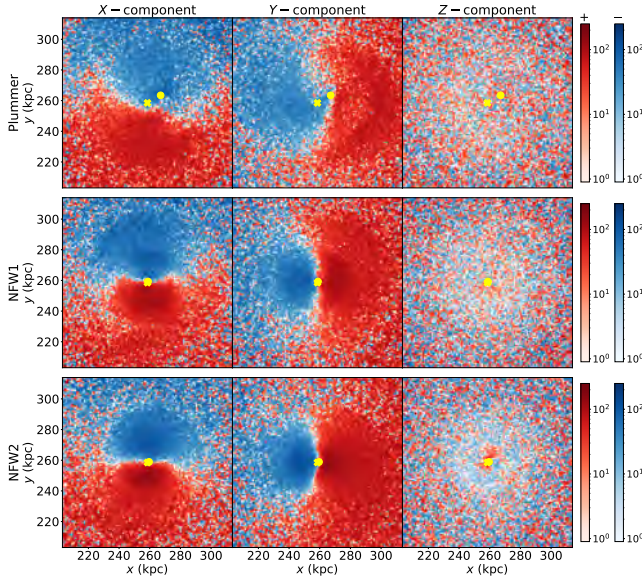


Fig. 7: Central kinematics at the end of the simulation for the three simulation models. Each row corresponds to a halo model, from top to bottom: Plummer, NFW1, and NFW2. The columns from left to right show the X, Y, and Z bulk velocity components, respectively. The first evident feature is that all the angular momentum exchange happens in the orbital plane (X–Y); therefore, the Z velocity component shows no pattern. Almost cylindrical rotation is evident in the simulations’ final stage.

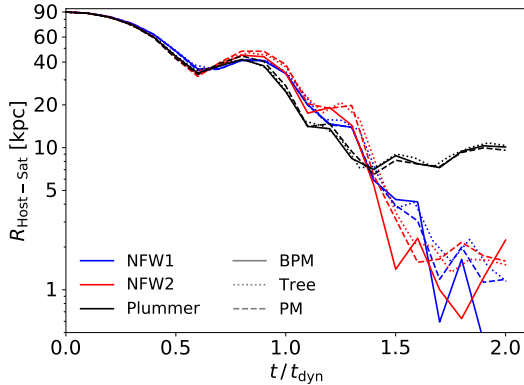


Fig. 8: Sinking satellite history for different models and codes. Solid, dotted, and dashed lines correspond to BPM, Tree, and PM codes, respectively. The three codes are consistent with each other up to $t = 1.3 t_{\text{dyn}}$ when small differences start to appear. The blue, red, and black colors correspond to NFW1, NFW2, and Plummer models simulations, respectively. The core stalling of the sinking satellite is evident in the Plummer model simulation.

Furthermore, from figures 11 and 12, the amplitude difference of the mode of order $\ell = 3$ between BPM and PM is lower for the simulations with 256^3 particles (figure 11) than the case with 128^3 , suggesting that there is a shot-noise effect. If we increase the number of particles to reduce such an effect in the N -body simulations, then the memory cost and, possibly, the execution time would be higher than the BPM simulation (with 256^3 particles).

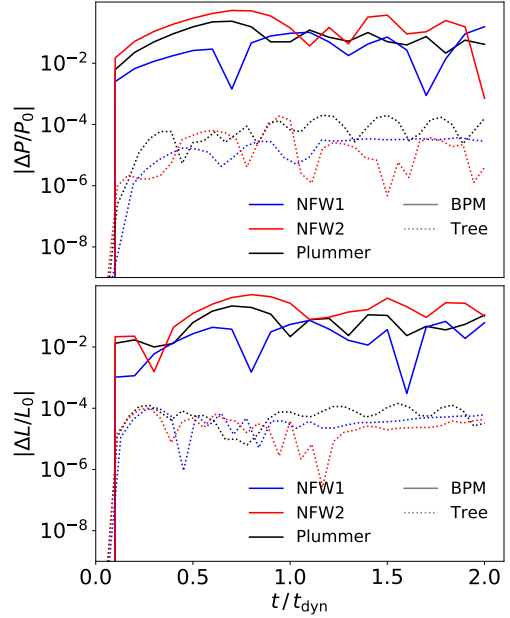


Fig. 9: Momentum conservation. The upper panel shows the linear momentum evolution for the three halos with two different codes, BPM (solid lines) and full N -body Tree code (dotted lines). The lower panel shows the angular momentum conservation for the three models with the corresponding codes.

5. Discussion and conclusions

The study of triggered modes in N -body simulations of sinking satellites, either to assess their signature in the stellar halo or to study their role in dynamical friction, has been a challenging task requiring large particle numbers and computational resources. Using a new hybrid non-linear CBE+Poisson solver, we successfully detected the over-density and kinematics response in sinking satellite experiments. We explored the effect on the host halo response related to its density profile. We also discussed the properties of the modes during the core stalling phase.

The upper panel in figure 13 shows the evolution of the density profile for the host+satellite system assuming the three host halo models (initial correspond to dashed lines and final to solid ones). It clearly shows the formation of a core in the central NFW1 (blue) that turns close to the Plummer (black) density profile. The NFW2 model (red) develops a smaller core. The behavior is not surprising due to the mass ratio between the satellite and the enclosed mass (Goerdt et al. 2010). The new code, BPM, that we present in this paper correctly captures such a process.

The evolution described above is consistent with the stalled sinking shown in figure 8. After $1.5 t_{\text{dyn}}$, it is possible to observe that the satellite falling holds. Such core stalling phase has been discussed by several studies, like Petts et al. (2015) and references therein. There is no consensus on the explanation. In our case, the stalling is observed at around 10 kpc for Plummer and around 2-1 kpc for NFW2 and NFW1.

Figure 4 shows the over-density modes within the Plummer core. A similar figure in the NFW models shows no structure, possibly because the core radius is smaller and hardly resolved. We may speculate that the central modes in the Plummer core compete with the global mode to weaken dynamical friction. The low amplitude of the density modes in figure 4 and the high

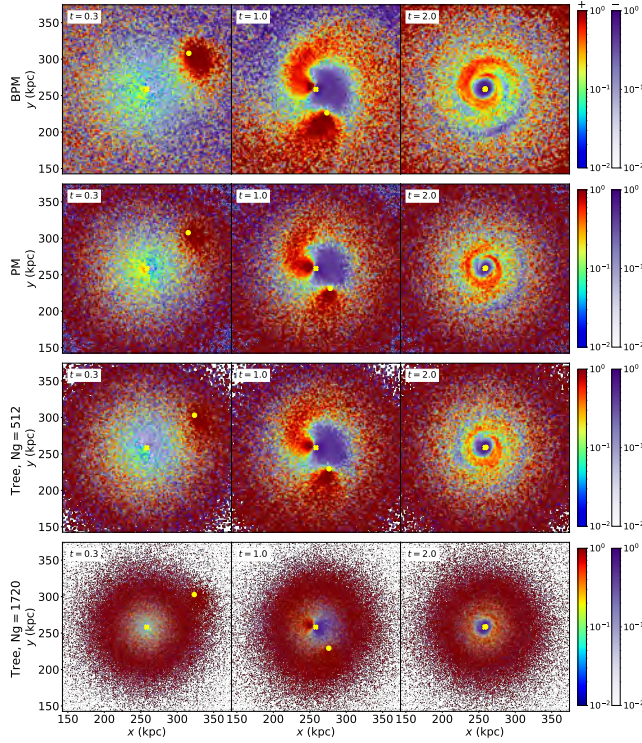


Fig. 10: Over-density maps noise effects with different codes and smoothing. The top row corresponds to BPM, where the over-density is calculated directly with the solution of Continuity+Poisson equations. The second row shows the PM over-density after applying CIC to the particle distribution. The two lowest rows correspond to the Tree code after applying CIC, using a grid resolution of 1.0 (512^3 cells) and 0.3 (1720^3 cells) kpc, respectively. The highest resolution hardly shows the wake and global response. However, the smoothed case shows a host halo response close to the PM case, showing dependence on the grid smoothing.

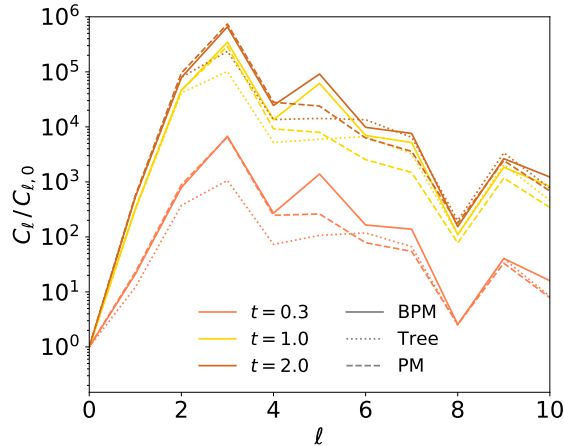


Fig. 11: Evolution of density angular modes for the NFW1 halo with different codes. The peak at $\ell = 3$ has the largest amplitude at any time. It has the same amplitude for BPM (solid line) and PM (dashed line), whereas it has a lower amplitude for the Tree code (dotted line). The amplitude of the peaks at $\ell = 5$ and $\ell = 7$ differs between codes.

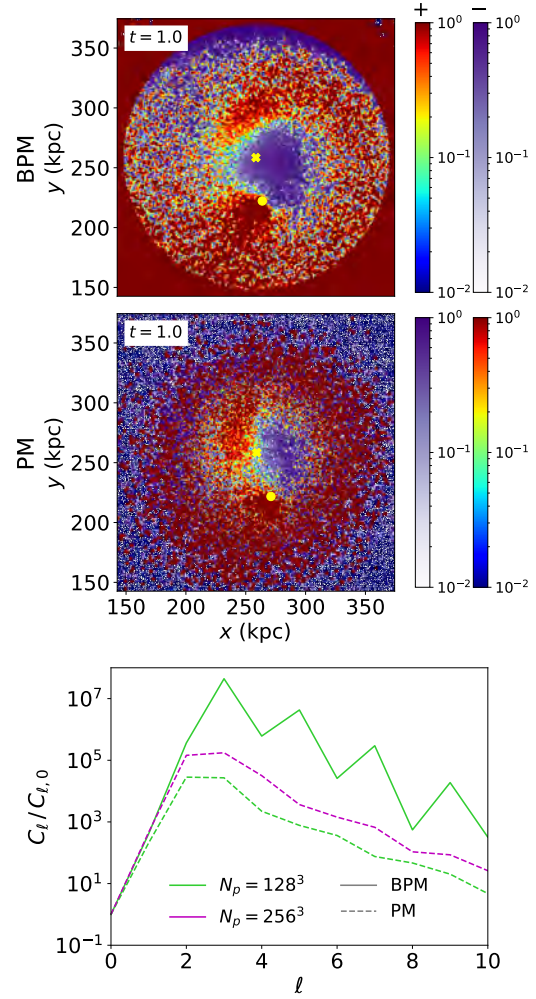


Fig. 12: Noise Damped Modes for the Plummer model. The upper panels show the over-density maps at the orbital plane using BPM (top) and PM (middle). The wake and the overall map are affected by noise in the PM test. The bottom panel shows the APS (green lines), where the $\ell = 3$ peak is severely damped for the PM code (dashed line). Both codes used 512^3 cells, 128^3 particles, and a spatial resolution of 1 kpc. Additionally, we include a PM case using 256^3 particles (magenta dashed line); the APS amplitude is greater than the previous test. The dependence on particle number suggests that there is a shot-noise effect in the PM APS.

V_{rms}/V_c value in figure 13 (see also Petts et al. 2015) may contribute to the weakening of dynamical friction (Inoue 2011), in agreement with recent linear CBE perturbation study (Kaur & Stone 2021, appeared after our submission).

Our results suggest that the response of the host halo is sensitive to its density profile (see figure 3). The smaller and compact NFW2 halo shows the strongest amplitude dipole, which is also observed in the flatter APS (figure 5) compared to the two massive halos, NFW1 and Plummer. All of them show the local wake with a good signal to noise. The evolution of the amplitude and modes of the APS during the simulation (figure 5) indicates that NFW2 is the least responsive halo in contrast to the Plummer

Table 2: Performance of the Simulations

Code	$N_p^{1/3}$	$N_g^{1/3}$	SR (kpc)	ET/ET _{BPM}
BPM	128	512	1.0	1.0
PM	128	512	1.0	0.39
BPM	256	512	1.0	1.0
PM	256	512	1.0	0.40
Tree	256		0.3	1.0

Notes. Summary of the simulations' setup and their performance. From left to right, the columns are code/technique, number of particles, grid size, spatial resolution, and the execution time normalized to the corresponding BPM simulation.

one. This behavior is understandable by looking at the middle panel in figure 13, where we show the initial circular velocity (dash-dotted lines) and 3D rms velocity (solid line with stars). It is clear that the NFW2 model (red) has had a higher velocity dispersion since the beginning, possibly blurring the orbital resonances. The lower panel shows the same (V_c, V_{rms}) curves at the end of the simulation; the behavior of V_{rms} is correlated with the core stalling radius. The final stage where core stalling is present is even more different; however, a common feature is that the local wake is not observed.

We can summarize our conclusions as:

1. Our code, named BPM, is able to capture the response modes (local wake, global dipole) in the density and velocity fields with a comparable computational cost to traditional N -body simulations. Since BPM is based on solving a moment hierarchy of the CBE+Poisson equations on a 3D Eulerian grid and using test particles kinematics to close the Boltzmann hierarchy, BPM is less sensitive to shot-noise than the standard N -body approach. Therefore, BPM facilitates the detection of subtle fluctuations in the density and velocity fields without special treatment (e.g., higher number of particles, post-processing analysis) that would be required to achieve the same level of accuracy in a standard N -body simulation.
2. Both the spatial distributions of over/under-densities and the normalized Angular Power Spectrum depend on the halo density profile, particularly the global dipole strength. The dependence of the over-density map is more evident when the satellite approaches the core region, and it is related to the central V_{rms}/V_c ratio.
3. Most of the angular momentum transfer occurs in the orbital plane. At advanced stages of the simulation, we found a central cylindrical rotation (figure 7). In the case of the presence of a stellar component, this may be a sign of past massive accretion events.
4. During the satellite core stalling stage, there is no clear signature of the local wake; however, low-density contrast modes are detected (figure 4). An interesting follow-up project will be to study their contribution to the slower sinking rate.
5. Based on the leftmost panels of figure 3, we conclude that the recent detection of the galactic stellar halo response (Conroy et al. 2021) may contribute to additional constraints to the mass distribution of the dark matter halo in the Milky Way. However, the situation is degenerated both in mass and density profile.
6. The capability of BPM for avoiding shot-noise makes it a convenient choice for problems that require resolving low-amplitude modes, with a moderate computational cost in memory. Situations like the modes discussed in galaxy dynamical problems or low surface brightness structures in

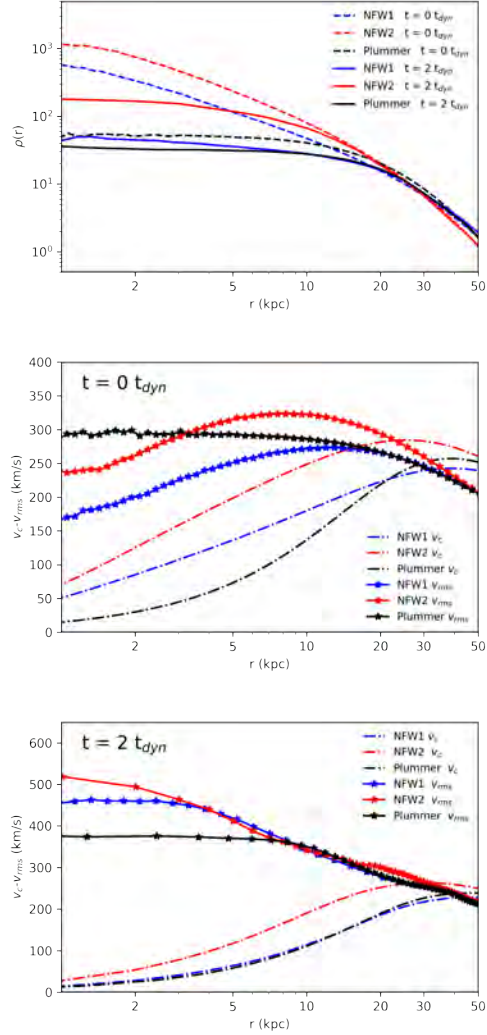


Fig. 13: Core formation. The upper panel shows the initial (dashed) and the final (solid) density profiles for the three halo models. Both NFW models clearly show that a core was developed at 20 (NFW1) and 10 (NFW2) kpc, respectively. The middle panel shows the initial circular velocity (dash-dotted line) and the initial 3D velocity dispersion (solid line with stars). The lowest panel shows the same kinematic profiles at the end of the simulation.

galaxies are natural choices. BPM also allows to study the response modes even in the case of multiple concurrent perturbations.

7. The right-hand-side term (source) of the Euler-like equation might consider additional dark matter physics such as self-interaction or other effects (equation of state) that produce a possible change in the halo density response (sound speed). An exploration of such an interesting avenue in density and kinematics will be presented in future studies.

Acknowledgements. GA acknowledges useful exchange with A. Banerjee. We thank S. Roca-Fàbrega and H. Velázquez for valuable discussions. GA and AT thank support from a CONACyT PhD fellowship. GA, AT, and OV acknowledge support from DGAPA-UNAM grants IN112518, IG101222, and AG101620. The authors thank for the facilities of cluster computers:Atocatl/LAMOD-UNAM

and Miztli/DGTIC-UNAM. LAMOD is a collaborative project of IA, ICN, and IQ Institutes at UNAM.

References

- Aquino-Ortíz, E., Valenzuela, O., Sánchez, S. F., et al. 2018, *MNRAS*, 479, 2133
- Banerjee, A. & Dalal, N. 2016, *J. Cosmology Astropart. Phys.*, 2016, 015
- Behroozi, P. S., Wechsler, R. H., & Wu, H.-Y. 2013, *ApJ*, 762, 109
- Cautun, M., Benítez-Llambay, A., Deason, A. J., et al. 2020, *MNRAS*, 494, 4291
- Conroy, C., Naidu, R. P., Garavito-Camargo, N., et al. 2021, *Nature*, 592, 534
- Cooper, A. P. 2021, in *American Astronomical Society Meeting Abstracts*, Vol. 53, *American Astronomical Society Meeting Abstracts*, 303.06
- Dalton, G., Trager, S. C., Abrams, D. C., et al. 2012, in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, Vol. 8446, *Ground-based and Airborne Instrumentation for Astronomy IV*, 84460P
- Dehnen, W. 2000, *ApJ*, 536, L39
- Dehnen, W. & McLaughlin, D. E. 2005, *MNRAS*, 363, 1057
- Fernández-Trincado, J. G., Beers, T. C., & Minniti, D. 2020, *A&A*, 644, A83
- Gaia Collaboration, Brown, A. G. A., Vallenari, A., et al. 2021, *A&A*, 649, A1
- Garavito-Camargo, N., Besla, G., Laporte, C. F. P., et al. 2021, *ApJ*, 919, 109
- Goerdt, T., Moore, B., Read, J. I., & Stadel, J. 2010, *ApJ*, 725, 1707
- Górski, K. M., Hivon, E., Banday, A. J., et al. 2005, *ApJ*, 622, 759
- Greengard, L. & Rokhlin, V. 1987, *Journal of Computational Physics*, 73, 325
- Harten, A. 1983, *Journal of Computational Physics*, 49, 357
- Hockney, R. W. & Eastwood, J. W. 1988, *Computer simulation using particles*
- Inoue, S. 2011, *MNRAS*, 416, 1181
- Kaur, K. & Stone, N. C. 2021, arXiv e-prints, arXiv:2112.10801
- Majewski, S. R., Schiavon, R. P., Frinchaboy, P. M., et al. 2017, *AJ*, 154, 94
- McMillan, P. J. & Dehnen, W. 2007, *MNRAS*, 378, 541
- Navarro, J. F., Frenk, C. S., & White, S. D. M. 1997, *ApJ*, 490, 493
- Ogiya, G. & Burkert, A. 2016, *MNRAS*, 457, 2164
- Petts, J. A., Gualandris, A., & Read, J. I. 2015, *MNRAS*, 454, 3778
- Planck Collaboration, Aghanim, N., Akrami, Y., et al. 2020, *A&A*, 641, A6
- Plummer, H. C. 1911, *MNRAS*, 71, 460
- Power, C., Navarro, J. F., Jenkins, A., et al. 2003, *MNRAS*, 338, 14
- Rich, R. M. 2018, in *Rediscovering Our Galaxy*, Vol. 334, 233–241
- Rodríguez-Puebla, A., Behroozi, P., Primack, J., et al. 2016, *MNRAS*, 462, 893
- Tamfal, T., Mayer, L., Quinn, T. R., et al. 2021, *ApJ*, 916, 55
- Toro, E. F. 2012, *The Riemann problem in computational science*, 87–103
- Trac, H. & Pen, U.-L. 2003, *PASP*, 115, 303
- Wang, W., Han, J., Cole, S., et al. 2018, *MNRAS*, 476, 5669
- Weinberg, M. D. 1989, *MNRAS*, 239, 549
- Yoshikawa, K., Tanaka, S., Yoshida, N., & Saito, S. 2020, *ApJ*, 904, 159
- Yoshikawa, K., Yoshida, N., & Umemura, M. 2013, *ApJ*, 762, 116
- Zonca, A., Singer, L., Lenz, D., et al. 2019, *The Journal of Open Source Software*, 4, 1298

Capítulo 8

Simulando la estructura en modelos de materia oscura con supresión en el espectro de potencias.

La identidad o existencia de la materia oscura es uno de los problemas más apremiantes de la cosmología contemporánea. A finales de los 90's se descubrió que propiedades observables de las galaxias como las curvas de rotación y la abundancia de galaxias satélites o fluctuaciones en imágenes asociadas a eventos de lentes gravitacionales están asociadas con las presumibles propiedades de la materia oscura. Con el fin de utilizar observaciones de catastros en desarrollo como Pan-STARRS [153], LSST [24] o Euclid [25] se requieren predicciones precisas asociadas a los modelos de materia oscura y la posibilidad de barrer un amplio espacio de parámetros. Existe una gran discusión respecto a la confiabilidad de tales predicciones debido al efecto del shot-noise, no existiendo actualmente una solución completa. En este capítulo proponemos utilizar el código Boltzmann-PM o BPM para realizar predicciones acerca de la abundancia de subestructura. Cabe mencionar que el trabajo mostrado en este capítulo es un proyecto aún en desarrollo.

8.1. Free-Streaming y la función de masa de halos, Espectro de Potencias

Los candidatos más naturales para partícula de materia oscura tienen origen térmico, en tales casos su masa está asociada a su temperatura cinética. Tal temperatura cinética determina su dispersión de velocidades y por lo tanto la importancia del llamado flujo libre (“free-streaming” en inglés). Determinar la escala en la que se ha suprimido la estructura cósmica o de flujo libre es equivalente a acotar la temperatura cinética de la materia oscura o su masa de partícula si es de origen térmico. Dicha escala puede ser estimada mediante teoría de perturbaciones lineal [154]:

$$R_{f,DM} \approx 0.31 \left(\frac{\Omega_{DM}}{0.3} \right)^{0.15} \left(\frac{h}{0.65} \right)^{1.3} \left(\frac{\text{keV}}{m_{DM}} \right)^{1.15} h^{-1} \text{Mpc} \quad (8.1)$$

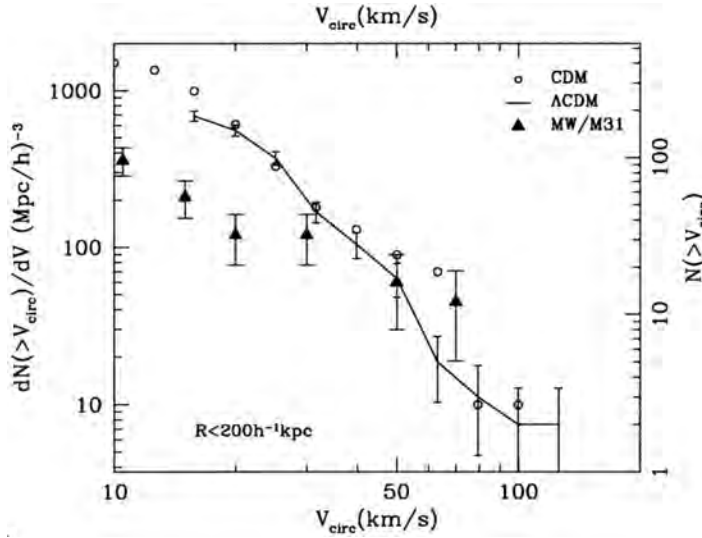


Figura 8.1: Subestructura en modelos LCDM. Función de velocidad circular máxima para simulaciones LCDM/CDM (línea continua/círculos abiertos). Los triángulos negros muestran el promedio para la Vía Láctea y M31. Es evidente la discrepancia. Figura tomada de [156]).

Esta escala de supresión ha sido incluida en modelos que calculan el número de fluctuaciones autogravitantes a un corrimiento al rojo dado como los modelos de Press-Schechter extendidos, por lo que tiene un efecto directo en la abundancia o función de masa de halos para diferentes tipos de materia oscura.

De forma similar al caso de neutrinos, debido a la velocidad de flujo libre las partículas “escapan” de las regiones sobre-densas de tamaño $R_{f,DM}$ o más pequeñas, lo que limita el crecimiento de estructura en estas escalas. Esto da como resultado una supresión o corte en el espectro de potencias de materia en dichas escalas. Esta supresión puede afectar la abundancia de halos de materia oscura por debajo de $R_{f,DM}$ (menos halos de baja masa) y también las propiedades internas de los halos [155] (por ejemplo su concentración).

8.2. El problema de los satélites faltantes o la conexión entre subestructura y la longitud de flujo libre de la materia oscura en Cosmología

A finales de los 90’s las simulaciones cosmológicas de formación de estructura no lograban capturar a los halos dentro de otros halos, los llamados subhalos que potencialmente albergarían a las galaxias satélites. A este problema se le llamó “overmerging” y existía una amplia discusión sobre si esto era un defecto de las simulaciones o una propiedad del modelo. En 1999, el estudio [156] logró capturar, por primera vez, la subestructura interna de los halos, este estudio fue realizado con un código de malla adaptiva ART (del inglés Adaptive Refinement Tree, [76]). El análisis encontró que

si se comparaban las observaciones promediadas para la Vía Láctea y Andrómeda, se encontraba un exceso cercano a dos órdenes de magnitud en el número de galaxias satélites (ver figura 8.1). Poco después otro estudio realizado con el código de árbol PKDGRAV [157] corroboró el resultado. A esta discrepancia se le llamó el “problema de los satélites faltantes” o de “la subestructura para escenarios LCDM” y actualmente es objeto de amplio estudio en la comunidad astronómica. Previamente [158] había hallado un exceso similar utilizando métodos de Press-Schechter semi-analíticos para estimar la abundancia de halos, sin embargo, los parámetros de tal modelo estaban débilmente constreñidos por lo que prefirieron modificarlos pensando que ilustraban una deficiencia de su modelo estadístico. Lo anterior ilustra que tales técnicas tienen poco poder predictivo pero sí un gran potencial de sintetizar resultados una vez calibrados con cálculos completos. Debido a esto el resultado de los satélites faltantes o de la conexión entre la abundancia de galaxias satélites/subestructura y la longitud de flujo libre o temperatura cinética de la partícula de materia oscura, es una predicción directamente debida a simulaciones computacionales del proceso no-lineal de formación de estructura.

8.3. Simulaciones WDM

Posteriormente, el estudio [155], utilizando el código ART [76], consideró un modelo con longitud/masa de flujo libre correspondiente al tamaño/masa de galaxias enanas y demostraron que podría ser una explicación al aparente exceso de galaxias satélites en simulaciones LCDM (ver figura 8.2). Incluso se calibró la masa de un posible neutrino “right-handed” fuera del modelo estándar de partículas para ajustar la abundancia de galaxias satélites observada en el Grupo Local. Esto terminó de establecer la conexión entre la naturaleza de la materia oscura y la abundancia de subestructura en galaxias. Trabajos posteriores profundizaron en el estudio de la estructura interna de los halos [159] o una población entera de halos [154] o se enfocaron en analizar la estructura de los halos por debajo de la longitud de flujo libre y el efecto en las velocidades intrínsecas térmicas de las partículas [160]. El modelo de una partícula con temperatura cinética tibia como candidato a materia oscura (LWDM) se estableció como una alternativa a LCDM.

8.4. Shot-Noise y subestructura espuria de materia oscura

El estudio [161] hizo notar que parte de la subestructura formada en simulaciones cosmológicas de N -cuerpos, es espuria (ver figura 8.3) y se debía a que la amplitud del espectro de potencias físico era comparable a la amplitud del ruido poissónico o shot-noise originado por la representación discreta del campo de densidad usando partículas. Esto implicaba que parte de la estructura en los estudios previos podría ser artificial y por lo tanto la calibración de las propiedades de la materia oscura. El trabajo estimó el número de partículas para amortiguar tal tipo de estructura artificial, sin embargo,

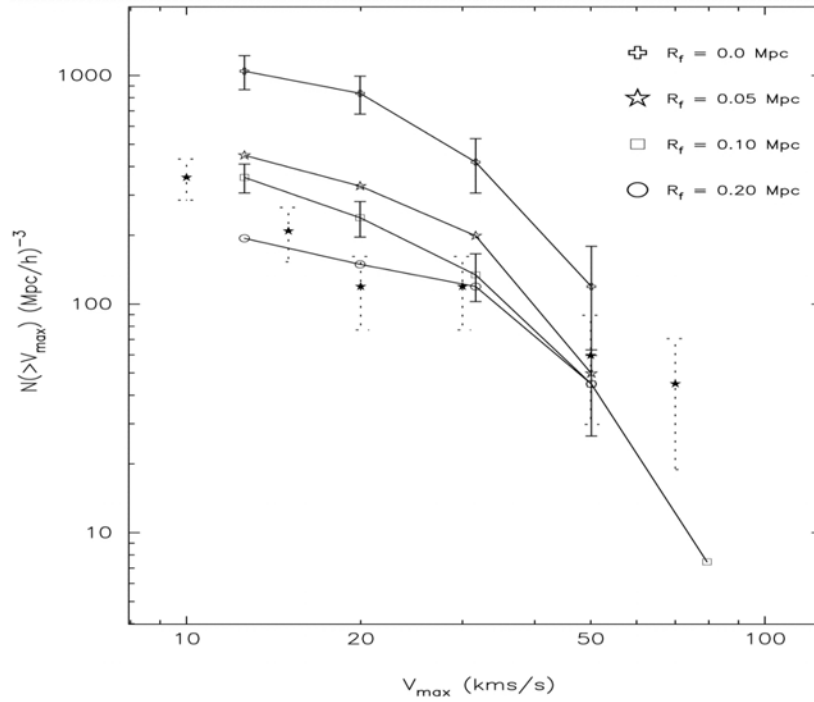


Figura 8.2: Subestructura en modelos LWCDM. Función de velocidad circular máxima para simulaciones LCDM (cruces abiertas) y para modelos LWDM con diferentes escalas de flujo libre. Las estrellas negras están basadas en observaciones de MW/M31. Es evidente que la abundancia de subhalos depende de la escala de flujo libre. Figura tomada de [155].

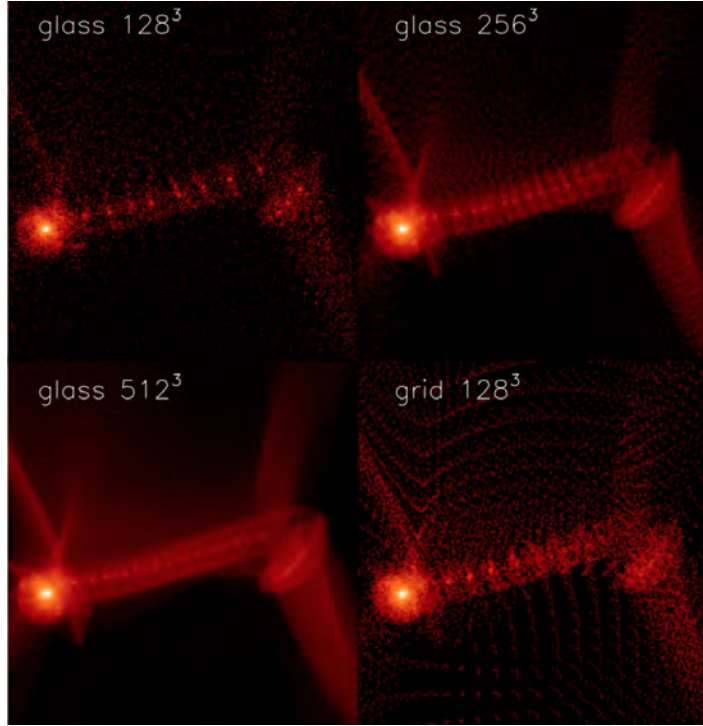


Figura 8.3: Efecto del shot-noise en la formación de estructura para modelos con supresión en el espectro de potencias. Nótese la fragmentación a lo largo de los filamentos incluso para condiciones iniciales aleatorizadas (GLASS) y con alta resolución (512^3). Figura tomada de [161].

esto alcanzaba números fuera del alcance computacional actual, pues la resolución de masa efectiva (m_p) para este tipo de simulaciones mejora sólo como la raíz cúbica del número de partículas empleadas (N), en contraste con las simulaciones CDM que mejora con N . Alternativamente, el trabajo propone un estimado de la escala de los halos originados por shot-noise, la cual podría utilizarse para filtrar tal efecto.

Posteriormente, [162] utilizando el código GADGET-3 realizan simulaciones de halos en una cosmología WDM, alcanzando mil millones de partículas dentro del radio virial. A pesar del gran costo computacional siguen observando contaminación por halos generados por el shot-noise y describen una estrategia para seleccionar y eliminar a los subhalos artificiales. Dicha estrategia explota tres propiedades de las estructuras espurias: masa, dependencia en la resolución y la forma de la distribución inicial de partículas. Sin embargo, resulta en una estrategia laboriosa pues implica rastrear las partículas que constituyen un halo hasta las condiciones iniciales y hacer simulaciones con diferentes resoluciones espaciales, además de hacer distintos cortes al catálogo de halos. Y aún así, a pesar del gran esfuerzo y el gran costo en recursos computacionales, es insatisfactorio y puede aún considerar halos espurios o peor aún eliminar subhalos reales, por lo que futuras comparaciones con observaciones verán afectada su interpretación. [160] discute que el ruido en las velocidades iniciales también contribuye a la potencia espuria en la formación de estructura.

Sim.	Código	N_p [10^9]	m_p [M_\odot]	CH [10^6]	N_{cores}	RAM [TB]	Datos [TB]	Ref.
Aquarius A-1	GADGET-3	4.3	1.7×10^3	3.5	1024	3	45	[165]
GHalo	PKDGRAV2	2.1	1.0×10^3	2	1000	1	60	[166]
Via Lactea II	PKDGRAV2	1.0	4.1×10^3	1.5	3000	0.3	20	[167]

Cuadro 8.1: Características y costo computacional de algunas “simulaciones zoom-in” (volúmenes pequeños, escalas galácticas) pequeños realizadas con códigos en el estado del arte. Datos tomados de [70]. N_p corresponde al número total de partículas, m_p es la resolución en masa, CH (core-hours) corresponde al número de horas por núcleo, N_{cores} indica el número total de núcleos usados, RAM y Datos corresponden a la memoria y espacio en disco, respectivamente, utilizados para las simulaciones.

8.5. Simulaciones de la subestructura en modelos de materia oscura tibia libres de ruido

Como es sugerido por el trabajo [163], un modelo de materia oscura con supresión del espectro de potencias a escalas pequeñas como la materia oscura tibia podría explicar la aparente discrepancia entre la abundancia de galaxias satélites y la subestructura predicha por las teorías cosmológicas. [161] han mostrado que el shot-noise en tales simulaciones puede crecer y producir estructura espuria. Actualmente es una combinación de utilizar un gran número de partículas y de un proceso de filtrado lo que descansa detrás de las predicciones sobre la abundancia de subestructura en modelos WDM [162, 164]. Cabe mencionar que tales simulaciones tienen un costo computacional cercano al millón de horas de CPU o incluso más, como se puede ver en la tabla 8.1, en donde la simulación Aquarius fue utilizada en los trabajos [162, 164].

El código desarrollado en esta tesis, BPM, presenta una alternativa a este tipo de estudios, ya que no describe el campo de densidad utilizando partículas, sino como un medio continuo. La figura 8.4 muestra el campo de densidad para una re-simulación de la caja presentada en [155], aunque en lugar de realizar un zoom sobre un halo, en este caso se simuló el volumen entero con la misma resolución. Es muy claro notar la ausencia de la fragmentación presente en [161] y [162].

8.5.1. Simulaciones WDM con BPM

Aunque la posibilidad de representar al campo de densidad de manera continua con nuestro código BPM es muy atractiva, como ocurre con cualquier técnica numérica, no se está libre de limitantes. Las principales son la resolución fija en nuestra implementación, lo cual podría ocasionar la ausencia de subestructura por debajo de una escala;

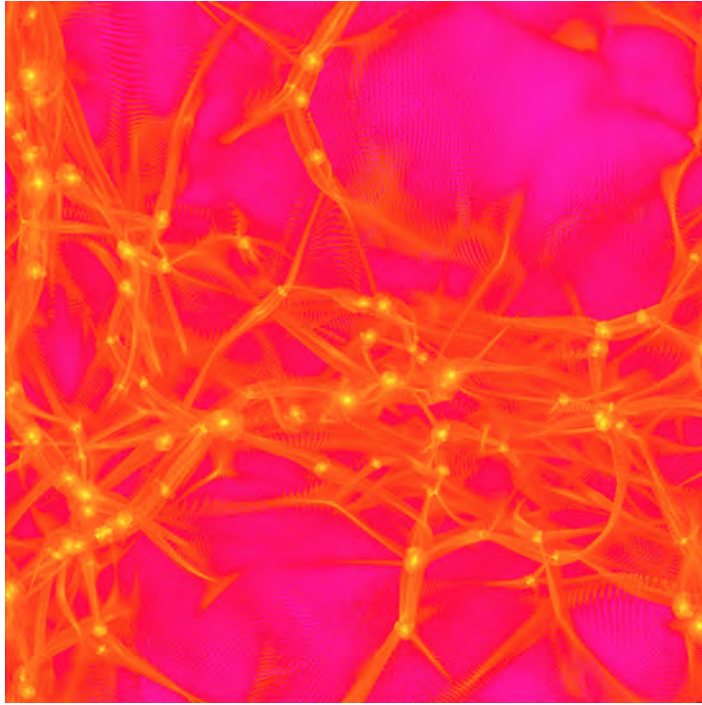


Figura 8.4: Campo de densidad para la materia oscura tibia, utilizando los parámetros de [155]. Note la ausencia de la fragmentación en los filamentos, lo cual es una ventaja intrínseca del código BPM.

la difusividad numérica la cual podría crear una fragilidad artificial en las estructuras; y la estrategia de cerradura para la jerarquía de momentos. Si bien la condición de cerradura no es perfecta, las pruebas de estabilidad de sistemas y del espectro de potencias a gran escala dan certeza de que no afectan de forma drástica a la abundancia de estructura. Seguramente las propiedades internas de los halos pueden ser sensibles, en particular la cinemática, evaluar esto requerirá un estudio sistemático posterior. A continuación cuantificaremos la importancia de la difusividad numérica y la resolución.

Pruebas numéricas para BPM: Difusividad y Resolución

La solución numérica de sistemas continuos tiene como limitante inherente la llamada difusividad numérica. Con el objetivo de caracterizar la amplitud de tal efecto realizamos tres pruebas. La primera de ellas es la estabilidad del perfil de densidad de un halo aislado. La prueba se ilustra en la figura 8.5, en esta se puede ver que una vez alcanzada la resolución en masa adecuada ($N_{\text{row}}^3 = 64^3, 128^3$ partículas) el código mantiene la estructura de densidad del sistema, a diferencia del caso con $N_{\text{row}}^3 = 32^3$ partículas. Además, alcanzada la resolución en masa adecuada la sensibilidad al paso de integración temporal es débil mientras sea menor que el tiempo de cruce a la velocidad v_{rms} (o al del sonido). Cabe mencionar que se realizó un estudio de convergencia para definir tal paso. Es importante enfatizar que esta prueba es altamente demandante, ya que al ser un corte del campo de densidad 3D cualquier fluctuación queda impresa, a diferencia de un perfil de densidad, como veremos en la próxima figura. Las fluctuacio-

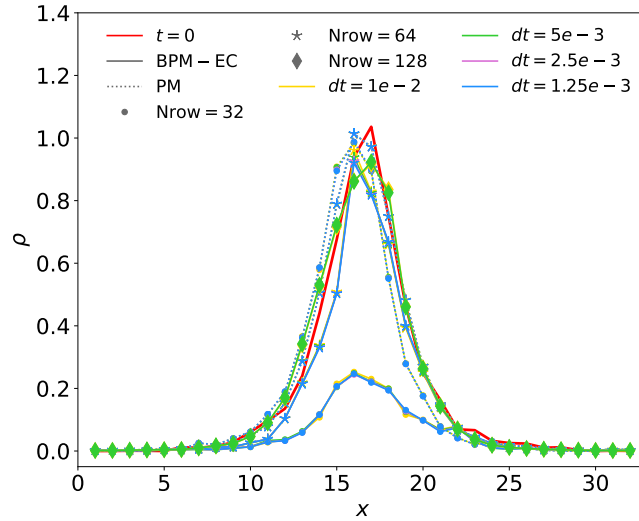


Figura 8.5: Halo aislado. La figura muestra un corte transversal de la densidad después de dos tiempos dinámicos, variando la resolución en masa (es decir, el número de partículas $N_p = N_{\text{row}}^3$) o el paso de integración, dt . Es notorio que el caso de menor resolución en masa $N_{\text{row}} = 32$ no puede sostener la densidad central, los casos con $N_{\text{row}} = 64, 128$ son poco sensibles al paso de integración.

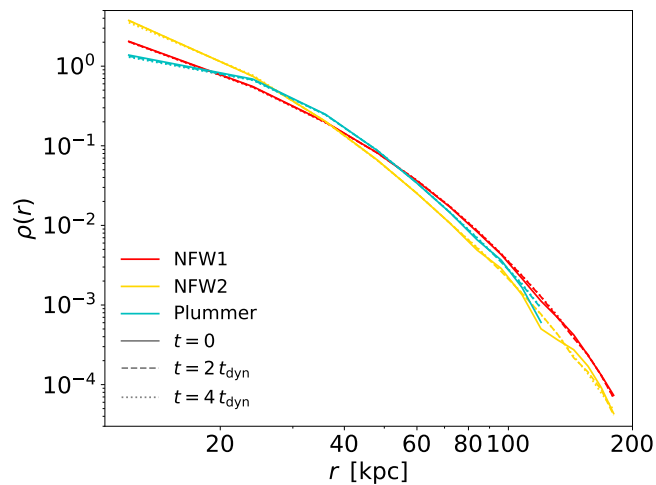


Figura 8.6: Conservación del perfil de densidad con BPM. La figura muestra el perfil de densidad para tres halos con diferentes modelos de densidad (Plummer, en celeste, y NFW, en rojo y amarillo), seguidos a lo largo de cuatro tiempos dinámicos usando BPM. Esencialmente no hay evolución.

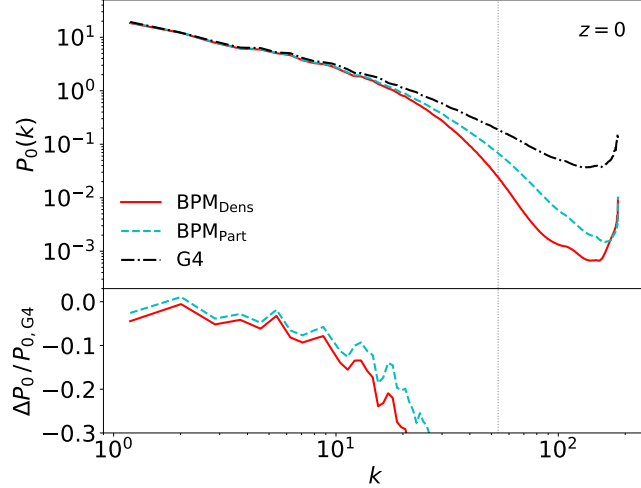


Figura 8.7: MPS para la simulación LCDM a $z = 0$. La caja simulada es de $7.5 \text{ Mpc}/h$, con $N_p = 256^3$ partículas y $N_g^3 = 256^3$ celdas. Se muestra el MPS medido para la simulación tanto de GADGET-4 (G4, negro), como para BPM ya sea trazado por las partículas prueba (celeste) o por la densidad de las celdas (rojo). A gran escala vemos diferencias entre 0% y 2% , lo que nos acota efectos de la difusividad, dado que las escalas están igualmente resueltas. A k 's mayores la resolución adaptiva de GADGET-4 ocasiona grandes diferencias respecto a BPM. Esto motiva a la necesidad de una versión híbrida o adaptiva de BPM. La línea punteada vertical indica la mitad de la frecuencia de Nyquist dada la malla.

nes vienen de utilizar partículas para las CI's, planeamos posteriormente inicializar la densidad con la densidad del modelo analítico para eliminar tales fluctuaciones. En la figura 8.6 mostramos la estabilidad del perfil de densidad para tres halos diferentes (un Plummer y dos NFW de distinta concentración) sobre varios tiempos dinámicos. La conclusión es que el código puede sostener sistemas autogravitantes por varios tiempos dinámicos sin estar dominado por difusividad.

Como segunda prueba de difusividad simulamos la formación de estructura tanto con BPM como con un código Tree-PM (GADGET-4 [168]), para un modelo LCDM. Como establecimos anteriormente un exceso de difusividad podría originar una fragilidad exacerbada de la estructura. La figura 8.7 muestra los resultados para ambos códigos a $z = 0$. Encontramos que la fracción de estructura es consistente entre ambas técnicas tan diferentes, a grandes escalas (k pequeño). El bino usado para calcular el MPS origina fluctuaciones, pero entre $k = 1$ y $5 \text{ Mpc}/h$ BPM es consistente con GADGET-4 (en adelante, G4, línea negra), $\sim 3\%$ para las partículas prueba como trazadoras (línea celeste) y $\sim 5\%$ para la densidad de las celdas como trazadoras (línea roja). De ser importante la difusividad parte de la potencia se vería amortiguada en zonas bien resueltas como las escalas mencionadas. Las grandes diferencias a muy pequeñas escalas son originadas por la diferencia en resolución, la cual es un factor de diez mejor para G4. Una mayor resolución que se traduce a mayor número de celdas permitiría mejor acuerdo con LCDM en las escalas resueltas. Esto pone de manifiesto la necesidad de una versión híbrida de BPM o una adaptiva.

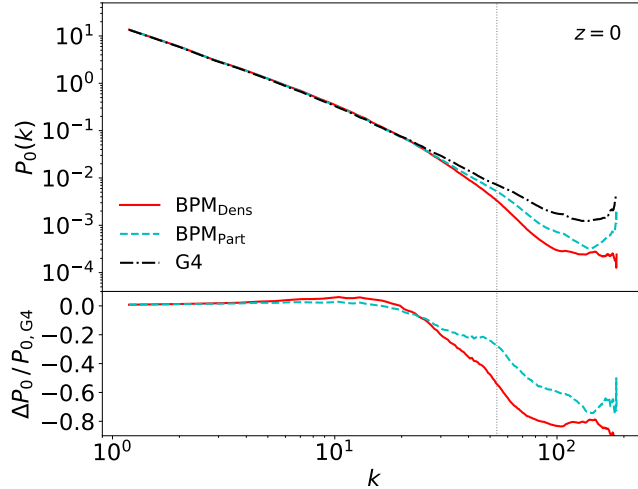


Figura 8.8: MPS a $z = 0$ para un modelo LWDM usando BPM y G4. Ambos códigos capturan correctamente las escalas grandes y la evolución lineal, reduciendo el posible efecto de la difusividad a unos cuantos por cientos. La discrepancia a pequeñas escalas (k 's grandes) es originada por la diferencia en resolución espacial. Para el caso de partículas prueba como trazadoras del MPS (celeste) el ruido poissonico crea potencia espuria durante la medición.

Como tercera prueba, simulamos la formación de estructura tanto con BPM como con G4 para un modelo WDM con una masa para la partícula tibia de $m_{\text{wdm}} = 0.25$ keV, usando una caja de 7.5 Mpc/h con $N_p = 256^3$ partículas y $N_g^3 = 256^3$ celdas. La figura 8.8 muestra el espectro de potencias de materia para ambos códigos a $z = 0$. A diferencia de [133], encontramos que, ambos códigos tienen la misma fracción de estructura a escalas grandes. A escalas pequeñas hay una diferencia entre ambos códigos, lo cual puede deberse a la diferencia de resolución en ambas simulaciones, combinado con el hecho de que BPM no presenta shot-noise, a diferencia de G4. Es importante señalar el siguiente punto, utilizar las partículas trazadoras para estimar la densidad y el MPS tiene un ruido poissonico asociado al proceso de medición, si se utiliza la densidad de la solución BPM directamente tal ruido no existe, esto se aprecia en la figura 8.8. Lo anterior es relevante para interpretar los resultados de las simulaciones con BPM.

8.5.2. Simulaciones Híbridas: BPM+GADGET-4

Dado que una de las áreas de interés es el estudio de la subestructura, la resolución fija es una limitante para BPM y para cualquier código de malla fija (p. ej. PM) a pesar de las ventajas de BPM en términos de la descripción del campo de densidad. En particular, en la literatura se busca amortiguar el origen de una población de halos espurios (ver sección 8.4), causados por el shot-noise en las condiciones iniciales, aumentando el número de partículas a miles de millones, además de un proceso artesanal para identificar posibles halos artificiales. En esta tesis proponemos una estrategia distinta. Proponemos dos fases para la simulación, la primera de $z = 127$ hasta el

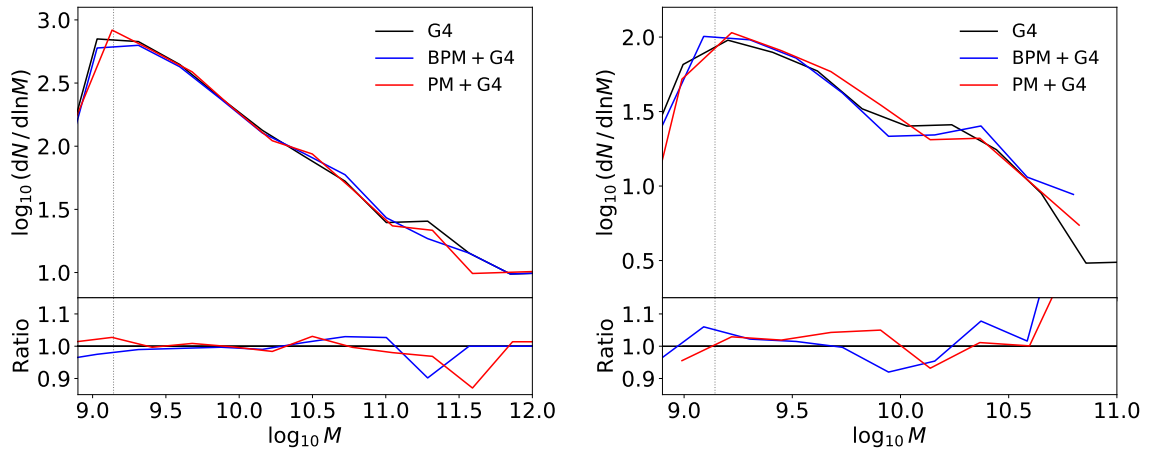


Figura 8.9: Distribución diferencial de masa para todos los halos (izquierda) y de todos los subhalos (derecha) en un universo LCDM a $z = 0$. La línea negra representa los resultados usando G4 desde las condiciones iniciales ($z = 127$), la azul usando BPM+G4 (cambiando de BPM a G4 en $z_c = 5$) y la roja usando PM+G4 (cambiando en $z_c = 5$). Los halos fueron hallados con el software Rockstar. La línea vertical punteada muestra la resolución de masa dado el número mínimo de partículas usado por halo (512 partículas). Los paneles inferiores muestran el cociente respecto a la simulación G4 (línea negra). Salvo fluctuaciones (posiblemente resultado de la elección de bins) los tres resultados coinciden. El acuerdo evidencia que los resultados BPM+G4 no están dominados por efectos de resolución o difusividad numérica y que en modelos con supresión del MPS, no habrá ausencia artificial de halos. Esta prueba es crítica para dar confiabilidad al estudio con supresión en el MPS.

momento en que los halos más pequeños resueltos se vuelvan autogravitantes, ese momento se estima en base a cálculos basados en el modelo de colapso esférico [169], a ese momento lo denominamos z_c . En ese instante continuamos la simulación con el código GADGET-4 (G4). A este conjunto de simulaciones las denominamos BPM+G4. A continuación presentamos una serie de pruebas para cuantificar la precisión y posibles sesgos de tales experimentos. De interés particular para esta tesis es la estadística en la abundancia de halos y subhalos.

Simulamos un caso sin corte en el espectro de masas, es decir LCDM. A $z = 0$ se identificó a la población de halos utilizando el software Rockstar [170]. El lado izquierdo de la figura 8.9 muestra la función de masa de todos los halos para los códigos G4 (línea negra), BPM+G4 (línea azul) y PM+G4 (línea roja), con éstos dos últimos haciendo el cambio a $z_c = 5$. Esta figura ilustra que en particular la abundancia de estructura no se ve afectada de forma significativa debido a las diferencias en las técnicas numéricas ya que los tres códigos producen la misma cantidad de estructura. Se observa una fluctuación en los halos más pequeños, posiblemente resultado del buscador de halos, así como el bino del histograma. Una prueba más exigente es la abundancia de subhalos, sujeta además del problema de difusividad de forma más crítica a límites de resolución. En este caso también se logra convergencia entre las técnicas como se aprecia en lado derecho de la figura 8.9. No es trivial lograr este acuerdo y demuestra que es posible estudiar la población de subhalos con la técnica híbrida BPM+G4 sin que esté dominada por difusividad.

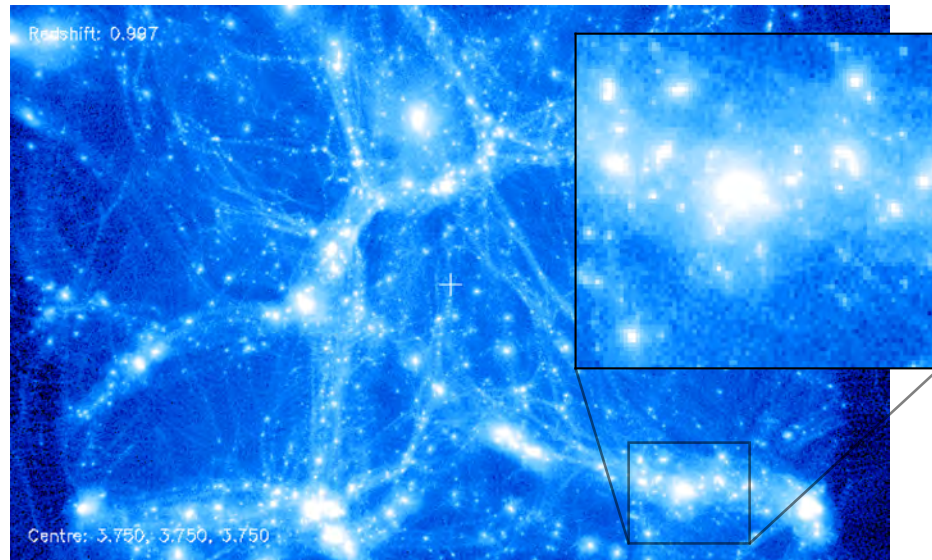
8.5.3. Simulaciones WDM BPM+GADGET-4

Una vez confirmada la robustez de la técnica BPM+G4 se realizaron simulaciones combinando el enfoque Boltzmann hasta $z_c = 1$, y a partir de ese momento el sistema se integró con el código G4, esto con el fin de revisar convergencia y lograr una mayor resolución espacial dado el carácter adaptivo de GADGET. Elegimos $z_c = 1$ debido a que es el momento en que estimaciones basadas en la técnica del colapso esférico, predicen el colapso de halos de $10^8 M_\odot$, lo cual está en la parte robustamente simulada por nuestro esquema y corresponde a halos de materia oscura cuya abundancia se predice ser substancialmente sensible al tipo de materia oscura. Los resultados indican que los límites a la masa de la partícula de materia oscura tibia podrían estar sobre estimados, una consecuencia de esto es que masas de partículas menores estarían permitidas, como podría ser otro neutrino.

En la figura 8.10 se muestra la misma sección del volumen simulado a $z = 1$ usando GADGET-4 (panel superior) y BPM (panel inferior) para la simulación WDM con masa para la partícula tibia $m_{\text{wdm}} = 3.5 \text{ keV}$. Es notorio que la simulación con BPM presenta menos estructura en escalas pequeñas que la simulación de GADGET-4, debido a que BPM no presenta shot-noise a diferencia de GADGET-4.

La figura 8.11 muestra el espectro de potencias de materia a $z = 1$ para las mismas simulaciones que en la figura 8.10. Vemos que BPM y G4 coinciden a escalas grandes (k 's pequeñas) indicando que no hay efectos de difusividad notorios. Mientras que a escalas pequeñas la diferencia es debida a que BPM no incluye a las estructuras originadas del shot-noise de las condiciones iniciales, así como las del proceso de medición

GADGET-4



BPM

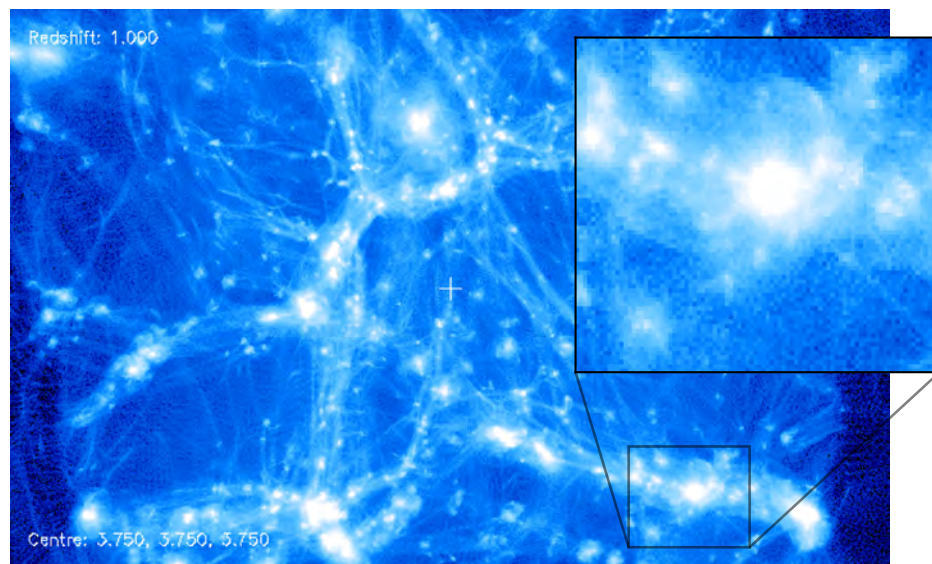


Figura 8.10: Sección del volumen simulado, a $z = 1$, usando GADGET-4 (panel superior) y BPM (panel inferior). La simulación con GADGET-4 tiene shot-noise, a diferencia de la de BPM. Notar que con BPM hay menos estructura en escalas pequeñas en comparación con GADGET.

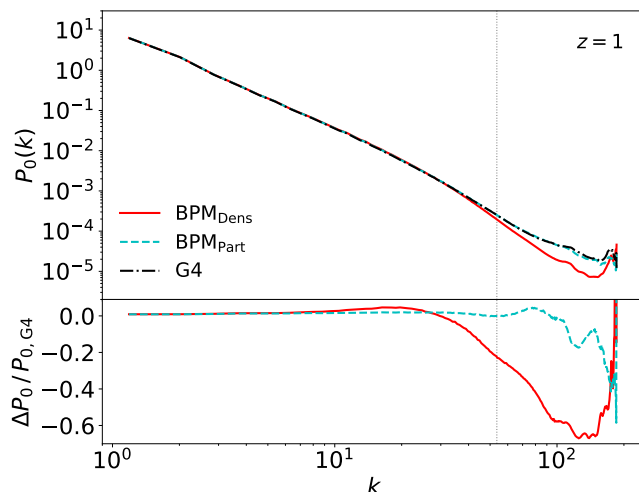


Figura 8.11: MPS a $z = 1$ para la simulación WDM usando ambos códigos (BPM y G4). El acuerdo a escalas grandes construye la posibilidad de efectos de difusividad numérica. El desacuerdo en escalas pequeñas es debido a que BPM no tiene shot-noise cuando el trazador son las celdas. La línea punteada vertical indica la mitad de la frecuencia de Nyquist dada la malla.

del MPS. El efecto es sutil, ya que las partículas pruebas en BPM no son fuente de gravedad por lo que sus fluctuaciones no crecen y solo están en el proceso de medición. El panel inferior de la figura 8.11 muestra un buen acuerdo con G4 hasta $k \sim 100$ Mpc/h. Esto no es sorprendente dado que a $z = 1$ apenas se comienzan a volverse autogravitantes halos de $10^8 M_\odot$.

En la figura 8.12 se muestra la función diferencial de masa de todos los halos (izquierda) y de todos los subhalos (derecha) a $z = 0$, para las simulaciones usando G4 de principio a fin (línea amarilla) y usando BPM+G4 (línea verde) haciendo el cambio de códigos en $z_c = 1$. Como comparación también se muestra el resultado para la simulación LCDM (línea morada). Se observa que el caso BPM+G4 presenta menos halos de baja masa que G4 “completo”, y como es de esperarse ambas simulaciones presentan una supresión en la población de halos de baja masa en comparación con LCDM, debido a la supresión (o corte) en el espectro de potencias inicial. Es importante notar que por debajo de $7.6 M_\odot$ la abundancia se levanta de nuevo, sin embargo con menor pendiente que el caso G4. Se concluye que las fluctuaciones presentes al realizar el cambio BPM a G4 logran crecer, aunque de forma neta disminuye el número de subhalos pues tienen menor tiempo para crecer.

La figura 8.13 muestra la función diferencial de masa de los subhalos de los diez halos más masivos para cada simulación de la figura 8.12. Las líneas muestran el promedio de cada caso. El resultado es similar al obtenido en dicha figura, reafirmando que con la técnica BPM+G4 podemos realizar simulaciones de cosmologías con supresión en el espectro de potencias sin tener shot-noise y por ende sin necesidad de hacer un proceso de selección de halos meticuloso. El panel derecho de la figura 8.13 muestra la función de masa de los subhalos de los halos que tienen masas entre 10^{11} y $3 \times 10^{12} M_\odot/h$, se observa un comportamiento similar a las figuras previas.

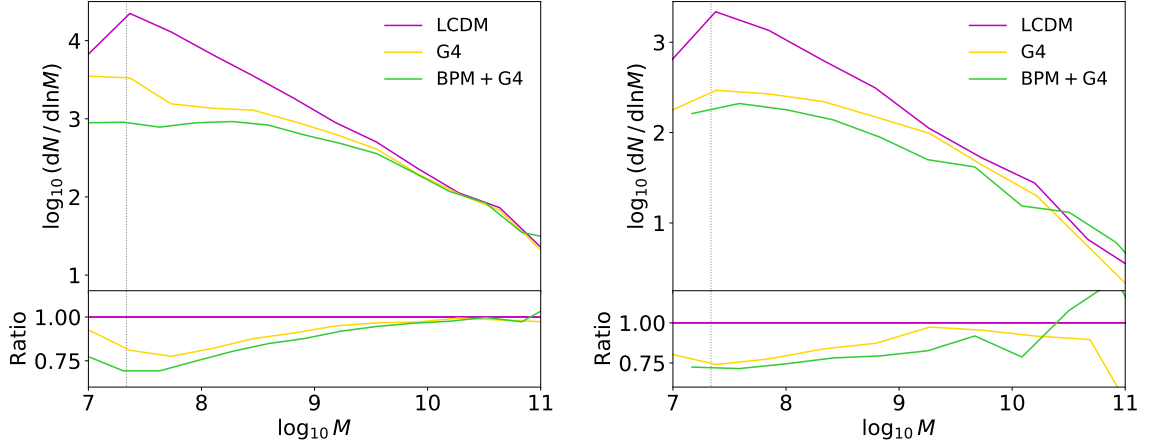


Figura 8.12: Función de masa de todos los halos (izquierda) y de todos los subhalos (derecha) a $z = 0$, para LCDM (línea magenta), G4 desde las condiciones iniciales (línea amarilla) y BPM+G4 haciendo el cambio en $z_c = 1$ (línea verde). También se muestra el cociente respecto a LCDM en cada caso (paneles inferiores). La línea vertical punteada muestra la resolución de masa dado el número mínimo de partículas por halo usado (64 partículas). Las simulaciones WDM muestran una supresión esperada para halos de baja masa en comparación con LCDM, siendo mayor para BPM+G4.

En particular en el panel de la izquierda 8.13 la predicción de BPM+G4 (verde) tiene un factor de ~ 1.6 menos subhalos que la predicción de LCDM, en acuerdo con [171], y más importante sin realizar un procedimiento artesanal (como el descrito en [162]) y con menos recursos en tiempo y memoria (10^6 CPU hr vs 8000 CPU hr).

8.6. Otros modelos de Materia Oscura con amortiguación en el MPS

Recientemente, con el descubrimiento de la partícula de Higgs, la cual está asociada a un campo escalar, tomó fuerza la posibilidad de que la materia oscura estuviese compuesta por uno de estos campos con una partícula asociada ultra ligera $m_{\text{FDM}} \approx 10^{-22}$ eV llamada también Fuzzy Dark Matter. Incluir como partícula este caso es técnicamente imposible para códigos de N -cuerpos (debido al shot-noise), por lo que códigos como el propuesto en esta tesis son indispensables. Alternativamente, se puede utilizar la ecuación de Schrödinger-Poisson la cual sigue al sistema en términos de la función de onda, es decir, medio continuo. Ambos enfoques, el Boltzmann-PM y el Schrödinger-Poisson, tienen un traslape a longitudes de De Broglie grandes y se puede mapear de una solución a otra [172]. Es importante mencionar también que en el caso de los códigos para resolver Schrödinger-Poisson, basados en métodos espectrales como los códigos PM, no se ha discutido aún de forma extensa el efecto de las limitantes numéricas de éstos (condiciones a la frontera, aliasing o solapamiento debido a FFT, captura de vorticidad) en las estructura de la distribución de materia oscura y la fricción

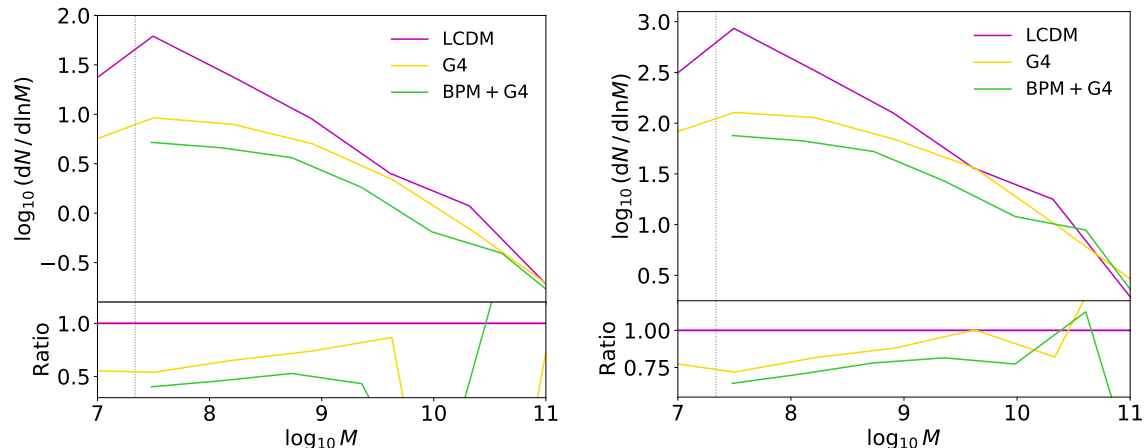


Figura 8.13: Como 8.12, pero para los subhalos de los diez halos más masivos (izquierda) y de los halos que tienen masas entre 10^{11} y $3 \times 10^{12} M_{\odot}/h$ (derecha) de cada simulación.

dinámica asociada [173].

De forma alternativa se ha propuesto que la materia oscura represente un grupo totalmente independiente del modelo estándar, posiblemente originado durante la transición QCD¹ (por sus siglas en inglés). Esta propuesta denominada “Bound Dark Matter” (BDM) [174, 175] asume que la materia oscura sufre una transición de fase entre fría (baja temperatura cinética) y caliente (relativista), afectando el crecimiento de las fluctuaciones y amortiguando el espectro de potencias. Este código puede capturar tales procesos de forma auto-consistente. Dependiendo de la masa asociada al campo escalar existe un régimen en el que la solución Boltzmann/Vlasov y la Schrödinger-Poisson convergen entre sí [172], justificando la discusión de esta sección.

8.7. Discusión

La aplicación del código desarrollado en esta tesis, BPM, al problema de la estructura en modelos con supresión en el espectro de potencias de materia es exitosa. En particular, se estudió el efecto de la difusión numérica y de la resolución, y se concluyó que no dominan el resultado para los parámetros considerados, a diferencia de lo que menciona [133] (aunque no lo muestra). Lo anterior puede deberse a que nosotros consideramos la forma conservativa de las ecuaciones obtenidas de la jerarquía de Boltzmann a diferencia de ellos, sin embargo, se requiere más estudio a futuro para entender la diferencia con tales resultados.

Para contrarrestar la limitante de resolución que supone usar cualquier código de malla fija, propusimos una estrategia que consta de dos fases de la simulación. En la primera se usa el código BPM desde las condiciones iniciales hasta el corrimiento al

¹QCD se refiere a Quantum Chromodynamics. Los constituyentes básicos de QCD son quarks y gluons, los cuales están confinados en hadrones (energías bajas). A temperaturas altas y/o densidades, se produce el desconfinamiento de quarks y la materia hadrónica sufre una transición de fase a un nuevo estado de la materia, el plasma quark-gluón.

rojo z_c en el que los halos resueltos más pequeños se vuelven autogravitantes. Para la segunda fase se sigue la evolución del modelo cosmológico desde z_c hasta la época actual ($z = 0$) usando, en particular, GADGET-4. En el capítulo mostramos que esta estrategia es robusta y converge a las simulaciones individuales para LCDM, como es de esperarse.

Todo lo anterior nos permite estudiar la formación de estructura sin efectos de shot-noise o al menos con una gran disminución a un costo computacional mucho menor que las técnicas puramente basadas en partículas, pues dada la descripción de la densidad como medio continuo no se requiere de un gran número de elementos de resolución (partículas) para disminuir estadísticamente el shot-noise, y al combinar BPM con otra técnica como lo es Tree (GADGET-4) podemos tener simulaciones de mayor resolución espacial. Otra ventaja de usar BPM sobre los métodos tradicionales de N -cuerpos, es que no se requiere un procedimiento artesanal para seleccionar los subhalos, como la descrita en [162]. Lo anterior permite que la exploración de las predicciones en la abundancia y propiedades de la subestructura en modelos de materia oscura con supresión del espectro de fluctuaciones es mucho más eficiente que la opción completamente N -cuerpos. Los códigos PM, tienen un comportamiento similar, sin embargo, a cada paso de integración el ruido participa en el CIC que construye el campo de densidad, en el potencial y por lo tanto en las velocidades, esto será relevante principalmente en sistemas simulados por varios tiempos dinámicos, no así en sistemas dinámicamente jóvenes. Una versión adaptiva de BPM terminaría de resolver el problema de forma completa. Finalmente desarrollos como los presentados por [176] son consistentes con nuestros resultados, pero difieren en que no pueden incluir el caso tibio, es decir con velocidades térmicas mientras que BPM si puede, como se ilustra en los capítulos con neutrinos.

Capítulo 9

Estado actual de las técnicas usadas en simulaciones cosmológicas de formación de estructura con neutrinos masivos

El descubrimiento de la oscilación de sabor de los neutrinos, con experimentos de laboratorio, ha establecido fuertemente que los neutrinos son partículas masivas, lo cual es una evidencia concluyente de nueva física más allá del modelo estándar de partículas.

De los experimentos de oscilación en laboratorio se sabe el ordenamiento de dos de los tres estados propios de masa de los neutrinos, sin embargo, no es posible saber el ordenamiento del estado restante con estos experimentos. Por lo anterior, existen dos escenarios: (1) el primero es la jerarquía normal (NH, por sus siglas en inglés), en el cual el tercer estado de masa es más masivo que los otros dos, (2) el segundo es la jerarquía inversa (IH) en el que el tercer estado es el menos masivo. Como vimos en el capítulo 2, el poder distinguir entre estos dos escenarios tiene implicaciones importantes en nuestra comprensión del Universo, ya que ayudaría a entender la bariogénesis y la simetría bariónica de éste. Sin embargo, ya sea en tal rango de masas o combinado con futuros experimentos terrestres como DUNE [177] y KATRIN¹, las constricciones cosmológicas serán relevantes.

Afortunadamente, es posible medir la suma de las masas de los neutrinos y determinar la jerarquía de los estados de masa de éstos desde un punto de vista cosmológico, debido a que los neutrinos dejan una huella característica (la cual depende de la masa de éstos) en los observables cosmológicos. Cabe mencionar que estudios recientes cuestionan la sensibilidad de las pruebas cosmológicas para restringir la jerarquía de masas, al menos si la suma de masas de las tres especies de los neutrinos es mayor que 0.1 eV. Es una situación de disyuntiva, pues entre más masivos sean los neutrinos, los efectos no-lineales, incluso en la distribución de neutrinos, serían no despreciables. Si alternativamente son muy ligeros, se podría constreñir la jerarquía utilizando observaciones cosmológicas. Ambas opciones abren la puerta a nueva física y/o requieren de

¹<https://www.katrin.kit.edu/>

nuevas técnicas.

Por otro lado, la cosmología observacional está alcanzando niveles de alta precisión (DESI logrará errores menores al 1% en la escala de BAO: 0.3% hasta $z = 1$, 1% a $z = 3$), por lo que se espera que con los sondeos de galaxias contemporáneos y futuros (BOSS, eBOSS, DESI, LSST, Euclid, etc.) se pueda restringir la suma total de masa de los neutrinos con 3σ o incluso con 5σ , y que, posiblemente, y si las condiciones son propicias, se pueda determinar la jerarquía de la masa de los neutrinos. Sin embargo, para lograr las metas científicas de estos sondeos se requiere de catálogos sintéticos realistas basados en simulaciones numéricas que permitan calcular el crecimiento no-lineal de la estructura cósmica y predigan la dependencia de las cantidades observables en los sondeos con los parámetros cosmológicos. Por lo tanto, resulta importante que las simulaciones tengan la calidad de precisión de las observaciones, y por ende requieren de métodos de cálculo e infraestructura con alta eficiencia en cómputo.

Dicho lo anterior, en este capítulo presentamos resultados preliminares de un trabajo aún en desarrollo en el cual pretendemos mostrar una comparación cuantitativa entre algunas de las técnicas en el estado del arte para incluir una componente explícita de neutrinos masivos en las simulaciones cosmológicas, incluyendo el código Boltzmann-PM (BPM) desarrollado en esta tesis.

9.1. Descripción de los códigos considerados

A continuación describimos los códigos que consideramos hasta ahora para la comparación de técnicas para simulaciones cosmológicas con neutrinos masivos (ν CDM).

9.1.1. Quijote

Las simulaciones Quijote [120] son un conjunto de 43100 simulaciones de N -cuerpos (usando la técnica Cold+Hot) barriendo más de 7000 modelos cosmológicos en el hiperplano de parámetros siguiente $\{\Omega_m, \Omega_b, h, n_s, \sigma_8, M_\nu, w\}$. A un corrimiento al rojo definido, las simulaciones incluyen más de 8.5 billones de partículas dentro de un volumen combinado de 43100 $(h^{-1}\text{Gpc})^3$. La realización de estas simulaciones ha tomado más de 35 millones de horas de CPU. Sus objetivos han sido: 1) cuantificar el contenido de información en las observables cosmológicas y 2) proveer una muestra de entrenamiento para algoritmos de Machine Learning. Las simulaciones se realizaron con el código TreePM GADGET-3 [36].

Las condiciones iniciales se generaron para todos los experimentos a $z = 127$. En particular, para los modelos con neutrinos masivos, las condiciones iniciales se generaron de la siguiente forma. El espectro de potencias de materia (MPS, por sus siglas en inglés) y la función de transferencia iniciales se obtuvieron usando el software REPS² que implementa el método perturbativo de reescalamiento descrito en [178], el cual asume dos especies (aproximación de dos-fluidos) y una componente de radiación (si bien es un efecto pequeño, al buscar precisiones menores al 1% es relevante [117]). A partir de la función de transferencia y MPS iniciales calcularon los desplazamientos

²<https://github.com/matteozennaro/reps>

y velocidades peculiares de las partículas usando la aproximación de Zeldovich [85], usando una versión modificada del software N-GenIC, llamada N-GenIC_growth³. Las velocidades térmicas de los neutrinos fueron generadas muestreando aleatoriamente la distribución de Fermi-Dirac al z inicial (es sabido que esto también puede inducir shot-noise en la simulación).

Las simulaciones Quijote representan un gran esfuerzo para la exploración del espacio de parámetros disponible, sin embargo, en el trabajo no discuten el efecto del shot-noise en las simulaciones y su interpretación.

9.1.2. FastPM

FastPM [141] es un código cuasi- N -cuerpos que implementa la técnica Partícula-Malla, similar a la discutida en el capítulo 6.2.1 (y detallada en B), pero utiliza una función de selección de paso de integración basada en una solución perturbativa para las fluctuaciones de materia oscura, construyendo el error en el espectro de potencias de masa [141]. Esto le permite en principio utilizar pasos grandes sin disminuir la precisión en el espectro de fluctuaciones en densidad, lo que da como resultado simulaciones rápidas. Recientemente, [91] han utilizado esta técnica para realizar simulaciones cosmológicas que incluyan el efecto de neutrinos masivos.

Al igual que en las simulaciones Quijote, para generar las condiciones iniciales usan el código de reescalamiento REPS [178], y la aproximación de Zeldovich (no especifican el software utilizado). Para las velocidades térmicas de los neutrinos usan el método desarrollado en [121], discutido en el capítulo 5.3.

Cabe mencionar que el análisis perturbativo que define el paso de integración, se realizó considerando sólo a la materia oscura, por lo cual en el caso con neutrinos tal esquema no es formalmente correcto y el código se comporta como un PM normal, obviamente el efecto solo será relevante en comparaciones de alta precisión. También, es importante mencionar que ajustar el paso de integración a lo largo del tiempo (cambio logarítmico a alto z y lineal posteriormente), puede reflejarse en efectos dinámicos a pequeña escala y afectar propiedades de los halos como masa y V_{\max} .

9.1.3. MG-PICOLA

MG-PICOLA es un código híbrido PM con una corrección perturbativa lagrangiana a orden lineal o a segundo orden (2LPT), que puede simular cosmologías que exhiben un crecimiento dependiente de la escala. En particular, para cosmologías con neutrinos masivos, sólo se sigue a las partículas de CDM, mientras que el efecto de los neutrinos sólo es incluido como efecto de fondo sobre la malla, al igual que en [126]. Cabe mencionar que la precisión que logra esta técnica en capturar el efecto de los neutrinos masivos en el crecimiento de estructura depende de la precisión de la solución perturbativa y de lo adecuado de la solución de la malla (resolución espacial, en masa, paso de integración).

³https://github.com/franciscovillaescusa/N-GenIC_growth

Código	n_s	σ_8	M_ν (eV)	$N_{\text{cdm}}^{1/3}$	$N_\nu^{1/3}$	Res. (Mpc/h)	z_{ini}	CI's
Quijote	0.9624	0.834	0.4	512	512	0.05	127	REPS+N-GenIC_growth+rFD
FastPM			0.2			3.91	99	REPS+Zeldovich+[121]
MG-PICOLA	0.966	0.8503	0.2			3.91	49	2LPT+ec.9.1
BPM	0.9624	0.834	0.4	512	512	1.95	127	REPS+N-GenIC_growth+rFD

Cuadro 9.1: Parámetros usados para las simulaciones. $M_\nu = \Sigma m_\nu$ es la suma de las masas de los neutrinos, Res. se refiere a la resolución espacial de la simulación, z_{ini} es el corrimiento al rojo inicial, REPS es el código de reescalamiento descrito en [178], N-GenIC_growth es el software utilizado para la aproximación de Zeldovich, y rFD se refiere a un muestreo aleatorio de la distribución de Fermi-Dirac para las velocidades de los neutrinos.

Para generar las condiciones iniciales, para CDM usan la teoría 2LPT (del inglés 2nd-order Lagrangian Perturbation Theory), mientras que para los neutrinos su densidad inicial, $\delta_\nu(\vec{k}, \tau_{\text{ini}})$, es calculada con:

$$\delta_\nu(\vec{k}, \tau_{\text{ini}}) = \delta_{\text{cb}}(\vec{k}, \tau_{\text{ini}}) \frac{T_\nu(k, \tau_{\text{ini}})}{T_{\text{cb}}(k, \tau_{\text{ini}})} \quad (9.1)$$

donde, δ_{cb} es la densidad CDM+bariones, T_ν y T_{cb} son las funciones de transferencia de los neutrinos y de CDM+bariones, respectivamente.

9.1.4. BPM

Finalmente, nuestro código BPM que resuelve la ecuación de Boltzmann a través de la jerarquía de momentos de ésta. En el capítulo 6 se describen los detalles numéricos. Como se mencionó, éste código no incluye shot-noise debido a la discretización del campo de densidad, si bien maneja la parte cinemática a través de la ecuación de Euler, es una opción cara computacionalmente. Como mencionamos en el capítulo 6, existen dos versiones del código para simulaciones con dos especies. En este capítulo usaremos la versión BPM-PM, en la cual los neutrinos son resueltos con el código BPM, mientras que CDM con el código PM, y ambas especies están acopladas gravitacionalmente mediante la ecuación de Poisson.

En nuestras pruebas elegimos emplear las condiciones iniciales utilizadas por el proyecto Quijote, las cuales incluyen dos especies (neutrinos y CDM) y el efecto del campo de radiación en el universo temprano. Esto tiene efectos mejorando la precisión en la escala de flujo libre de los neutrinos.

La simulación con la que comparamos resuelve la ecuación tipo continuidad (ec. 6.4). Como mostramos en el capítulo 6, utilizar la ecuación tipo continuidad 6.4 únicamente crea una mayor supresión que el caso Euler, esto será relevante solo en casos de alta precisión.

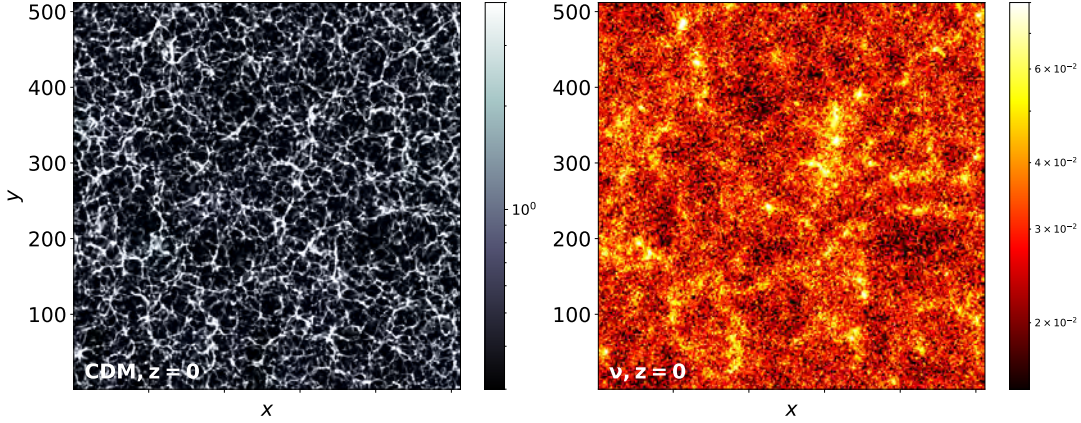


Figura 9.1: Estructura en la simulación con BPM. Rebanada (en el plano XY) del campo de densidad de neutrinos (derecha) y CDM (izquierda) para $z = 0$.

9.2. Resultados preliminares

En todas las simulaciones presentadas en este capítulo la caja simulada es de 1000 Mpc/h por lado, $\Omega_m = 0.3175$, $\Omega_b = 0.049$, y $h = 0.6711$. La tabla 9.1 presenta el resto de los parámetros utilizados en las simulaciones.

La figura 9.1 muestra una rebanada (plano XY) del campo de densidad de la componente CDM (izquierda) y de los neutrinos (derecha) a $z = 0$ para la simulación con BPM. De esta figura podemos ver la estructura formada en la simulación, en especial vemos que la componente de neutrinos traza la estructura de la red cósmica subyacente establecida por la componente CDM, pero es difusa a escalas pequeñas. Es importante mencionar que para simulaciones puramente de N -cuerpos no es trivial visualizar la estructura de la componente de neutrinos debido al shot-noise, en especial la masa de los neutrinos es pequeña, ver figura 5.2 (tomada de [137]).

En la figura 9.2 se muestra una comparación del espectro de potencias de materia (MPS) total (ν +CDM, fila de arriba) entre la simulación realizada con BPM (línea amarilla) y la correspondiente de Quijote (línea negra). Cabe aclarar que el MPS de ambas simulaciones fue calculado con herramientas diferentes. Salvo fluctuaciones de BPM (originadas por el bino usado al calcular el MPS de éste), a escalas grandes (k 's pequeñas) ambas simulaciones son consistentes. En contraste, para $0.1 \lesssim k \lesssim 0.8$ Mpc/h BPM muestra menos potencia debido a que no tiene shot-noise a diferencia de Quijote, esta diferencia es de entre 5 y 8% a $z = 0.5$, como puede verse en el panel de en medio de la figura. Los efectos para $k \gtrsim 0.8$ Mpc/h son debidos a la diferencia entre la resolución espacial de cada simulación. En la fila inferior de la misma figura se muestra la supresión del MPS al comparar las simulaciones con neutrinos con la correspondiente simulación sin neutrinos (es decir, puramente Λ CDM). Podemos apreciar que BPM presenta una mayor supresión (entre 10 y 20%) del MPS que Quijote para $0.1 \lesssim k \lesssim 0.8$ Mpc/h.

La figura 9.3 es como la figura 9.2 pero comparando FastPM con MG-PICOLA. En este caso se observa una mayor diferencia entre ambas simulaciones a escalas pe-

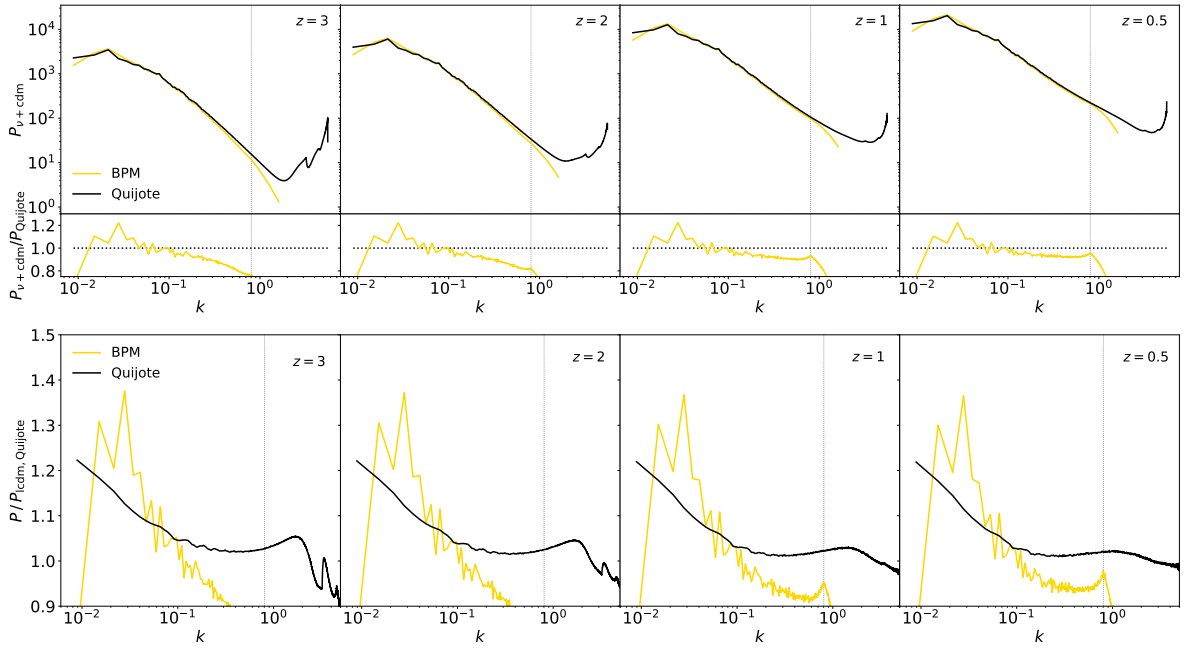


Figura 9.2: MPS total (ν +CDM, fila de arriba) y supresión del MPS total respecto a LCDM (fila de abajo) para una de las simulaciones Quijote (línea negra) y para la simulación con BPM (línea amarilla) en cuatro corrimientos al rojo: $z = 3, 2, 1$ y 0.5 , de izquierda a derecha, respectivamente. Las líneas punteadas verticales indican la mitad de la frecuencia de Nyquist dada la malla. BPM no presenta shot-noise.

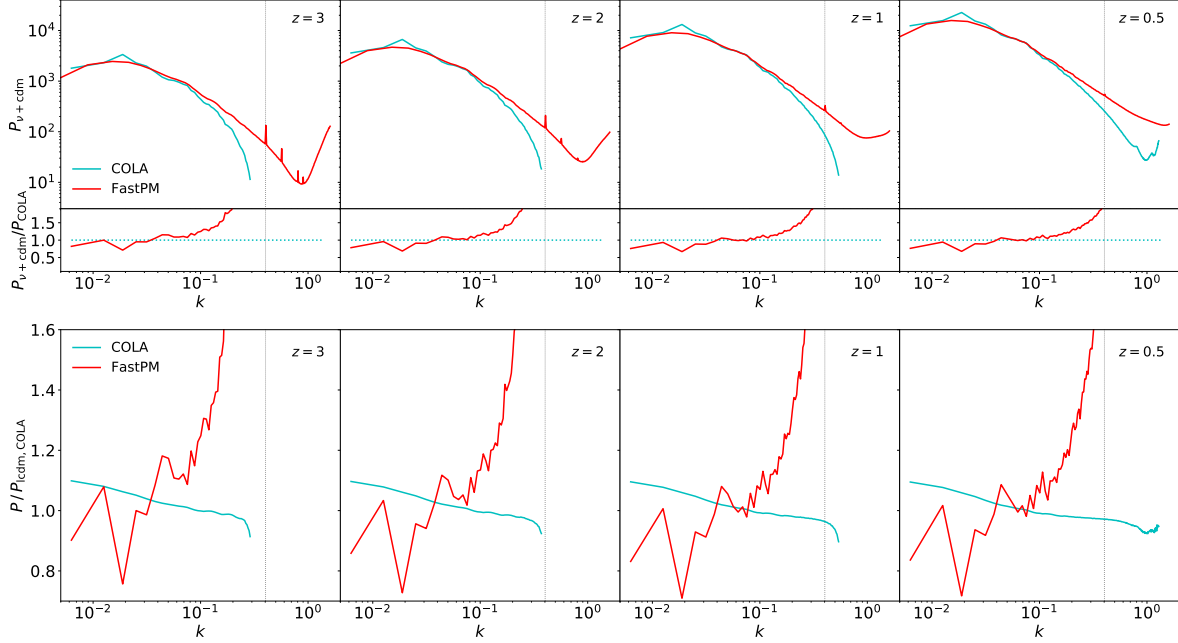


Figura 9.3: Como la figura 9.2, pero para las simulaciones con FastPM (rojo) y COLA (cían). Las diferencias entre las simulaciones son mayores que en el caso BPM vs. Quijote. El método de cancelación en CI's es menos eficiente a escalas pequeñas y podría ser responsable de la falta de supresión con FastPM, sin embargo, el efecto de la malla para los neutrinos puede ocasionar parte de la estructura. Las líneas punteadas verticales indican la mitad de la frecuencia de Nyquist dada la malla. BPM no presenta shot-noise.

queñas ($k \gtrsim 0.07$ Mpc/h), en donde COLA presenta menos potencia que FastPM. A escalas grandes ($k \lesssim 0.07$ Mpc/h) ambas simulaciones son consistentes. En cuanto a la supresión del MPS (fila inferior de la figura), el comportamiento de ambos códigos es diferente, sin embargo, es importante aclarar que el MPS de LCDM considerado para calcular la supresión para ambas simulaciones fue el obtenido de la simulación usando MG-PICOLA, y de ahí puede venir parte de la diferencia en el comportamiento. También, la diferencia puede ser a causa de que la malla de neutrinos imprima estructura en la distribución CDM, como lo discuten en [125, 91]. Sin embargo, se requiere de mayor estudio para entender esta diferencia.

Finalmente, por motivos comparativos con trabajos en el campo (por ejemplo [91]), realizamos una comparación entre la simulación FastPM y una simulación Quijote con $M_\nu = 0.2$ eV (el resto de parámetros es igual a la simulación Quijote descrita en la tabla 9.1), a la cual le llamaremos Quijote2. La figura 9.4 muestra los resultados. Hay algunas diferencias entre ambas simulaciones, pero en general, son consistentes hasta $k \lesssim 0.4$ Mpc/h, a partir de ahí comienzan a diferir teniendo menos estructura la simulación con FastPM. La diferencia entre ambas simulaciones se deba a la diferencia en resolución espacial de éstas, las líneas punteadas verticales indican la mitad de la frecuencia de Nyquist tomando en cuenta la resolución espacial de FastPM. Cabe mencionar que para obtener la supresión del MPS (fila inferior de la figura) de FastPM se tomó

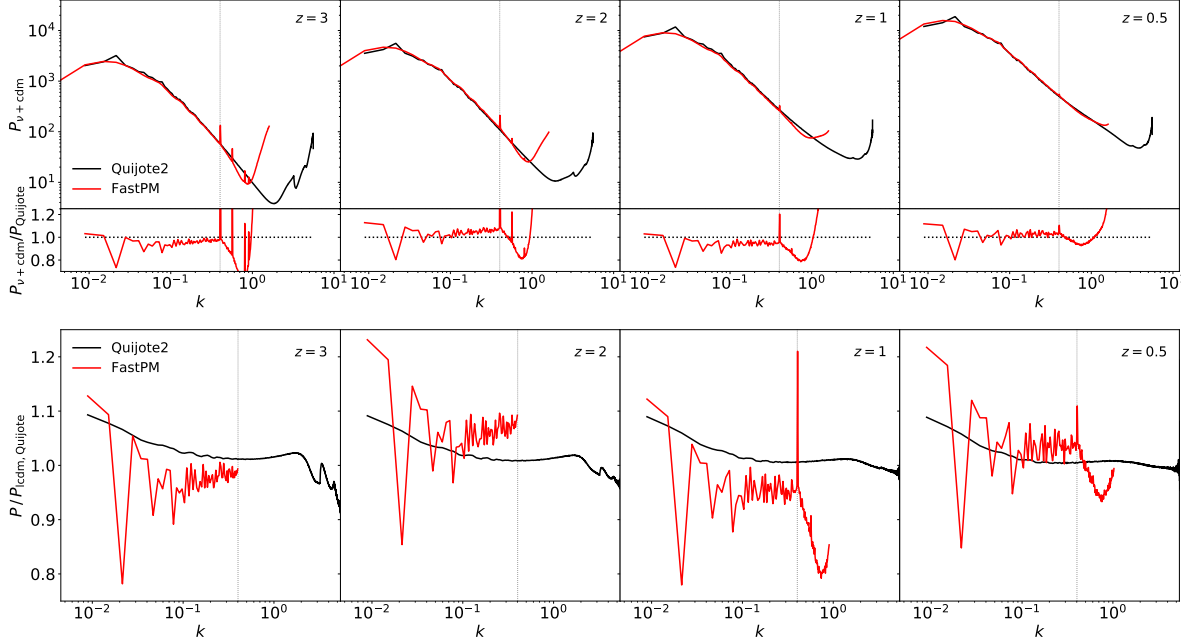


Figura 9.4: Como la figura 9.2, pero comparando la simulación FastPM (rojo) con una simulación Quijote similar a la de la tabla 9.1 pero con $M_\nu = 0.2$ eV. Las líneas punteadas verticales indican la mitad de la frecuencia de Nyquist dada la malla. BPM no presenta shot-noise.

como referencia la correspondiente simulación LCDM de Quijote, lo cual puede tener un efecto en el resultado, por lo cual requiere de más estudio. Al comparar nuestros resultados con los presentados en [91] (ver figura 3 de dicho artículo), se observa un comportamiento similar, pero, en nuestro caso no vemos el relieve (“bump”) que ellos observan. Sin embargo, como ellos discuten, esto puede deberse a la malla utilizada en nuestra simulación, pues ellos observan que este relieve crece cuando se usa una malla más fina.

9.3. Discusión

Nuestro estudio aunque prometedor, se encuentra en una etapa inicial. Falta realizar un barrido a escala gruesa en el espacio de parámetros de la simulación Quijote. Sin embargo, podemos concluir algunos puntos interesantes.

Como se anticipaba BPM desarrolla menos estructura que GADGET-3 (simulaciones Quijote) para el caso $M_\nu = 0.4$ eV particularmente a escalas menores a $k = 0.1$ Mpc/h. Nuestra estimación del MPS (bineo) para la simulación BPM requiere ajustes, lo cual explica oscilaciones a escalas grandes. La supresión debido a los neutrinos es mayor (entre 10 – 20%) en el caso BPM que en el caso GADGET para k entre 0.1 y 1 Mpc/h. Además, con BPM se puede visualizar la estructura de la componente de neutrinos (fig. 9.1), lo cual no es trivial de lograr para códigos puramente de N -cuerpos.

Las pruebas con los códigos MG-PICOLA y FastPM se realizaron considerando una

suma de masas de neutrinos $M_\nu = 0.2$ eV, lo cual implica una menor supresión que los casos Quijote y BPM. El MPS de ambas simulaciones coincide hasta casi $k = 0.1$ Mpc/h a $z = 0.5$, a partir de valores mayores de k , la potencia en el experimento FastPM es cada vez mayor, incluso con algunos picos en el MPS. Es posible que la malla de neutrinos imprima estructura en la distribución de CDM. A pesar de que la resolución es similar, los resultados tienen marcadas diferencias.

Los resultados obtenidos al comparar FastPM con la simulación Quijote2 (como Quijote en la tabla 9.1 pero con $M_\nu = 0.2$ eV), son similares a los de la figura 3 de [91], pero con algunas diferencias, las cuales pueden deberse a la resolución espacial utilizada, en comparación con Quijote2.

Capítulo 10

Discusión, trabajo futuro y conclusiones

Actualmente existe evidencia observacional sólida de una expansión cósmica acelerada, pero no hay evidencia contundente de que la causa es una nueva componente de energía. Por lo que el establecimiento de cualquiera de las explicaciones para la denominada energía oscura causaría un cambio dramático en nuestra comprensión del Universo en su conjunto. Es por esto que los sondeos de galaxias contemporáneos tienen como propósito mapear la estructura a gran escala del Universo con niveles de precisión altos (errores menores al 1 % en la escala de BAO), lo que permitirá utilizar a la red cósmica de galaxias como un detector de partículas, en ocasiones rivalizando con enfoques de laboratorio, por lo que en los próximos años se podría encontrar evidencia de la necesidad de una nueva física; por ejemplo, para neutrinos se podría determinar si son de Majorana o no, lo cual ayudaría a entender la bariogénesis y la simetría bariónica (sin embargo, hay que considerar las dificultades discutidas por [179], lo cual sugeriría una combinación de experimentos y cosmología). Incluir los efectos sutiles no-lineales en las predicciones teóricas requiere tradicionalmente de simulaciones muy detalladas, además, para lograr las metas científicas de estos sondeos se requiere del apoyo de simulaciones para cuantificar y eventualmente corregir diversos tipos de incertidumbres, interpretar sus resultados y orientar campañas de observación, entonces resulta importante que las simulaciones tengan la calidad de precisión de las observaciones. El poder realizar tal tipo de estudios requiere de grandes infraestructuras, pero además, como en el caso de los neutrinos, es aún un problema abierto la búsqueda de métodos de cálculo con la suficiente eficiencia computacional, y aunque actualmente existe una gran actividad en el campo, no se ha alcanzado una solución final. En esta línea se ubica esta tesis.

Las llamadas simulaciones de N -cuerpos se han constituido como un pilar importante de nuestra comprensión del Universo, pues su rango de aplicabilidad, en astronomía, va desde sistemas planetarios hasta cúmulos de estrellas y galaxias, así como en el estudio de cosmologías LCDM. A pesar de ello, en particular, no han sido tan exitosas para el estudio de cosmologías cuyo espectro de potencias de materia (MPS) presenta un corte/supresión generado por las velocidades térmicas grandes de sus partículas, como lo es el caso de WDM o una componente explícita de neutrinos, debido a que

el shot-noise, inherente en dichas simulaciones, puede dominar el MPS físico de estas cosmologías y generar estructuras espurias en la simulación, lo que puede sesgar la interpretación de las simulaciones.

Ahora bien, una de las posibles extensiones del modelo LCDM considera de forma explícita la presencia de materia oscura tibia (WDM) o incluso incorporar una componente caliente más no dominante (neutrino, radiación oscura, etc.) además de la componente fría (CDM). Siendo los neutrinos la segunda partícula masiva más abundante en el Universo, estos tienen un efecto importante en el crecimiento de estructura, debido a su flujo libre, lo cual, como vimos, deja huellas características en los observables cosmológicos. Por lo anterior, resulta importante considerarlos al hacer la interpretación de datos de los sondeos, así como en las simulaciones para tener una mejor comprensión del Universo, ya que de lo contrario puede erróneamente llevar a la necesidad de un nuevo tipo de energía o gravedad. Entonces, para poder estudiar cosmologías con neutrinos masivos con la precisión necesaria por las observaciones, se requiere que las simulaciones resuelvan de forma precisa la estructura a gran escala, es decir, que puedan resolver varias escalas de BAO, pero también deben resolver las escalas pequeñas, es decir, deben poder resolver las escalas de los trazadores observacionales (como galaxias). Para lograr esto, es necesario que la aproximación a usarse en las simulaciones con neutrinos masivos naturalmente pueda resolver escalas no-lineales, y que no esté dominada por el shot-noise mencionado anteriormente. Por tal motivo, en esta tesis desarrollamos un nuevo código, llamado Boltzmann-PM (BPM), el cual implementa la nueva técnica de momentos de la ecuación de Boltzmann (no-lineal) para incluir una componente explícita de neutrinos. Dicha técnica consiste en describir a los neutrinos simultáneamente como un medio continuo sobre una malla y como partículas prueba, en donde éstas últimas únicamente son usadas para cerrar la jerarquía de Boltzmann, y el medio continuo se evoluciona usando momentos de dicha jerarquía. Dada la descripción de medio continuo, esta nueva técnica suprime el shot-noise presente en las simulaciones de N -cuerpos sin necesidad de usar un gran número de partículas, permitiendo estudiar la formación de estructura de una forma más precisa y eficiente. Es importante mencionar que códigos con estas características no habían sido necesarios dada la precisión previa en las observaciones lo cual está cambiando con la nueva generación de censos de galaxias. Un aspecto a destacar en el formalismo que utiliza BPM, es que el término no homogéneo de la ecuación tipo Euler puede incluir el efecto de autointeracciones o decaimientos de forma natural sin necesidad de aproximaciones bruscas para las probabilidades de dichos procesos, en contraste con las técnicas de N -cuerpos. Esta avenida es interesante en el caso de los neutrinos, pues una autointeracción puede afectar las constricciones que la estructura cósmica a gran escala establezca para su masa, e incluso podría ser relevante para la tensión entre los valores de H_0 estimados a alto y bajo corrimiento al rojo o para constreñir la física de la leptogénesis y el origen de la masa de los neutrinos [180].

Como comentamos a lo largo de la tesis, el código BPM puede ser usado para distintas aplicaciones en astronomía, por ejemplo, dentro de Cosmología puede usarse para describir modelos de materia oscura (DM) con un corte en el espectro de potencias como WDM, Fuzzy DM y otros. Como una de estas aplicaciones, mostramos que BPM puede capturar muy eficientemente los modos de respuesta en halos oscuros y estelares

de galaxias, añadiendo posibles constricciones extras a las propiedades de los halos y de la materia oscura basadas en observaciones de halos estelares. Por lo que un estudio sistemático de este problema será abordable de manera práctica por BPM. Cabe mencionar que existen estudios en tal dirección con algunas restricciones en simetría (BFE) o un gran costo computacional, por ello BPM representa una propuesta novedosa y competitiva.

También, como resultado de las capacidades del código BPM, estudiamos modelos cosmológicos considerando WDM, mostrando que con un costo computacional considerablemente menor que con códigos únicamente con partículas como GADGET, se pueden explorar espacios de parámetros de modelos alternativos de materia oscura sin la necesidad de usar un gran número de partículas para suprimir el shot-noise. Otra ventaja que tiene BPM es que, al no tener shot-noise, no se requiere de un procedimiento artesanal para distinguir los subhalos “reales” de los espurios en la simulación, lo que lo hace más eficiente que la opción comúnmente usada de N -cuerpos.

En la tesis, también mostramos resultados preliminares para un trabajo futuro en el cual planeamos dar una comparación cuantitativa entre algunos de los métodos/técnicas utilizadas para simular cosmologías ν CDM. En particular, mostramos (de forma preliminar) a través del espectro de potencias de materia, que BPM puede simular sin efectos de shot-noise dichas cosmologías a diferencia de métodos basados en partículas (simulaciones Quijote), como era de esperarse. Este es un resultado prometedor, sin embargo, requiere de más investigación en otros observables (función de masa de halos, función de correlación, voids, etc.).

Como otro resultado de la capacidad de BPM, encontramos que éste preserva “imprints” del MPS inicial, intrínsecos en modelos alternativos de materia oscura, a lo largo de la simulación a diferencia de un código PM u otras técnicas de N -cuerpos, pues éstos destruyen rápidamente dichos “imprints”. Por lo que puede ser utilizado para estudiar modelos de materia oscura ultraligera.

A pesar de que el código BPM aún no está paralelizado, hemos mostrado que puede ser competitivo con los códigos contemporáneos, debido a que su descripción como medio continuo puede suprimir el shot-noise inherente en métodos puramente de N -cuerpos, permitiendo hacer simulaciones precisas para escalas lineales y cuasi-lineales, importante para la nueva generación de datos observacionales. Por lo tanto, BPM tiene potencialmente un valor para explorar espacios de parámetros de cosmologías y la interpretación de observaciones con una combinación de eficiencia y precisión. Además, tiene gran versatilidad para poder usarse en distintos problemas astrofísicos y no sólo en cosmología.

Cabe mencionar que, como todo método/técnica numérica, BPM también tiene limitantes, las cuales son: 1)tiene un resolución fija dada por la malla a usarse en la simulación, 2)las simulaciones pueden llegar a tener difusividad y 3)la estrategia de cerradura para la jerarquía de momentos. Sin embargo, en la tesis mostramos que las dos últimas limitantes mencionadas no representaban un problema para las simulaciones realizadas, pues mostramos que el código es capaz de mantener la estabilidad de sistemas como un halo aislado y también del espectro de potencias a gran escala, lo que da certeza de que no afectan de forma drástica la abundancia de halos en las simulaciones realizadas. En cuanto a la primera limitante mencionada, mostramos que el

uso combinado de BPM (hasta cierto z_c) con códigos de N -cuerpos como GADGET-4 (de z_c hasta $z = 0$) representa una buena alternativa (una vez estudiados la precisión y los posibles sesgos) para alcanzar mayor resolución espacial en las simulaciones, y con ello no requerir una malla demasiado fina para alcanzar una mayor resolución.

Comparado con la técnica puramente PM, BPM involucra resolver más ecuaciones (los momentos de Boltzmann) y estimar momentos de la función de distribución de partículas de orden mayor (como la velocidad de grupo (“bulk velocity” en inglés) o la dispersión de velocidades), por lo cual uno podría preguntarse si vale la pena usar BPM en lugar de un simple PM. Para responder esto, es importante mencionar que la única fuente posible de shot-noise durante una simulación con BPM es a la hora de estimar la “bulk velocity” o la dispersión de velocidades sobre la malla a partir de las partículas prueba, de igual forma en que se genera shot-noise al estimar el campo de densidad con las posiciones de las partículas en un PM. Sin embargo, una diferencia importante entre el shot-noise producido por las velocidades de las partículas y el producido por las posiciones de éstas, es que, para una simulación cosmológica, el ruido de velocidades disminuye sobre el tiempo conforme el universo se expande, en contraste, el ruido en el campo de densidad estimado con las partículas crece con el tiempo. Esto ilustra una de las ventajas de BPM sobre el PM. Además, dado que en el BPM el campo de densidad se sigue a través de una ecuación tipo continuidad, y se usan técnicas hidrodinámicas de advección, BPM es capaz de capturar y mantener mejor perturbaciones en el campo de densidad (ver por ejemplo, capítulo 7 o la figura 6.17) a diferencia del PM, donde el campo de densidad es construido a través de las posiciones de las partículas.

Ahora bien, dado que los códigos competitivos contemporáneos usan más de una técnica numérica para aumentar su eficiencia computacional, resulta conveniente en un futuro combinar BPM con otra técnica para aumentar su resolución espacial y eficiencia. Motivados por esto, desarrollamos el código de N -cuerpos $\text{NP}_{\text{splitt}}$ que usa la técnica Partícula-Partícula (PP, también conocida como suma directa) para el cálculo de las fuerzas. Este código tiene implementados los integradores, comúnmente usados, Leapfrog (paso de tiempo constante), Hermite, AKDK (*Kick-Drift-Kick* adaptivo) y el nuevo integrador de división hamiltoniana jerárquica (HHS por sus siglas en inglés). Además, el código está adaptado para arquitecturas computacionales contemporáneas (Serial X86, POWER, ARM; Paralela: OMP, MPI, CUDA, CUDA-MPI), incluyendo una opción multi-GPU. En la tesis presentamos su implementación y rendimiento para los diferentes integradores.

El gran rango dinámico en simulaciones cosmológicas requiere el uso de integradores con pasos de tiempo adaptivos pero que logren seguir la evolución del sistema de forma precisa y eficiente. Es por ello que analizamos y caracterizamos la pertinencia del integrador HHS para el caso de simulaciones no-colisionales, y comparamos cuantitativamente los resultados con el integrador AKDK (usado típicamente en simulaciones cosmológicas). Como resultado encontramos que HHS es aproximadamente reversible, por lo que es más estable en las integrales de movimiento permitiendo pasos mayores con mejores rendimientos que AKDK. De esta forma, HHS es una opción atractiva para usar en simulaciones cosmológicas a un menor costo computacional que AKDK, garantizando conservación de momento lineal y angular a precisión de la máquina. Cabe mencionar que aunque nuestra implementación tendría retos para ser incluida en

todo tipo de códigos, en particular en el método P³M (PP+PM) se vería beneficiado, particularmente con el uso de GPU's.

10.1. Futuro del código

Actualmente en la cosmología existe un importante interés por métodos perturbativos (como HALOFIT) y recientemente por métodos basados en entrenar algoritmos de clasificación (redes neuronales, árboles aleatorios, etc.) para reproducir el resultado de las simulaciones cosmológicas y poder generar varias a costo reducido. Algo que es importante considerar es que en ambos casos mencionados se requiere sintonizar el procedimiento utilizando simulaciones que incluyan la física completa. En el caso de las estrategias de Machine Learning el entrenamiento requiere de snapshots, y entre mayor sea el conjunto de información, mejor será el entrenamiento, incrementando el costo computacional. Es aún un reto abierto generar simulaciones con el mismo grado de detalle y de precisión, aunque si se han logrado casos en el que estadísticos mantienen el nivel de precisión, cabe mencionar que la generalidad de varios de tales resultados esta abierta y no es evidente el cambio del resultado utilizando algoritmos distintos. Aún más complicado es el caso en que la física del problema no se incluya en las simulaciones de entrenamiento. Un área interesante considera que el entrenamiento lleve a mapeos de las condiciones iniciales a snapshots evolucionados, de la misma manera que los integradores discutidos en esta tesis. Como mencionamos, todos estos casos requieren de haberse explorado la física del problema y en tal exploración, nuestra aportación también puede ser utilizada.

Como mencioné previamente, actualmente los códigos competitivos como HACC, ABACUS o GADGET son híbridos, es decir, combinan más de una técnica numérica para el cálculo de las fuerzas con el fin de ser computacionalmente eficientes. Por ejemplo, GADGET usa la técnica Tree-PM (un código de árbol más un PM) o HACC es un P³M (técnica partícula-partícula más un PM y dependiendo la arquitectura un Tree-PM). Por lo anterior, planeamos acoplar el código Boltzmann-PM (BPM) con el código de suma directa N_{splitt} , en donde el primero se usaría para la parte de menor resolución espacial y el segundo para resolver la parte de alta resolución en la simulación, y con ello alcanzar mayor resolución espacial, a un costo computacional menor y con mejor precisión.

El hecho de que N_{splitt} esté paralelizado sobre múltiples GPUs y que tiene implementado el integrador HHS, haría que al acoplarlo con el código Boltzmann-PM, se obtendría una herramienta bastante competitiva, con los códigos en el estado del arte, que puede simular de forma precisa modelos cosmológicos con una supresión en el espectro de potencias de materia, además de que permitiría barrer espacios de parámetros cosmológicos y de esta forma ayudar en la interpretación de las nuevas observaciones.

Otra forma alternativa de hacer adaptivo al código BPM y con ello mejorar su alcance en resolución espacial, es utilizar algoritmos para transformada rápida de Fourier en mallas inhomogéneas (NFFT). Exploraremos tal opción en el futuro.

También, para aumentar el nivel de competitividad del código BPM con los otros códigos en el estado del arte, buscaremos paralelizar a éste. La parte de FFT es casi

trivial de realizar, y es la parte de la solución de los momentos de Boltzmann la que requiere más trabajo (para poder identificar las partes del código que consumen más tiempo, y por ende las que se paralelizaran, se requiere de perfilar el código). Análisis preliminares sugieren que debido a que la parte de Boltzmann depende de las celdas y sus vecinos relativamente locales, bastaría con dividir la malla y tratar cada dominio como un problema secuencial y después se unirían los resultados.

Así mismo, la versión multi-especie del código BPM, por ahora tiene dos opciones. La primera es utilizar el BPM para ambas especies (le llamamos BPM-BPM); la segunda opción, es utilizar el BPM para una especie y el PM para la otra especie (le llamamos BPM-PM). Sin embargo, dado que las especies de partículas únicamente están acopladas gravitacionalmente, entonces puede usarse el BPM para una especie y otra técnica/código para la otra especie, por ejemplo, GADGET o HACC. Por lo anterior, en un futuro buscaremos tener este tipo de variantes/extensiones para la versión multi-especie.

Finalmente, planeamos estudiar más a fondo la difusividad que puede inducir el usar un método numérico TVD (del inglés total variational diminishing) al resolver las ecuaciones de momentos de la ecuación de Boltzmann en el código BPM, y exploraremos la posibilidad de ir a momentos de Boltzmann de orden superior para disminuir este efecto. O bien caracterizar dicha difusividad. En esta misma línea, es de interés explorar la posibilidad de que en lugar de utilizar momentos de Boltzmann resolver los 6 problemas de advección de manera similar a lo que ha realizado [115, 116], utilizando además métodos de orden superior. Es la solución más precisa, sin embargo, las primeras aplicaciones prácticas se han realizado en la supercomputadora Fugaku, con un código que tiene estructura muy similar a BPM. Esta exploración posiblemente deba esperar a la siguiente generación de computadoras exascale.

Apéndice A

Deducción de las ecuaciones usadas en Boltzmann-PM

A.1. Ecuación de Boltzmann no-colisional en la norma newtoniana

La ecuación de Boltzmann no-colisional está dada por:

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial x^i} \frac{dx^i}{dt} + \frac{\partial f}{\partial p^i} \frac{dp^i}{dt} = 0 \quad (\text{A.1})$$

donde $f(\mathbf{x}, \mathbf{p})$ es la función de distribución del espacio fase de la especie dinámica. Adoptando la métrica Newtoniana [138], la cual es usada para el modo escalar de las perturbaciones métricas. Las perturbaciones están caracterizadas por dos potenciales escalares Φ y Ψ . La métrica está dada por:

$$ds^2 = -(1 + 2\Psi) dt^2 + a^2(t) (1 - 2\Phi) \delta_{ij} dx^i dx^j$$

donde δ_{ij} es la delta de Kronecker. A primer orden en los potenciales [139]:

$$\begin{aligned} \frac{dx^i}{dt} &= \frac{p^i}{aE} (1 - \Phi + \Psi) \\ \frac{dp^i}{dt} &= -p^i \frac{\partial \Phi}{\partial t} - p^i \frac{\dot{a}}{a} - \frac{E}{a} \frac{\partial \Psi}{\partial x^i} \end{aligned}$$

donde $E = \sqrt{p^i p^i + m^2}$ y a es el factor de escala. \dot{a} significa una derivada respecto al tiempo, y no respecto al tiempo conforme. Substituyendo estas expresiones en A.1 obtenemos:

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial x^i} \left[\frac{p^i}{aE} (1 - \Phi + \Psi) \right] + \frac{\partial f}{\partial p^i} \left[-p^i \frac{\partial \Phi}{\partial t} - p^i \frac{\dot{a}}{a} - \frac{E}{a} \frac{\partial \Psi}{\partial x^i} \right] = 0$$

A.2. Ecuación cosmológica tipo Continuidad

Para obtener la ecuación cosmológica tipo continuidad (momento cero de Boltzmann) multiplicamos por E la ecuación A.1 e integramos sobre el momento:

$$\begin{aligned} \int \left(E \frac{\partial f}{\partial t} \right) d^3p + \int \left(\frac{\partial f}{\partial x^i} \left[\frac{p^i}{a} (1 - \Phi + \Psi) \right] \right) d^3p \\ - \int \left(E p^i \frac{\partial f}{\partial p^i} \left[\frac{\partial \Phi}{\partial t} + \frac{\dot{a}}{a} \right] \right) d^3p - \int \left(\frac{E^2}{a} \frac{\partial f}{\partial p^i} \frac{\partial \Psi}{\partial x^i} \right) d^3p = 0 \end{aligned} \quad (\text{A.2})$$

Simplificando la ecuación,

- *Primer término:*

$$\int \left(E \frac{\partial f}{\partial t} \right) d^3p = \frac{\partial}{\partial t} \left[\int E f d^3p \right] = \frac{\partial \rho}{\partial t}$$

donde definimos la densidad del espacio fase real en términos de la densidad del espacio fase, como: $\rho(\mathbf{x}) = \int E f(\mathbf{x}, \mathbf{p}) d^3p$. La primera igualdad es debida al hecho de que el rango de integración no depende de t y entonces $\frac{\partial}{\partial t}$ puede salir de la integral.

- *Segundo término:*

$$\begin{aligned} \int \left(\frac{\partial f}{\partial x^i} \left[\frac{p^i}{a} (1 - \Phi + \Psi) \right] \right) d^3p &= \frac{(1 - \Phi + \Psi)}{a} \int \frac{\partial}{\partial x^i} (p^i f) d^3p \\ &= \frac{(1 - \Phi + \Psi)}{a} \frac{\partial}{\partial x^i} \left[\int p^i f d^3p \right] \\ &= \frac{(1 + 2\Psi)}{a} \frac{\partial}{\partial x^i} (\bar{\rho} \Pi^i) \end{aligned}$$

donde definimos $\bar{\rho} \Pi^i = \int p^i f(\mathbf{x}, \mathbf{p}) d^3p$, y asumimos que $\Psi = -\Phi$, lo cual es válido para épocas durante el dominio de materia ($z \lesssim 300$) cuando el esfuerzo anisotrópico general es pequeño.

- *Tercer término:*

$$\begin{aligned} - \int \left(E p^i \frac{\partial f}{\partial p^i} \left[\frac{\partial \Phi}{\partial t} + \frac{\dot{a}}{a} \right] \right) d^3p &= - \left(\frac{\partial \Phi}{\partial t} + H \right) \int E p^i \frac{\partial f}{\partial p^i} d^3p \\ &= - \left(\frac{\partial \Phi}{\partial t} + H \right) \left\{ \int \frac{\partial}{\partial p^i} (E p^i f) d^3p - \int f \frac{\partial}{\partial p^i} (E p^i) d^3p \right\} \\ &= - \left(\frac{\partial \Phi}{\partial t} + H \right) \left\{ \int E p^i f d\vec{S} - \int f E \frac{\partial p^i}{\partial p^i} d^3p - \int f p^i \frac{\partial E}{\partial p^i} d^3p \right\} \\ &= \left(\frac{\partial \Phi}{\partial t} + H \right) \left\{ 3 \int E f d^3p + \int \frac{p^i p^i}{E} f d^3p \right\} \\ &= \left(\frac{\partial \Phi}{\partial t} + H \right) \{ 3\rho + 3\rho W \} \\ &= 3 \left(\frac{\partial \Phi}{\partial t} + H \right) \rho (1 + W) \end{aligned}$$

donde $H = \dot{a}/a$ es el parámetro de Hubble. Para la segunda igualdad usamos integración por partes. En la tercera igualdad usamos el teorema de la divergencia para la primera integral (del paso previo) y la regla de la cadena para la otra integral (en el paso previo). Para la cuarta igualdad, usamos que $\int E p^i f d\vec{S} = 0$. Para la quinta igualdad usamos la definición de ρ y definimos $\rho W = \int \frac{p^i p^i}{3E} f(\mathbf{x}, \mathbf{p}) d^3p$.

■ *Cuarto término:*

$$\begin{aligned}
-\int \left(\frac{E^2}{a} \frac{\partial f}{\partial p^i} \frac{\partial \Psi}{\partial x^i} \right) d^3p &= -\frac{1}{a} \frac{\partial \Psi}{\partial x^i} \int E^2 \frac{\partial f}{\partial p^i} d^3p \\
&= -\frac{1}{a} \frac{\partial \Psi}{\partial x^i} \left\{ \int \frac{\partial}{\partial p^i} (E^2 f) d^3p - \int f \frac{\partial E^2}{\partial p^i} d^3p \right\} \\
&= -\frac{1}{a} \frac{\partial \Psi}{\partial x^i} \left\{ \int E^2 f d\vec{S} - 2 \int p^i f d^3p \right\} \\
&= \frac{2}{a} \frac{\partial \Psi}{\partial x^i} \bar{\rho} \Pi^i
\end{aligned}$$

Para la segunda igualdad usamos integración por partes. En la tercera igualdad usamos el teorema de la divergencia para la primera integral (en el paso previo).

Luego, sustituyendo los términos anteriores en la ecuación A.2, tenemos:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{(1+2\Psi)}{a} \frac{\partial}{\partial x^i} (\bar{\rho} \Pi^i) + 3 \left(\frac{\partial \Phi}{\partial t} + H \right) \rho (1+W) + \frac{2}{a} \frac{\partial \Psi}{\partial x^i} \bar{\rho} \Pi^i &= 0 \\
\implies \frac{\partial \rho}{\partial t} + \frac{1}{a} \frac{\partial [(1+2\Psi) \bar{\rho} \Pi^i]}{\partial x^i} + 3 (\dot{\Phi} + H) \rho (1+W) &= 0 \\
\implies \frac{\partial [\bar{\rho} (1+\delta)]}{\partial t} + \frac{\bar{\rho}}{a} \frac{\partial [(1+2\Psi) \Pi^i]}{\partial x^i} + 3 (\dot{\Phi} + H) \bar{\rho} (1+\delta) (1+W) &= 0 \quad (\text{A.3})
\end{aligned}$$

en donde, usamos el contraste de densidad $\delta = (\rho - \bar{\rho})/\bar{\rho}$ en el último paso. Expandiendo el primer término de A.3 tenemos:

$$\begin{aligned}
\frac{\partial [\bar{\rho} (1+\delta)]}{\partial t} &= \bar{\rho} \frac{\partial \delta}{\partial t} + (1+\delta) \frac{\partial}{\partial t} (a^{-3(1+\omega)}) \\
&= \bar{\rho} \frac{\partial \delta}{\partial t} - (1+\delta) 3(1+\omega) H \bar{\rho}
\end{aligned}$$

donde usamos la regla de la cadena y que $\bar{\rho} = a^{-3(1+\omega)}$, con $\omega = \bar{P}/\bar{\rho}$ es la ecuación de estado del fondo. Substituyendo esto en la ecuación A.3 tenemos:

$$\begin{aligned}
\bar{\rho} \frac{\partial \delta}{\partial t} - 3(1+\omega) H \bar{\rho} (1+\delta) + \frac{\bar{\rho}}{a} \frac{\partial [(1+2\Psi) \Pi^i]}{\partial x^i} + 3 (\dot{\Phi} + H) \bar{\rho} (1+\delta) (1+W) &= 0 \\
\implies \frac{\partial \delta}{\partial t} + \frac{1}{a} \frac{\partial [(1+2\Psi) \Pi^i]}{\partial x^i} + 3 \dot{\Phi} (1+\delta) (1+W) + 3H (1+\delta) (W-\omega) &= 0 \quad (\text{A.4})
\end{aligned}$$

Esta última ecuación es la ecuación cosmológica tipo continuidad. Podemos hacer algunas simplificaciones en ésta ecuación. En el régimen de dominación de materia, $\dot{\Phi}$ es pequeña comparada con la derivada espacial de Φ , y podemos despreciar el tercer término sin perder precisión. Además, si las partículas son relativistas, el último término de A.4 es pequeño o nulo porque las velocidades de sonido local y del fondo son la misma, y $(W - \omega)$ se anula. Cuando las partículas son no-relativistas, W y ω son individualmente pequeños ($\sim 10^{-6}$) e incluso en regímenes no-lineales el último término permanece mucho más pequeño que el segundo término. Por lo anterior, podemos despreciar los últimos dos términos de A.4, obteniendo la ecuación 6.4:

$$\dot{\delta} + \frac{1}{a} \frac{\partial [(1 + 2\Psi) \Pi^i]}{\partial x^i} = 0$$

A.3. Ecuación cosmológica tipo Euler

Para obtener la ecuación cosmológica tipo Euler (primer momento de la ecuación de Boltzmann) multiplicamos por p^i la ecuación A.1 e integramos sobre el momento:

$$\begin{aligned} \int \left(p^i \frac{\partial f}{\partial t} \right) d^3p + \int \left(p^i \frac{\partial f}{\partial x^j} \left[\frac{p^j}{aE} (1 - \Phi + \Psi) \right] \right) d^3p \\ - \int \left[p^i p^j \frac{\partial f}{\partial p^j} (\dot{\Phi} + H) \right] d^3p - \int \left(p^i \frac{E}{a} \frac{\partial f}{\partial p^j} \frac{\partial \Psi}{\partial x^j} \right) d^3p = 0 \end{aligned} \quad (\text{A.5})$$

Simplificando la ecuación,

- *Primer término:*

$$\int \left(p^i \frac{\partial f}{\partial t} \right) d^3p = \frac{\partial}{\partial t} \left[\int p^i f d^3p \right] = \frac{\partial}{\partial t} (\bar{\rho} \Pi^i) = \bar{\rho} \frac{\partial \Pi^i}{\partial t} - \Pi^i 3(1 + \omega) H \bar{\rho}$$

La primera igualdad es debido a que el rango de integración no depende de t , por lo tanto $\frac{\partial}{\partial t}$ puede salir de la integral.

- *Segundo término:*

$$\begin{aligned} \int \left(p^i \frac{\partial f}{\partial x^j} \left[\frac{p^j}{aE} (1 - \Phi + \Psi) \right] \right) d^3p &= \frac{(1 + 2\Psi)}{a} \int \frac{p^i p^j}{E} \frac{\partial f}{\partial x^j} d^3p \\ &= \frac{(1 + 2\Psi)}{a} \frac{\partial}{\partial x^j} \int \frac{p^i p^j}{E} f d^3p \\ &= \frac{(1 + 2\Psi)}{a} \frac{\partial (\rho W^{ij})}{\partial x^j} \end{aligned}$$

donde en la última igualdad definimos $\rho W^{ij} = \int \frac{p^i p^j}{E} f d^3p$.

- *Tercer término:*

$$\begin{aligned} - \int \left(p^i p^j \frac{\partial f}{\partial p^j} [\dot{\Phi} + H] \right) d^3p &= - (\dot{\Phi} + H) \left\{ \int \frac{\partial}{\partial p^j} (p^i p^j f) d^3p - \int f \frac{\partial (p^i p^j)}{\partial p^j} d^3p \right\} \\ &= - (\dot{\Phi} + H) \left\{ \int p^i p^j f d\vec{S} - 4 \int p^i f d^3p \right\} \\ &= (\dot{\Phi} + H) 4\bar{\rho} \Pi^i \end{aligned}$$

en donde en la primer igualdad usamos integración por partes y en la segunda igualdad usamos el teorema de la divergencia.

■ *Cuarto término:*

$$\begin{aligned}
-\int \left(p^i \frac{E}{a} \frac{\partial f}{\partial p^j} \frac{\partial \Psi}{\partial x^j} \right) d^3p &= -\frac{1}{a} \frac{\partial \Psi}{\partial x^j} \left\{ \int \frac{\partial}{\partial p^j} (E p^i f) d^3p - \int f \frac{\partial}{\partial p^j} (E p^i) d^3p \right\} \\
&= -\frac{1}{a} \frac{\partial \Psi}{\partial x^j} \left\{ \int E p^i f d\vec{S} - \int f \left(E \delta^{ij} + \frac{p^i p^j}{E} \right) d^3p \right\} \\
&= \frac{1}{a} \frac{\partial \Psi}{\partial x^j} (\rho \delta^{ij} + \rho W^{ij}) \\
&= \frac{\rho}{a} \frac{\partial \Psi}{\partial x^i} + \frac{\rho W^{ij}}{a} \frac{\partial \Psi}{\partial x^j}
\end{aligned}$$

donde δ^{ij} es la delta de Kronecker. En la primera igualdad usamos integración por partes y en la segunda el teorema de la divergencia.

Ahora, sustituyendo los términos anteriores en la ecuación A.5 tenemos:

$$\begin{aligned}
\bar{\rho} \frac{\partial \Pi^i}{\partial t} - \Pi^i 3(1 + \omega) H \bar{\rho} + \frac{(1 + 2\Psi)}{a} \frac{\partial (\rho W^{ij})}{\partial x^j} + (\dot{\Phi} + H) 4\bar{\rho} \Pi^i + \frac{\rho}{a} \frac{\partial \Psi}{\partial x^i} + \frac{\rho W^{ij}}{a} \frac{\partial \Psi}{\partial x^j} &= 0 \\
\implies \frac{\partial \Pi^i}{\partial t} + (1 - 3\omega) \Pi^i H + \frac{1}{\bar{\rho} a} \frac{\partial (\rho W^{ij})}{\partial x^j} + \frac{\rho}{\bar{\rho} a} \frac{\partial \Psi}{\partial x^i} + 4\Pi^i \dot{\Phi} + \frac{2\Psi}{\bar{\rho} a} \frac{\partial (\rho W^{ij})}{\partial x^j} + \frac{\rho W^{ij}}{\bar{\rho} a} \frac{\partial \Psi}{\partial x^j} &= 0
\end{aligned} \tag{A.6}$$

Ahora, como en el caso de la ecuación de continuidad, podemos despreciar el quinto término sin perder precisión debido a que, en regímenes de dominación de materia, $\dot{\Phi}$ es pequeño comparado con las derivadas espaciales de Φ . También podemos despreciar los últimos dos términos. Introduciendo el contraste de densidad $\delta = (\rho - \bar{\rho}) / \bar{\rho}$ en la ecuación A.6, después de las simplificaciones, obtenemos la ecuación tipo Euler:

$$\dot{\Pi}^i + (1 - 3\omega) H \Pi^i + \frac{1}{a} \frac{\partial [(1 + \delta) W^{ij}]}{\partial x^j} = -\frac{1 + \delta}{a} \frac{\partial \Psi}{\partial x^i}$$

Apéndice B

Técnica Partícula-Malla

Partícula-Malla (Particle-Mesh en inglés, PM, o también conocida como Particle-In-Cell) es una técnica computacional para determinar las fuerzas en un sistema de partículas que aprovecha la formulación de fuerza en un punto y una ecuación de campo para el potencial. Esta técnica consiste en convertir el sistema de partículas a simular en una malla de densidad, donde el potencial generado por ésta se resuelve a través de la ecuación de Poisson, y con ello las respectivas fuerzas son aplicadas a cada partícula.

El esquema principal del método PM consta de los siguientes pasos (ver figura B.1 para un esquema pictórico del método):

1. Construir el campo de densidad sobre la malla a partir de las posiciones de las partículas del sistema a simular. A esto se le conoce como asignación de densidad.
2. Resolver la ecuación de Poisson sobre la malla usando dos Transformadas de Fourier Rápidas (Fast Fourier Transforms en inglés, FFTs) en 3D.
3. Calcular la fuerza sobre cada partícula basándose en su posición en la malla.
4. Evolucionar las posiciones y velocidades de las partículas del sistema.
5. Avanzar en el tiempo.
6. Volver al paso 1.

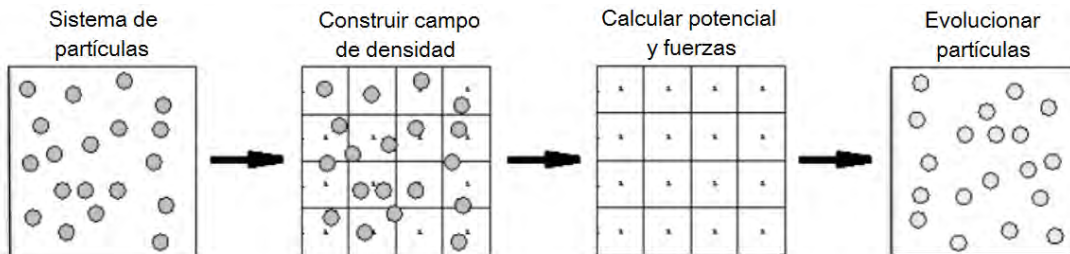


Figura B.1: Esquema pictórico del algoritmo de Particle-Mesh (PM).

En la técnica PM el número de operaciones por partícula y por paso de tiempo es pequeño a comparación de otras técnicas, como Partícula-Partícula (también conocida como suma directa) o TREE, haciendo que sea más eficiente (menor costo computacional) que éstas. Por lo anterior, puede usar un número mayor de partículas que las otras técnicas. En esta técnica el tiempo de ejecución, t_E escala como:

$$t_E \propto O(N_p) + O(N_c \ln N_c)$$

donde N_p es el número de partículas del sistema y N_c es el número de celdas en la malla de la simulación.

B.1. Asignación de Densidad

La técnica Partícula-Malla asume que las partículas tienen cierto tamaño, masa, forma y densidad interna. Esto determina el esquema de interpolación que se usará para construir el campo de densidad sobre la malla. Con tal propósito se introduce una forma de partícula [140]. Si $S(x)$ (función de forma) es la densidad a la distancia x de la partícula y Δx es el tamaño de la celda, entonces la densidad a la distancia $\mathbf{r} = (x, y, z)$ es el producto $S(\mathbf{r}) = S(x)S(y)S(z)$. Luego, la fracción de masa de una partícula p con función de forma $S(\mathbf{r}_p)$ asignada a la celda ijk con centro $\mathbf{r}_{ijk} = (x_{ijk}, y_{ijk}, z_{ijk})$ es dicha función de forma promediada sobre la celda:

$$W(x_p - x_{ijk}) = \int_{x_{i,j,k}-\Delta x/2}^{x_{i,j,k}+\Delta x/2} S(x_p - x') dx'$$

$$W(\mathbf{r}_p - \mathbf{r}_{ijk}) = W(x_p - x_{ijk}) W(y_p - y_{ijk}) W(z_p - z_{ijk})$$

Finalmente, la densidad en la celda ijk está dada por:

$$\rho_{ijk} = \sum_{p=1}^{N_p} m_p W(\mathbf{r}_p - \mathbf{r}_{ijk})$$

Las funciones de forma $S(x)$ más comunes en la literatura (y sus funciones de peso $W(x)$) son:

- “Punto de la malla más cercano” (NGP, por sus siglas en inglés): se asume que las partículas son un punto y toda la masa de la partícula es asignada a la única celda que la contiene. En este caso $S(x)$ y $W(x)$ están dados por,

$$S(x) = \frac{1}{\Delta x} \delta\left(\frac{x}{\Delta x}\right) = \delta(x)$$

$$W(x) = \begin{cases} 1, & |x| \leq \Delta x/2 \\ 0, & \text{en otro caso} \end{cases}$$

donde $\delta(x)$ es la función delta de Dirac.

- “Nube en la celda” (CIC, por sus siglas en inglés) [143]: se asume que las partículas son cubos (en 3D) de densidad uniforme y de tamaño de una celda. Para este caso $S(x)$ y $W(x)$ están dados por,

$$S(x) = \frac{1}{\Delta x} \begin{cases} 1, & |x| \leq \Delta x/2 \\ 0, & \text{en otro caso} \end{cases}$$

$$W(x) = \begin{cases} 1 - \frac{|x|}{\Delta x}, & |x| \leq \Delta x \\ 0, & \text{en otro caso} \end{cases}$$

- “Nube de forma triangular” (TSC, por sus siglas en inglés): en este caso $S(x)$ y $W(x)$ están dados por,

$$S(x) = \frac{1}{\Delta x} \begin{cases} 1 - \frac{|x|}{\Delta x}, & |x| \leq \Delta x \\ 0, & \text{en otro caso} \end{cases}$$

$$W(x) = \begin{cases} \frac{3}{4} - \left(\frac{|x|}{\Delta x}\right)^2, & |x| \leq \frac{\Delta x}{2} \\ \frac{1}{2} \left(\frac{3}{2} - \frac{|x|}{\Delta x}\right)^2, & \frac{\Delta x}{2} < |x| \leq \frac{3\Delta x}{2} \\ 0, & \text{en otro caso} \end{cases}$$

B.2. Cálculo del Potencial

Una vez construido el campo de densidad, ρ , sobre la malla, podemos calcular el potencial gravitacional, ϕ , de éste a través de la ecuación de Poisson:

$$\nabla^2 \phi = 4\pi G \rho(\mathbf{x})$$

Para ello, primero se aplica una transformada de Fourier rápida (FFT, por sus siglas en inglés) en 3D al campo de densidad. Esto da como resultado el campo de densidad en el espacio de Fourier, $\tilde{\rho}_{\mathbf{k}}$, donde \mathbf{k} es un vector con componentes enteras en el rango $0, 1, \dots, N_g - 1$. Luego, en el espacio de Fourier la ecuación de Poisson se simplifica enormemente, y el potencial en el espacio de Fourier, $\tilde{\phi}$, está dado por el producto de $\tilde{\rho}_{\mathbf{k}}$ por una función de Green, $\mathcal{G}(\mathbf{k})$, es decir:

$$\tilde{\phi} = 4\pi G \tilde{\rho}(\mathbf{k}) \mathcal{G}(\mathbf{k})$$

Por lo que, para obtener el potencial gravitacional en el espacio “real” (es decir, ϕ) basta con aplicar una FFT inversa a $\tilde{\phi}$.

Ahora bien, existen diversas funciones de Green que pueden usarse. Para la presente tesis usamos las funciones de 1er. (\mathcal{G}_0) y 2o. orden (\mathcal{G}_1) con condiciones de frontera periódicas (p. ej. [145]): respectivamente:

$$\mathcal{G}_0(\mathbf{k}) = -\frac{1}{k_x^2 + k_y^2 + k_z^2}$$

$$\mathcal{G}_1(\mathbf{k}) = -\left[\sin^2\left(\frac{k_x}{2}\right) + \sin^2\left(\frac{k_y}{2}\right) + \sin^2\left(\frac{k_z}{2}\right) \right]^{-1}$$

donde,

$$k_x = 2\pi l/L, \quad k_y = 2\pi m/L, \quad k_z = 2\pi n/L, \quad \text{para la componente } (l, m, n)$$

y L es el tamaño de la caja a simular.

Finalmente, actualmente existen diferentes herramientas en la literatura que permiten calcular FFT's. En particular, para esta tesis usamos la biblioteca FFTW3 [144].

B.3. Cálculo de Fuerzas

Una vez calculado el potencial gravitacional sobre la malla, se calcula la fuerza (sobre la malla), \mathbf{F} , usando la relación:

$$\mathbf{F}(\mathbf{x}) = -m \nabla \phi(\mathbf{x})$$

Para ello se requiere calcular la derivada del potencial. Existen diferentes métodos numéricos para aproximar derivadas, y entre los más populares están las diferencias finitas centradas. Para esta tesis usamos las de 2o y 4o orden (también conocido como el “operador de cuatro-puntos”), que en 1D están dadas por, respectivamente:

$$F_{2,x}(\mathbf{x}_i) = -m \frac{\phi(\mathbf{x}_{i+1}) - \phi(\mathbf{x}_{i-1})}{2\Delta x}$$

$$F_{4,x}(\mathbf{x}_i) = -\frac{m}{12\Delta x} [\phi(\mathbf{x}_{i-2}) - 8\phi(\mathbf{x}_{i-1}) + 8\phi(\mathbf{x}_{i+1}) - \phi(\mathbf{x}_{i+2})]$$

Posteriormente, se realiza una interpolación de la fuerza sobre la malla, \mathbf{F} para obtener la fuerza de cada partícula, \mathbf{f} . Es importante mantener coherencia entre los esquemas de interpolación porque esto garantiza la ausencia de fuerzas propias artificiales y la conservación del momento [140]. Por ello, debe usarse el mismo esquema de interpolación que se usó en la etapa de asignación de densidad.

Cabe mencionar que otra posibilidad para el cálculo de fuerzas es calcular ésta en el espacio de Fourier ($\tilde{\mathbf{F}}$) usando $\tilde{\phi}$, pues las derivadas pueden realizarse en éste, por ejemplo siguiendo el esquema de derivada de Lanczos, con el fin de minimizar la propagación de error numérico. Y luego hacer una FFT inversa de $\tilde{\mathbf{F}}$ para obtener \mathbf{F} en el espacio “real”, y finalmente, hacer una interpolación de ésta para obtener \mathbf{f} .

B.4. Integración ecuaciones de movimiento

Una vez calculada la fuerza de cada partícula, se prosigue a evolucionar a estas mediante un integrador. En la literatura existe una gran variedad de integradores, pero los más usados son los que vimos en los capítulos 3 y 4. La mayoría de los códigos usan ya sea Leapfrog con paso de tiempo constante o la versión adaptiva, AKDK. En particular para esta tesis, usamos el primero para el código PM.

Es importante mencionar que, como se mostró en los capítulos antes citados, el integrador HHS es una opción atractiva por su compromiso entre precisión y desempeño computacional, sin embargo, la implementación de éste en el código HHS se deja para un trabajo futuro.

Apéndice C

Pseudo-códigos del integrador de división hamiltoniana jerárquica

```
evolve(system, dt)
    calculate_timestep(system)
    fast, slow = sort_in_fast_and_slow(system, dt)
    if(fast.size>0) evolve(fast, dt/2)
    drift(slow, dt/2)
    kick(slow, fast, dt)
    kick(fast, slow, dt)
    drift(slow, dt/2)
    if(fast.size>0) evolve(fast, dt/2)
```

Figura C.1: Pseudo-código del integrador HHS en su versión recursiva.

La figura C.1 muestra el pseudo-código del algoritmo básico del integrador de división hamiltoniana jerárquica (HHS), donde las *pairwise force calculations* son realizadas en los pasos *kick*. El primer *kick* en el algoritmo es debido a los términos $V_{SS} + V_{SF}$ y el segundo es debido al término V_{FF} en la ecuación 3 del capítulo 4.

La figura C.2 muestra el pseudo-código de nuestra versión iterativa del integrador HHS.


```

evolve(system,dt,k)
  while(k>=0)
    if(fast.size[k]>0)
      calculate_timestep(system)
      fast, slow = sort_in_fast_and_slow(system,dt,k)
      system = fast[k]
      dt = dt/2
      k++
    else
      for(up=1;up<=2;up++)
        drift(slow,dt/2,k)
        kick(slow,fast,dt,k)
        kick(fast,slow,dt,k)
        drift(slow,dt/2,k)
        move_level[k]=1
        while(move_level[k]=1 && up=1)
          k--
          dt = 2*dt
          if(k=-1) return

```

Figura C.2: Pseudo-código de la versión iterativa del integrador HHS.

Apéndice D

Artículos Aceptados

D.1. Artículos Producto de la Presente Tesis

Derivado de la presente tesis fueron aceptados los siguientes dos artículos:

- G. Aguilar-Argüello, O. Valenzuela y A. Trelles. “A Boltzmann Poisson like Method to Simulate Galactic Halos Response to Satellite Accretion. Dependence on halo density profile”. En: *Astronomy and Astrophysics*. Aceptado 21 de Abril de 2022. No. Ref. AA/2021/41988
- G. Aguilar-Argüello, O. Valenzuela, H. Velázquez, J. C. Clemente y J. A. Trelles. “Assessing the Hierarchical Hamiltonian Splitting Integrator for Collisionless N -body Simulations”. En: *Revista Mexicana de Astronomía y Astrofísica* 58 (2022). Aceptado 18 Marzo de 2022.

D.2. Otros Artículos

Durante el proyecto doctoral fueron aceptados los siguientes dos artículos en colaboración:

- N. Chandrachani Devi, M. Jaber-Bravo, G. Aguilar-Argüello, O. Valenzuela, H. Velázquez y A. de la Macorra. “Non-linear structure formation for dark energy models with a Steep Equation of State”. En: *Journal of Cosmology and Astroparticle Physics* 2020.9 (2020), pág. 050. DOI: 10.1088/1475-7516/2020/09/050
- B. Abolfathi, D. S. Aguado, Gabriela Aguilar, et al. “The Fourteenth Data Release of the Sloan Digital Sky Survey: First Spectroscopic Data from the Extended Baryon Oscillation Spectroscopic Survey and from the Second Phase of the Apache Point Observatory Galactic Evolution Experiment”. En: *Astrophysical Journal, Supplement* 235.2 (2018), pág. 42. DOI: 10.3847/1538-4365/aa9e8a

Bibliografía

- [1] Adam G. Riess y col. «Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant». En: *Astronomical Journal* 116.3 (1998), págs. 1009-1038. DOI: 10.1086/300499.
- [2] S. Perlmutter y col. «Measurements of Ω and Λ from 42 High-Redshift Supernovae». En: *Astrophysical Journal* 517.2 (1999), págs. 565-586. DOI: 10.1086/307221.
- [3] Albert Einstein. «Kosmologische Betrachtungen zur allgemeinen Relativitätstheorie». En: *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften (Berlin)* (1917), págs. 142-152.
- [4] P. de Bernardis y col. «A flat Universe from high-resolution maps of the cosmic microwave background radiation». En: *Nature* 404.6781 (2000), págs. 955-959. DOI: 10.1038/35010035.
- [5] S. Hanany y col. «MAXIMA-1: A Measurement of the Cosmic Microwave Background Anisotropy on Angular Scales of 10° - 5° ». En: *Astrophysical Journal, Letters* 545.1 (2000), págs. L5-L9. DOI: 10.1086/317322.
- [6] Steven Weinberg. «The cosmological constant problem». En: *Reviews of Modern Physics* 61.1 (1989), págs. 1-23. DOI: 10.1103/RevModPhys.61.1.
- [7] Bharat Ratra y P. J. E. Peebles. «Cosmological consequences of a rolling homogeneous scalar field». En: *Physical Review D: Particles, Fields, Gravitation and Cosmology* 37.12 (1988), págs. 3406-3427. DOI: 10.1103/PhysRevD.37.3406.
- [8] Ivaylo Zlatev, Limin Wang y Paul J. Steinhardt. «Quintessence, Cosmic Coincidence, and the Cosmological Constant». En: *Physical Review Letters* 82.5 (1999), págs. 896-899. DOI: 10.1103/PhysRevLett.82.896.
- [9] Bhuvnesh Jain y Justin Khoury. «Cosmological tests of gravity». En: *Annals of Physics* 325.7 (2010), págs. 1479-1516. DOI: 10.1016/j.aop.2010.04.002.
- [10] Michael J. Mortonson, David H. Weinberg y Martin White. «Dark Energy: A Short Review». En: *arXiv e-prints*, arXiv:1401.0046 (2013), arXiv:1401.0046. eprint: 1401.0046.
- [11] Kazuya Koyama. «Cosmological tests of modified gravity». En: *Reports on Progress in Physics* 79.4 (2016), págs. 046902. DOI: 10.1088/0034-4885/79/4/046902.

- [12] Salvatore Capozziello y L. Z. Fang. «Curvature Quintessence». En: *International Journal of Modern Physics D* 11.4 (2002), págs. 483-491. DOI: 10.1142/S0218271802002025.
- [13] Sean M. Carroll y col. «Is cosmic speed-up due to new gravitational physics?». En: *Physical Review D* 70.4 (2004), pág. 043528. DOI: 10.1103/PhysRevD.70.043528.
- [14] B. Bertotti, L. Iess y P. Tortora. «A test of general relativity using radio links with the Cassini spacecraft». En: *Nature* 425.6956 (2003), págs. 374-376. DOI: 10.1038/nature01997.
- [15] S. S. Shapiro y col. «Measurement of the Solar Gravitational Deflection of Radio Waves using Geodetic Very-Long-Baseline Interferometry Data, 1979–1999». En: *Physical Review Letters* 92.12 (2004), pág. 121101. DOI: 10.1103/PhysRevLett.92.121101.
- [16] Clifford M. Will. «The Confrontation between General Relativity and Experiment». En: *Living Reviews in Relativity* 9.1 (2006), pág. 3. DOI: 10.12942/lrr-2006-3.
- [17] Y. Fukuda y col. «Evidence for Oscillation of Atmospheric Neutrinos». En: *Physical Review Letters* 81.8 (1998), págs. 1562-1567. DOI: 10.1103/PhysRevLett.81.1562.
- [18] F. P. An y col. «Observation of Electron-Antineutrino Disappearance at Daya Bay». En: *Physical Review Letters* 108.17 (2012), pág. 171803. DOI: 10.1103/PhysRevLett.108.171803.
- [19] Andreu Font-Ribera y col. «DESI and other Dark Energy experiments in the era of neutrino mass measurements». En: *Journal of Cosmology and Astroparticle Physics* 2014.5 (2014), pág. 023. DOI: 10.1088/1475-7516/2014/05/023.
- [20] David H. Weinberg y col. «Observational probes of cosmic acceleration». En: *Physics Reports* 530.2 (2013), págs. 87-255. DOI: 10.1016/j.physrep.2013.05.001.
- [21] Kyle S. Dawson y col. «The Baryon Oscillation Spectroscopic Survey of SDSS-III». En: *Astronomical Journal* 145.1 (2013), pág. 10. DOI: 10.1088/0004-6256/145/1/10.
- [22] Kyle S. Dawson y col. «The SDSS-IV Extended Baryon Oscillation Spectroscopic Survey: Overview and Early Data». En: *Astronomical Journal* 151.2 (2016), pág. 44. DOI: 10.3847/0004-6256/151/2/44.
- [23] Michael Levi y col. «The DESI Experiment, a whitepaper for Snowmass 2013». En: *arXiv e-prints*, arXiv:1308.0847 (2013), arXiv:1308.0847.
- [24] Z. Ivezić y col. «Large Synoptic Survey Telescope: From Science Drivers To Reference Design». En: *Serbian Astronomical Journal* 176 (2008), págs. 1-13. DOI: 10.2298/SAJ0876001I.
- [25] R. Laureijs y col. «Euclid Definition Study Report». En: *arXiv e-prints*, arXiv:1110.3193 (2011), arXiv:1110.3193.

- [26] D. Spergel y col. «Wide-Field InfraRed Survey Telescope-Astrophysics Focused Telescope Assets WFIRST-AFTA Final Report». En: *arXiv e-prints*, arXiv:1305.5422 (2013), arXiv:1305.5422.
- [27] G. J. Hill y col. «The Hobby-Eberly Telescope Dark Energy Experiment (HETDEX): Description and Early Pilot Survey Results». En: *Panoramic Views of Galaxy Formation and Evolution*. Ed. por T. Kodama, T. Yamada y K. Aoki. Vol. 399. Astronomical Society of the Pacific Conference Series. 2008, pág. 115. arXiv: 0806.0183.
- [28] P. J. E. Peebles. «Structure of the Coma Cluster of Galaxies». En: *Astronomical Journal* 75 (1970), pág. 13. DOI: 10.1086/110933.
- [29] William H. Press y Paul Schechter. «Formation of Galaxies and Clusters of Galaxies by Self-Similar Gravitational Condensation». En: *Astrophysical Journal* 187 (1974), págs. 425-438. DOI: 10.1086/152650.
- [30] A. A. Klypin y S. F. Shandarin. «Three-dimensional numerical model of the formation of large-scale structure in the Universe.» En: *Monthly Notices of the RAS* 204 (1983), págs. 891-907. DOI: 10.1093/mnras/204.3.891.
- [31] M. Davis y col. «The evolution of large-scale structure in a universe dominated by cold dark matter». En: *Astrophysical Journal* 292 (1985), págs. 371-394. DOI: 10.1086/163168.
- [32] G. Efstathiou y col. «Numerical techniques for large cosmological N-body simulations». En: *Astrophysical Journal, Supplement* 57 (1985), págs. 241-260. DOI: 10.1086/191003.
- [33] G. Efstathiou, W. J. Sutherland y S. J. Maddox. «The cosmological constant and cold dark matter». En: *Nature* 348.6303 (1990), págs. 705-707. DOI: 10.1038/348705a0.
- [34] A. Jenkins y col. «Evolution of Structure in Cold Dark Matter Universes». En: *Astrophysical Journal* 499.1 (1998), págs. 20-40. DOI: 10.1086/305615.
- [35] Anatoly Klypin y Jon Holtzman. «PMCode: Particle-Mesh Code for Cosmological Simulations». En: ascl:9909.001 (1999), ascl:9909.001.
- [36] Volker Springel. «The cosmological simulation code GADGET-2». En: *Monthly Notices of the RAS* 364.4 (2005), págs. 1105-1134. DOI: 10.1111/j.1365-2966.2005.09655.x.
- [37] Volker Springel y col. «Simulations of the formation, evolution and clustering of galaxies and quasars». En: *Nature* 435.7042 (2005), págs. 629-636. DOI: 10.1038/nature03597.
- [38] Anatoly A. Klypin, Sebastian Trujillo-Gomez y Joel Primack. «Dark Matter Halos in the Standard Cosmological Model: Results from the Bolshoi Simulation». En: *Astrophysical Journal* 740.2 (2011), pág. 102. DOI: 10.1088/0004-637X/740/2/102.

- [39] R. E. Angulo y col. «Scaling relations for galaxy clusters in the Millennium-XXL simulation». En: *Monthly Notices of the RAS* 426.3 (2012), págs. 2046-2062. DOI: 10.1111/j.1365-2966.2012.21830.x.
- [40] Jean-Michel Alimi y col. «DEUS Full Observable Λ CDM Universe Simulation: the numerical challenge». En: *arXiv e-prints*, arXiv:1206.2838 (2012), arXiv:1206.2838.
- [41] Salman Habib y col. «HACC: Extreme Scaling and Performance Across Diverse Architectures». En: *SC '13 Proceedings of SC13: International Conference for High Performance Computing*. 2013, pág. 6. DOI: 10.1145/2503210.2504566.
- [42] Salman Habib y col. «HACC: Simulating sky surveys on state-of-the-art supercomputing architectures». En: *New Astronomy* 42 (2016), págs. 49-65. DOI: 10.1016/j.newast.2015.06.003.
- [43] Katrin Heitmann y col. «The Mira-Titan Universe: Precision Predictions for Dark Energy Surveys». En: *Astrophysical Journal* 820.2 (2016), pág. 108. DOI: 10.3847/0004-637X/820/2/108.
- [44] Frederick Reines y Clyde L. Cowan. «The Neutrino». En: *Nature* 178.4531 (1956), págs. 446-449. DOI: 10.1038/178446a0.
- [45] Jr. Cowan C. L. y col. «Detection of the Free Neutrino: A Confirmation». En: *Science* 124.3212 (1956), págs. 103-104. DOI: 10.1126/science.124.3212.103.
- [46] G. Danby y col. «Observation of High-Energy Neutrino Reactions and the Existence of Two Kinds of Neutrinos». En: *Physical Review Letters* 9.1 (1962), págs. 36-44. DOI: 10.1103/PhysRevLett.9.36.
- [47] M. L. Perl y col. «Evidence for anomalous lepton production in e^+e^- annihilation». En: *Physical Review Letters* 35.22 (1975), págs. 1489-1492. DOI: 10.1103/PhysRevLett.35.1489.
- [48] T. D. Lee y C. N. Yang. «Parity Nonconservation and a Two-Component Theory of the Neutrino». En: *Physical Review* 105.5 (1957), págs. 1671-1675. DOI: 10.1103/PhysRev.105.1671.
- [49] L. Landau. «On the conservation laws for weak interactions». En: *Nuclear Physics* 3.1 (1957), págs. 127-131. DOI: 10.1016/0029-5582(57)90061-5.
- [50] Abdus Salam. «On parity conservation and neutrino mass». En: *Il Nuovo Cimento* 5.1 (1957), págs. 299-301. DOI: 10.1007/BF02812841.
- [51] Scott Dodelson, Evalyn Gates y Albert Stebbins. «Cold + Hot Dark Matter and the Cosmic Microwave Background». En: *Astrophysical Journal* 467 (1996), pág. 10. DOI: 10.1086/177581.
- [52] Wayne Hu y Scott Dodelson. «Cosmic Microwave Background Anisotropies». En: *Annual Review of Astronomy and Astrophysics* 40 (2002), págs. 171-216. DOI: 10.1146/annurev.astro.40.060401.093926.
- [53] Julien Lesgourgues y Sergio Pastor. «Neutrino mass from Cosmology». En: *arXiv e-prints*, arXiv:1212.6154 (2012), arXiv:1212.6154.

- [54] Julien Lesgourgues y col. *Neutrino Cosmology*. 2013.
- [55] Wayne Hu, Daniel J. Eisenstein y Max Tegmark. «Weighing Neutrinos with Galaxy Surveys». En: *Physical Review Letters* 80.24 (1998), págs. 5255-5258. DOI: 10.1103/PhysRevLett.80.5255.
- [56] Julien Lesgourgues y Sergio Pastor. «Massive neutrinos and cosmology». En: *Physics Reports* 429.6 (2006), págs. 307-379. DOI: 10.1016/j.physrep.2006.04.001.
- [57] Gong-Bo Zhao y col. «The clustering of galaxies in the SDSS-III Baryon Oscillation Spectroscopic Survey: weighing the neutrino mass using the galaxy power spectrum of the CMASS sample». En: *Monthly Notices of the Royal Astronomical Society* 436.3 (2013), págs. 2038-2053. DOI: 10.1093/mnras/stt1710.
- [58] Sunny Vagnozzi y col. «Unveiling ν secrets with cosmological data: Neutrino masses and mass hierarchy». En: *Physical Review D: Particles, Fields, Gravitation and Cosmology* 96.12 (2017), pág. 123503. DOI: 10.1103/PhysRevD.96.123503.
- [59] Amol Upadhye. «Neutrino mass and dark energy constraints from redshift-space distortions». En: *Journal of Cosmology and Astroparticle Physics* 2019.5 (2019), pág. 041. DOI: 10.1088/1475-7516/2019/05/041.
- [60] Nathalie Palanque-Delabrouille y col. «Hints, neutrino bounds, and WDM constraints from SDSS DR14 Lyman- α and Planck full-survey data». En: *Journal of Cosmology and Astroparticle Physics* 2020.4 (2020), pág. 038. DOI: 10.1088/1475-7516/2020/04/038.
- [61] Eleonora Di Valentino, Stefano Gariazzo y Olga Mena. «Most constraining cosmological neutrino mass bounds». En: *Physical Review D* 104.8 (2021), pág. 083504. DOI: 10.1103/PhysRevD.104.083504.
- [62] Darren S. Reed y col. «Towards an accurate mass function for precision cosmology». En: *Monthly Notices of the Royal Astronomical Society* 431.2 (2013), págs. 1866-1882. DOI: 10.1093/mnras/stt301.
- [63] Katrin Heitmann y col. «The Outer Rim Simulation: A Path to Many-core Supercomputers». En: *Astrophysical Journal, Supplement* 245.1 (2019), pág. 16. DOI: 10.3847/1538-4365/ab4da1.
- [64] C. L. Bennett y col. «Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Final Maps and Results». En: *Astrophysical Journal, Supplement* 208.2 (2013), pág. 20. DOI: 10.1088/0067-0049/208/2/20.
- [65] Planck Collaboration y col. «Planck 2013 results. XVI. Cosmological parameters». En: *Astronomy and Astrophysics* 571 (2014), A16. DOI: 10.1051/0004-6361/201321591.
- [66] Behzad Ansarinejad y Tom Shanks. «Detection significance of baryon acoustic oscillations peaks in galaxy and quasar clustering». En: *Monthly Notices of the RAS* 479.3 (2018), págs. 4091-4107. DOI: 10.1093/mnras/sty1740.

- [67] Lauren Anderson y col. «The clustering of galaxies in the SDSS-III Baryon Oscillation Spectroscopic Survey: baryon acoustic oscillations in the Data Releases 10 and 11 Galaxy samples». En: *Monthly Notices of the Royal Astronomical Society* 441.1 (2014), págs. 24-62. DOI: 10.1093/mnras/stu523.
- [68] S. Johnston y col. «Science with ASKAP. The Australian square-kilometre-array pathfinder». En: *Experimental Astronomy* 22.3 (2008), págs. 151-273. DOI: 10.1007/s10686-008-9124-7.
- [69] P. E. Dewdney y col. «The Square Kilometre Array». En: *IEEE Proceedings* 97.8 (2009), págs. 1482-1496. DOI: 10.1109/JPROC.2009.2021005.
- [70] Michael Kuhlen, Mark Vogelsberger y Raul Angulo. «Numerical simulations of the dark universe: State of the art and the next decade». En: *Physics of the Dark Universe* 1.1-2 (2012), págs. 50-93. DOI: 10.1016/j.dark.2012.10.002.
- [71] R. Teyssier. «Cosmological hydrodynamics with adaptive mesh refinement. A new high resolution code called RAMSES». En: *Astronomy and Astrophysics* 385 (2002), págs. 337-364. DOI: 10.1051/0004-6361:20011817.
- [72] Juhan Kim y col. «The New Horizon Run Cosmological N-Body Simulations». En: *Journal of Korean Astronomical Society* 44.6 (2011), págs. 217-234. DOI: 10.5303/JKAS.2011.44.6.217.
- [73] John Dubinski y col. «GOTPM: a parallel hybrid particle-mesh treecode». En: *New Astronomy* 9.2 (2004), págs. 111-126. DOI: 10.1016/j.newast.2003.08.002.
- [74] W. A. Watson y col. «The Jubilee ISW project - I. Simulated ISW and weak lensing maps and initial power spectra results». En: *Monthly Notices of the Royal Astronomical Society* 438.1 (2014), págs. 412-425. DOI: 10.1093/mnras/stt2208.
- [75] Joachim Harnois-Deraps y col. *CUBEP3M: High performance P3M N-body code*. 2012.
- [76] Andrey V. Kravtsov, Anatoly A. Klypin y Alexei M. Khokhlov. «Adaptive Refinement Tree: A New High-Resolution N-Body Code for Cosmological Simulations». En: *Astrophysical Journal, Supplement* 111.1 (1997), págs. 73-94. DOI: 10.1086/313015.
- [77] Tomoaki Ishiyama, Toshiyuki Fukushima y Junichiro Makino. «GreeM: Massively Parallel TreePM Code for Large Cosmological N-body Simulations». En: *Publications of the Astronomical Society of Japan* 61 (2009), pág. 1319. DOI: 10.1093/pasj/61.6.1319.
- [78] Tomoaki Ishiyama, Keigo Nitadori y Junichiro Makino. «4.45 Pflops Astrophysical N-Body Simulation on K computer – The Gravitational Trillion-Body Problem». En: *arXiv e-prints*, arXiv:1211.4406 (2012), arXiv:1211.4406.
- [79] P. Jetley y col. «Massively parallel cosmological simulations with ChaNGa». En: *2008 IEEE International Symposium on Parallel and Distributed Processing*. 2008, págs. 1-12. DOI: 10.1109/IPDPS.2008.4536319.

- [80] P. Jetley y col. «Scaling Hierarchical N-body Simulations on GPU Clusters». En: *SC '10: Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*. 2010, págs. 1-11. DOI: 10.1109/SC.2010.49.
- [81] Lehman H. Garrison y col. «The ABACUS cosmological N-body code». En: *Monthly Notices of the RAS* 508.1 (2021), págs. 575-596. DOI: 10.1093/mnras/stab2482.
- [82] Cameron Grove y col. «The DESI N-body Simulation Project I: Testing the Robustness of Simulations for the DESI Dark Time Survey». En: *arXiv e-prints*, arXiv:2112.09138 (dic. de 2021), arXiv:2112.09138. arXiv: 2112.09138 [astro-ph.CO].
- [83] Anand Raichoor y col. «The completed SDSS-IV extended Baryon Oscillation Spectroscopic Survey: large-scale structure catalogues and measurement of the isotropic BAO between redshift 0.6 and 1.1 for the Emission Line Galaxy Sample». En: *Monthly Notices of the RAS* 500.3 (2021), págs. 3254-3274. DOI: 10.1093/mnras/staa3336.
- [84] Cheng Zhao y col. «The completed SDSS-IV extended Baryon Oscillation Spectroscopic Survey: 1000 multi-tracer mock catalogues with redshift evolution and systematics for galaxies and quasars of the final data release». En: *Monthly Notices of the RAS* 503.1 (2020), págs. 1149-1173. DOI: 10.1093/mnras/stab510.
- [85] Y. B. Zel'Dovich. «Gravitational instability: an approximate theory for large density perturbations.» En: *Astronomy and Astrophysics* 5 (1970), págs. 84-89.
- [86] R. E. Smith y col. «Stable clustering, the halo model and non-linear cosmological power spectra». En: *Monthly Notices of the RAS* 341.4 (2003), págs. 1311-1332. DOI: 10.1046/j.1365-8711.2003.06503.x.
- [87] Ryuichi Takahashi y col. «Revising the Halofit Model for the Nonlinear Matter Power Spectrum». En: *Astrophysical Journal* 761.2 (2012), pág. 152. DOI: 10.1088/0004-637X/761/2/152.
- [88] Simeon Bird, Matteo Viel y Martin G. Haehnelt. «Massive neutrinos and the non-linear matter power spectrum». En: *Monthly Notices of the RAS* 420.3 (2012), págs. 2551-2561. DOI: 10.1111/j.1365-2966.2011.20222.x.
- [89] Hans A. Winther y col. «COLA with scale-dependent growth: applications to screened modified gravity models». En: *Journal of Cosmology and Astroparticle Physics* 2017.8 (2017), pág. 006. DOI: 10.1088/1475-7516/2017/08/006.
- [90] Bill S. Wright, Hans A. Winther y Kazuya Koyama. «COLA with massive neutrinos». En: *Journal of Cosmology and Astroparticle Physics* 2017.10 (2017), pág. 054. DOI: 10.1088/1475-7516/2017/10/054.
- [91] Adrian E. Bayer, Arka Banerjee y Yu Feng. «A fast particle-mesh simulation of non-linear cosmological structure formation with massive neutrinos». En: *Journal of Cosmology and Astroparticle Physics* 2021.1 (2021), pág. 016. DOI: 10.1088/1475-7516/2021/01/016.

- [92] S. J. Aarseth. «Numerical Experiments on the N-Body Problem (Papers appear in the Proceedings of IAU Colloquium No. 10 Gravitational N-Body Problem (ed. by Myron Lecar), R. Reidel Publ. Co., Dordrecht-Holland.)» En: *Astrophysics and Space Science* 14 (1971), págs. 20-34. DOI: 10.1007/BF00649191.
- [93] W. Dehnen y J. I. Read. «N-body simulations of gravitational dynamics». En: *European Physical Journal Plus* 126 (2011), pág. 55. DOI: 10.1140/epjp/i2011-11055-3.
- [94] Guohong Xu. «A New Parallel N-Body Gravity Solver: TPM». En: *Astrophysical Journal Supplement* 98 (1995), pág. 355. DOI: 10.1086/192166.
- [95] Paul Bode, Jeremiah P. Ostriker y Guohong Xu. «The Tree Particle-Mesh N-Body Gravity Solver». En: *Astrophysical Journal Supplement* 128 (2000), págs. 561-569. DOI: 10.1086/313398.
- [96] J. S. Bagla. «TreePM: A Code for Cosmological N-Body Simulations». En: *Journal of Astrophysics and Astronomy* 23 (2002), págs. 185-196. DOI: 10.1007/BF02702282.
- [97] Paul Bode y Jeremiah P. Ostriker. «Tree Particle-Mesh: An Adaptive, Efficient, and Parallel Code for Collisionless Cosmological Simulation». En: *Astrophysical Journal Supplement* 145 (2003), págs. 1-13. DOI: 10.1086/345538.
- [98] Jens Verner Villumsen. «A New Hierarchical Particle-Mesh Code for Very Large Scale Cosmological N-Body Simulations». En: *Astrophysical Journal Supplement* 71 (1989), pág. 407. DOI: 10.1086/191380.
- [99] Chris Jessop, Martin Duncan y W. Y. Chau. «Multigrid Methods for N-Body Gravitational Systems». En: *Journal of Computational Physics* 115 (1994), págs. 339-351. DOI: 10.1006/jcph.1994.1200.
- [100] Piet Hut, Jun Makino y Steve McMillan. «Building a Better Leapfrog». En: *Astrophysical Journal Letters* 443 (1995), pág. L93. DOI: 10.1086/187844.
- [101] Federico I. Pelupessy, Jürgen Jänes y Simon Portegies Zwart. «N-body integrators with individual time steps from Hierarchical splitting». En: *New Astronomy* 17 (2012), págs. 711-719. DOI: 10.1016/j.newast.2012.05.009.
- [102] R. Capuzzo-Dolcetta, A. Mastrobuono-Battisti y D. Maschietti. «NBSymple, a double parallel, symplectic N-body code running on graphic processing units». En: *New Astronomy* 16 (2011), págs. 284-295. DOI: 10.1016/j.newast.2010.11.004.
- [103] Keigo Nitadori y Sverre J. Aarseth. «Accelerating NBODY6 with graphics processing units». En: *Monthly Notices of the RAS* 424 (2012), págs. 545-552. DOI: 10.1111/j.1365-2966.2012.21227.x.
- [104] J. Bédorf y S. Portegies Zwart. «A pilgrimage to gravity on GPUs». En: *European Physical Journal Special Topics* 24 (2012), págs. 201-216. DOI: 10.1140/epjst/e2012-1647-6.

- [105] R. Capuzzo-Dolcetta, M. Spera y D. Punzo. «A fully parallel, high precision, N-body code running on hybrid computing platforms». En: *Journal of Computational Physics* 236 (2013), págs. 580-593. DOI: 10.1016/j.jcp.2012.11.013.
- [106] Junichiro Makino y Sverre J. Aarseth. «On a Hermite Integrator with Ahmad-Cohen Scheme for Gravitational Many-Body Problems». En: *Publications of the Astronomical Society of Japan* 44 (1992), págs. 141-151.
- [107] A. N. Baushev y S. V. Pilipenko. «The central cusps in dark matter halos: fact or fiction?» En: *arXiv e-prints* arXiv:1808.03088 (2018), arXiv:1808.03088.
- [108] Junichiro Makino. «A Modified Aarseth Code for GRAPE and Vector Processors». En: *Publications of the Astronomical Society of Japan* 43 (1991), págs. 859-876.
- [109] Eiichiro Kokubo, Keiko Yoshinaga y Junichiro Makino. «On a time-symmetric Hermite integrator for planetary N-body simulation». En: *Monthly Notices of the RAS* 297.4 (1998), págs. 1067-1072. DOI: 10.1046/j.1365-8711.1998.01581.x.
- [110] David M. Hernandez. «Should N-body integrators be symplectic everywhere in phase space?» En: *Monthly Notices of the RAS* 486 (2019), págs. 5231-5238. DOI: 10.1093/mnras/stz884.
- [111] Julio F. Navarro, Carlos S. Frenk y Simon D. M. White. «A Universal Density Profile from Hierarchical Clustering». En: *Astrophysical Journal* 490 (1997), págs. 493-508. DOI: 10.1086/304888.
- [112] Carl Pearson y col. «NUMA-Aware Data-Transfer Measurements for Power/NVLink Multi-GPU Systems». En: *ISC Workshops* 111 (2018), págs. 73-94. DOI: 10.1007/978-3-030-02465-9_32.
- [113] David M. Hernandez, Sam Hadden y Junichiro Makino. «Are long-term N-body simulations reliable?» En: *Monthly Notices of the RAS* 493 (2020), págs. 1913-1925. DOI: 10.1093/mnras/staa388.
- [114] Ernst Hairer, Christian Lubich y Gerhard Wanner. *Geometric numerical integration Structure -Preserving algorithms for ordinary differential equations*. 1.^a ed. Vol. 31. Springer-Verlag Berlin Heidelberg, 2002. DOI: 10.1007/978-3-662-05018-7.
- [115] Kohji Yoshikawa, Naoki Yoshida y Masayuki Umemura. «Direct Integration of the Collisionless Boltzmann Equation in Six-dimensional Phase Space: Self-gravitating Systems». En: *Astrophysical Journal* 762.2 (2013), pág. 116. DOI: 10.1088/0004-637X/762/2/116.
- [116] Kohji Yoshikawa y col. «Cosmological Vlasov-Poisson Simulations of Structure Formation with Relic Neutrinos: Nonlinear Clustering and the Neutrino Mass». En: *arXiv e-prints*, arXiv:2010.00248 (2020), arXiv:2010.00248.
- [117] Caio Bastos de Senna Nascimento y Marilena Loverde. «Neutrinos in N-body simulations». En: *arXiv e-prints*, arXiv:2102.05690 (2021), arXiv:2102.05690.

- [118] J. D. Emberson y col. «Cosmological neutrino simulations at extreme scale». En: *Research in Astronomy and Astrophysics* 17.8 (2017), pág. 085. DOI: 10.1088/1674-4527/17/8/85.
- [119] Joachim Harnois-Déraps y col. «High-performance P³M N-body code: CUBEP³M». En: *Monthly Notices of the Royal Astronomical Society* 436.1 (2013), págs. 540-559. DOI: 10.1093/mnras/stt1591.
- [120] Francisco Villaescusa-Navarro y col. «The Quijote Simulations». En: *Astrophysical Journal, Supplement* 250.1 (2020), pág. 2. DOI: 10.3847/1538-4365/ab9d82.
- [121] Arka Banerjee y col. «Reducing noise in cosmological N-body simulations with neutrinos». En: *Journal of Cosmology and Astroparticle Physics* 2018.9 (2018), pág. 028. DOI: 10.1088/1475-7516/2018/09/028.
- [122] Anatoly Klypin y col. «Structure Formation with Cold plus Hot Dark Matter». En: *Astrophysical Journal* 416 (1993), pág. 1. DOI: 10.1086/173210.
- [123] J. A. Sellwood. «Quiet starts for galaxy simulations.» En: *Journal of Computational Physics* 50 (1983), págs. 337-359. DOI: 10.1016/0021-9991(83)90102-X.
- [124] Jack A. Byers y Mahesh Grewal. «Perpendicularly Propagating Plasma Cyclotron Instabilities Simulated with a One-Dimensional Computer Model». En: *Physics of Fluids* 13.7 (1970), págs. 1819-1830. DOI: 10.1063/1.1693160.
- [125] Jacob Brandbyge, Steen Hannestad y Thomas Tram. «Momentum space sampling of neutrinos in N-body simulations». En: *Journal of Cosmology and Astroparticle Physics* 2019.3 (2019), pág. 047. DOI: 10.1088/1475-7516/2019/03/047.
- [126] Jacob Brandbyge y Steen Hannestad. «Grid based linear neutrino perturbations in cosmological N-body simulations». En: *Journal of Cosmology and Astroparticle Physics* 2009.5 (2009), pág. 002. DOI: 10.1088/1475-7516/2009/05/002.
- [127] Maria Archidiacono y Steen Hannestad. «Efficient calculation of cosmological neutrino clustering in the non-linear regime». En: *Journal of Cosmology and Astroparticle Physics* 2016.6 (2016), pág. 018. DOI: 10.1088/1475-7516/2016/06/018.
- [128] Yacine Ali-Haïmoud y Simeon Bird. «An efficient implementation of massive neutrinos in non-linear structure formation simulations». En: *Monthly Notices of the Royal Astronomical Society* 428.4 (2013), págs. 3375-3389. DOI: 10.1093/mnras/sts286.
- [129] Jia Liu y col. «MassiveNuS: cosmological massive neutrino simulations». En: *Journal of Cosmology and Astroparticle Physics* 2018.3 (2018), pág. 049. DOI: 10.1088/1475-7516/2018/03/049.
- [130] Svetlin Tassev, Matias Zaldarriaga y Daniel J. Eisenstein. «Solving large scale structure in ten easy steps with COLA». En: *Journal of Cosmology and Astroparticle Physics* 2013.6 (2013), pág. 036. DOI: 10.1088/1475-7516/2013/06/036.

- [131] Svetlin V. Tassev. *COLAcode: COmoving Lagrangian Acceleration code*. 2016.
- [132] Willem Elbers y col. «An optimal nonlinear method for simulating relic neutrinos». En: *arXiv e-prints*, arXiv:2010.07321 (2020), arXiv:2010.07321.
- [133] Arka Banerjee y Neal Dalal. «Simulating nonlinear cosmological structure formation with massive neutrinos». En: *Journal of Cosmology and Astroparticle Physics* 2016.11 (2016), pág. 015. DOI: 10.1088/1475-7516/2016/11/015.
- [134] Jeppe Dakin y col. « ν CONCEPT: cosmological neutrino simulations from the non-linear Boltzmann hierarchy». En: *Journal of Cosmology and Astroparticle Physics* 2019.2 (2019), pág. 052. DOI: 10.1088/1475-7516/2019/02/052.
- [135] Diego Blas, Julien Lesgourgues y Thomas Tram. «The Cosmic Linear Anisotropy Solving System (CLASS). Part II: Approximation schemes». En: *Journal of Cosmology and Astroparticle Physics* 2011.7 (2011), pág. 034. DOI: 10.1088/1475-7516/2011/07/034.
- [136] Julien Lesgourgues. «The Cosmic Linear Anisotropy Solving System (CLASS) III: Comparison with CAMB for LambdaCDM». En: *arXiv e-prints*, arXiv:1104.2934 (2011), arXiv:1104.2934.
- [137] Graziano Rossi y col. «Suite of hydrodynamical simulations for the Lyman- α forest with massive neutrinos». En: *Astronomy and Astrophysics* 567 (2014), A79. DOI: 10.1051/0004-6361/201423507.
- [138] V. F. Mukhanov, H. A. Feldman y R. H. Brandenberger. «Theory of cosmological perturbations». En: *Physics Reports* 215.5-6 (1992), págs. 203-333. DOI: 10.1016/0370-1573(92)90044-Z.
- [139] Scott Dodelson. *Modern cosmology*. 2003.
- [140] R. W. Hockney y J. W. Eastwood. *Computer simulation using particles*. 1988.
- [141] Yu Feng y col. «FASTPM: a new scheme for fast simulations of dark matter and haloes». En: *Monthly Notices of the RAS* 463.3 (2016), págs. 2273-2286. DOI: 10.1093/mnras/stw2123.
- [142] Anatoly Klypin y Francisco Prada. «Dark matter statistics for large galaxy catalogues: power spectra and covariance matrices». En: *Monthly Notices of the RAS* 478.4 (2018), págs. 4602-4621. DOI: 10.1093/mnras/sty1340.
- [143] Charles K. Birdsall y Dieter Fuss. «Clouds-in-clouds, clouds-in-cells physics for many-body plasma simulation». En: *Journal of Computational Physics* 3.4 (1969), págs. 494-511. DOI: 10.1016/0021-9991(69)90058-8.
- [144] Matteo Frigo y Steven G. Johnson. «The Design and Implementation of FFTW3». En: *Proceedings of the IEEE* 93.2 (2005), págs. 216-231. DOI: 10.1109/JPROC.2004.840301.
- [145] Vladimír Fuka. «PoisFFT - A Free Parallel Fast Poisson Solver». En: *arXiv e-prints*, arXiv:1409.8116 (2014), arXiv:1409.8116.
- [146] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer-Verlag Berlin Heidelberg, 2009. DOI: 10.1007/b79761.

- [147] P. L. Roe. «Characteristic-based schemes for the euler equations». En: *Annual Review of Fluid Mechanics* 18 (1986), págs. 337-365. DOI: 10.1146/annurev.fl.18.010186.002005.
- [148] Ami Harten. «High Resolution Schemes for Hyperbolic Conservation Laws». En: *Journal of Computational Physics* 49.3 (1983), págs. 357-393. DOI: 10.1016/0021-9991(83)90136-5.
- [149] S. Gottlieb y C. W. Shu. «Total variation diminishing Runge-Kutta schemes». En: *Mathematics of Computation* 67.221 (1988), págs. 73-85. DOI: 10.1090/S0025-5718-98-00913-2.
- [150] H. C. Plummer. «On the problem of distribution in globular star clusters». En: *Monthly Notices of the RAS* 71 (1911), págs. 460-470. DOI: 10.1093/mnras/71.5.460.
- [151] Nicolas Garavito-Camargo y col. «Quantifying the impact of the Large Magellanic Cloud on the structure of the Milky Way's dark matter halo using Basis Function Expansions». En: *arXiv e-prints*, arXiv:2010.00816 (2020), arXiv:2010.00816.
- [152] Tomas Tamfal y col. «Revisiting Dynamical Friction: The Role of Global Modes and Local Wakes». En: *Astrophysical Journal* 916.1 (2021), págs. 55. DOI: 10.3847/1538-4357/ac0627.
- [153] K. C. Chambers y col. «The Pan-STARRS1 Surveys». En: *arXiv e-prints*, arXiv:1612.05560 (2016), arXiv:1612.05560.
- [154] Paul Bode, Jeremiah P. Ostriker y Neil Turok. «Halo Formation in Warm Dark Matter Models». En: *Astrophysical Journal* 556.1 (2001), págs. 93-107. DOI: 10.1086/321541.
- [155] Pedro Colín, Vladimir Avila-Reese y Octavio Valenzuela. «Substructure and Halo Density Profiles in a Warm Dark Matter Cosmology». En: *Astrophysical Journal* 542.2 (2000), págs. 622-630. DOI: 10.1086/317057.
- [156] Anatoly Klypin y col. «Where Are the Missing Galactic Satellites?» En: *Astrophysical Journal* 522.1 (1999), págs. 82-92. DOI: 10.1086/307643.
- [157] Ben Moore y col. «Dark Matter Substructure within Galactic Halos». En: *Astrophysical Journal, Letters* 524.1 (1999), págs. L19-L22. DOI: 10.1086/312287.
- [158] G. Kauffmann, S. D. M. White y B. Guiderdoni. «The formation and evolution of galaxies within merging dark matter haloes.» En: *Monthly Notices of the RAS* 264 (1993), págs. 201-218. DOI: 10.1093/mnras/264.1.201.
- [159] Vladimir Avila-Reese y col. «Formation and Structure of Halos in a Warm Dark Matter Cosmology». En: *Astrophysical Journal* 559.2 (2001), págs. 516-530. DOI: 10.1086/322411.
- [160] Pedro Colín, Octavio Valenzuela y Vladimir Avila-Reese. «On the Structure of Dark Matter Halos at the Damping Scale of the Power Spectrum with and without Relict Velocities». En: *Astrophysical Journal* 673.1 (2008), págs. 203-214. DOI: 10.1086/524030.

- [161] Jie Wang y Simon D. M. White. «Discreteness effects in simulations of hot/warm dark matter». En: *Monthly Notices of the RAS* 380.1 (2007), págs. 93-103. DOI: 10.1111/j.1365-2966.2007.12053.x.
- [162] Mark R. Lovell y col. «The properties of warm dark matter haloes». En: *Monthly Notices of the RAS* 439.1 (2014), págs. 300-317. DOI: 10.1093/mnras/stt2431.
- [163] P. Colín, V. Avila-Reese y O. Valenzuela. «Substructure and Halo Density Profiles in a Warm Dark Matter Cosmology». En: *Galaxy Disks and Disk Galaxies*. Ed. por José G. Funes y Enrico Maria Corsini. Vol. 230. Astronomical Society of the Pacific Conference Series. 2001, págs. 651-652. arXiv: astro-ph/0009317.
- [164] Mark R. Lovell y col. «The spatial distribution of Milky Way satellites, gaps in streams and the nature of dark matter». En: *arXiv e-prints*, arXiv:2104.03322 (2021), arXiv:2104.03322.
- [165] V. Springel y col. «The Aquarius Project: the subhaloes of galactic haloes». En: *Monthly Notices of the RAS* 391.4 (2008), págs. 1685-1711. DOI: 10.1111/j.1365-2966.2008.14066.x.
- [166] J. Stadel y col. «Quantifying the heart of darkness with GHALO - a multibillion particle simulation of a galactic halo». En: *Monthly Notices of the RAS* 398.1 (2009), págs. L21-L25. DOI: 10.1111/j.1745-3933.2009.00699.x.
- [167] J. Diemand y col. «Clumps and streams in the local dark matter distribution». En: *Nature* 454.7205 (2008), págs. 735-738. DOI: 10.1038/nature07153.
- [168] Volker Springel y col. «Simulating cosmic structure formation with the GADGET-4 code». En: *Monthly Notices of the RAS* 506.2 (2021), págs. 2871-2949. DOI: 10.1093/mnras/stab1855.
- [169] Rennan Barkana, Zoltán Haiman y Jeremiah P. Ostriker. «Constraints on Warm Dark Matter from Cosmological Reionization». En: *Astrophysical Journal* 558.2 (2001), págs. 482-496. DOI: 10.1086/322393.
- [170] Peter S. Behroozi, Risa H. Wechsler y Hao-Yi Wu. «The ROCKSTAR Phase-space Temporal Halo Finder and the Velocity Offsets of Cluster Cores». En: *Astrophysical Journal* 762.2 (2013), pág. 109. DOI: 10.1088/0004-637X/762/2/109.
- [171] Sownak Bose y col. «Substructure and galaxy formation in the Copernicus Complexio warm dark matter simulations». En: *Monthly Notices of the RAS* 464.4 (2017), págs. 4520-4533. DOI: 10.1093/mnras/stw2686.
- [172] Philip Mocz y col. «Schrödinger-Poisson-Vlasov-Poisson correspondence». En: *Physical Review D* 97.8 (2018), pág. 083519. DOI: 10.1103/PhysRevD.97.083519.
- [173] Philip Mocz y col. «Galaxy formation with BECDM - I. Turbulence and relaxation of idealized haloes». En: *Monthly Notices of the RAS* 471.4 (2017), págs. 4559-4570. DOI: 10.1093/mnras/stx1887.

- [174] Axel de la Macorra y Erick Almaraz. «Theoretical and Observational Constraints of Bound Dark Energy with Precision Cosmological Data». En: *Physical Review Letters* 121.16 (2018), pág. 161303. DOI: 10.1103/PhysRevLett.121.161303.
- [175] Erick Almaraz y Axel de la Macorra. «Bound dark energy: Towards understanding the nature of dark energy». En: *Physical Review D* 99.10 (2019), pág. 103504. DOI: 10.1103/PhysRevD.99.103504.
- [176] Raul E. Angulo, Oliver Hahn y Tom Abel. «The warm dark matter halo mass function below the cut-off scale». En: *Monthly Notices of the RAS* 434.4 (2013), págs. 3337-3347. DOI: 10.1093/mnras/stt1246.
- [177] B. Abi y col. «Volume I. Introduction to DUNE». En: *Journal of Instrumentation* 15.8 (2020), T08008. DOI: 10.1088/1748-0221/15/08/T08008.
- [178] M. Zennaro y col. «Initial conditions for accurate N-body simulations of massive neutrino cosmologies». En: *Monthly Notices of the RAS* 466.3 (2017), págs. 3244-3258. DOI: 10.1093/mnras/stw3340.
- [179] Shouvik Roy Choudhury, Steen Hannestad y Thomas Tram. «Updated constraints on massive neutrino self-interactions from cosmology in light of the H_0 tension». En: *Journal of Cosmology and Astroparticle Physics* 2021.3 (2021), pág. 084. DOI: 10.1088/1475-7516/2021/03/084.
- [180] Miguel Escudero y Samuel J. Witte. «The hubble tension as a hint of leptogenesis and neutrino mass generation». En: *European Physical Journal C* 81.6, 515 (jun. de 2021), pág. 515. DOI: 10.1140/epjc/s10052-021-09276-5. arXiv: 2103.03249 [hep-ph].