

UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO  
ESCUELA DE CIENCIAS FÍSICO-MATEMÁTICAS  
LUIS MANUEL RIVERA GUTIÉRREZ

IMPLEMENTACIÓN DE LA RETROALIMENTACIÓN  
DE LAS ESTRELLAS RECIÉN NACIDAS SOBRE LA  
NUBE MADRE



TESIS

QUE PARA OBTENER EL TÍTULO DE  
LICENCIADO EN CIENCIAS FÍSICO-MATEMÁTICAS

PRESENTA  
RAÚL NARANJO ROMERO

ASESOR: DR. ENRIQUE VÁZQUEZ SEMADENI  
CO-ASESOR: DR. GILBERTO C. GÓMEZ REYES

CENTRO DE RADIOASTRONOMÍA y ASTROFÍSICA,  
UNAM, CAMPUS MORELIA





*A ti*

Dios mio, a ustedes mamá y papá, a mis tres amores:  
Susy, Antar Nahim y Rhaúl Abraham

*que con tu amor y comprensión me motivaste para  
llegar hasta aquí...*



Gracias a todos los que hicieron esto posible:

En especial a mi amada esposa Susy con quien quedaré eternamente agradecido por el gran esfuerzo con que me ha apoyado para continuar mis estudios.

A mis dos hijos Antar Nahim y Rhaúl Abraham, por quienes me supero día a día y llenan alegría y esperanza mi vida.

A mis asesores, el Dr. Enrique Vázquez Semadeni y el Dr. Gilberto C. Gómez Reyes por compartir sus conocimientos no solo en el desarrollo del presente trabajo sino para que tuviese otra perspectiva de mi formación académica. A mi tutor, el Dr. Javier Ballesteros Paredes por la invitación a colaborar en algunos proyectos que me permitieron desarrollar mi potencial. A ustedes debo gran parte de mis expectativas, gracias por su apoyo, confianza y sincera amistad.

A mis profesores, en especial a la Dra. Tatjana Vukasinac, Dr. Abdón Choque Rivero, Dr. Héctor Tejeda Villela, Dr. Mario César Suárez Arriaga, Dra. Longina Castellanos, Dr. Francisco Domínguez Mota y Dr. Laurent Loinard que han motivado en gran medida mis estudios.

A mis amigos F.M. Oscar Ramírez y F.M. Jesús Rivera Castañeda por los momentos compartidos.

# Índice general

<b>1. Introducción</b>	<b>11</b>
<b>2. El Medio Interestelar</b>	<b>15</b>
2.1. Dinámica y termodinámica del Medio Interestelar . . . . .	16
2.1.1. Ecuaciones . . . . .	17
2.1.2. Termodinámica . . . . .	19
2.1.3. Regiones HII . . . . .	24
2.1.4. Evolución de las regiones HII . . . . .	29
2.1.5. Turbulencia . . . . .	37
2.2. Componentes del medio interestelar . . . . .	41
2.2.1. Gas ionizado caliente . . . . .	44
2.2.2. Gas ionizado tibio . . . . .	44
2.2.3. Gas neutro tibio . . . . .	45
2.2.4. Gas neutro frío . . . . .	45
2.2.5. Gas molecular . . . . .	46
2.2.6. Regiones HII . . . . .	46
2.3. Nubes Moleculares . . . . .	47
2.3.1. Estructura de las nubes moleculares . . . . .	48
2.3.2. Evolución . . . . .	50
<b>3. Formación estelar</b>	<b>53</b>
3.1. Principios básicos . . . . .	53
3.1.1. Inestabilidad gravitacional . . . . .	53
3.1.2. Tiempo de caída libre . . . . .	54
3.1.3. Fragmentación de Hoyle . . . . .	55
3.1.4. Campo magnético. Soporte y relación masa-flujo . . . . .	56
3.2. Problemas contemporáneos . . . . .	57
3.2.1. Modelo “estándar” de FE regulada magnéticamente . . . . .	57

3.2.2.	La tasa y la eficiencia de la formación estelar . . . . .	60
3.2.3.	Función de masa inicial . . . . .	62
3.3.	Mecanismos de regulación de la FE . . . . .	63
3.3.1.	Difusión ambipolar . . . . .	63
3.3.2.	Retroalimentación . . . . .	63
<b>4.</b>	<b>Descripción del código GADGET2.</b>	<b>69</b>
4.1.	Método SPH . . . . .	69
4.2.	Cálculo de la fuerza gravitacional . . . . .	71
4.2.1.	Esquema de integración temporal. . . . .	75
4.3.	Versión CRyA . . . . .	80
4.3.1.	Enfriamiento y calentamiento . . . . .	80
4.3.2.	Partículas sumidero (“sinks”) . . . . .	83
<b>5.</b>	<b>Implementación de la retroalimentación</b>	<b>85</b>
5.1.	Algoritmo de Dale <i>et al.</i> (2007) . . . . .	85
5.2.	Incorporación a GADGET2 . . . . .	88
5.2.1.	Lectura de parámetros y condiciones iniciales . . . . .	89
5.2.2.	Cálculo de la FMI para las partículas sumidero . . . . .	90
5.2.3.	Encendido y apagado de estrellas en partículas sumidero . . . . .	90
5.2.4.	Retroalimentación por fotoionización . . . . .	91
<b>6.</b>	<b>Pruebas y resultados</b>	<b>93</b>
6.1.	Parámetros de la simulación . . . . .	93
6.1.1.	Interpretación cualitativa de la simulación . . . . .	94
6.2.	Prueba del algoritmo . . . . .	98
6.2.1.	Interpretación física de la simulación . . . . .	102
<b>7.</b>	<b>Conclusiones</b>	<b>107</b>
7.1.	Resumen . . . . .	107
7.2.	Discusión . . . . .	108
7.2.1.	Limitaciones . . . . .	108
7.3.	Trabajo a futuro . . . . .	109
7.3.1.	Adaptaciones necesarias . . . . .	109
7.4.	Aplicaciones . . . . .	110
7.5.	Conclusiones finales . . . . .	111

<b>A. Algoritmo</b>	<b>113</b>
A.1. Diagrama de Flujo GADGET2 . . . . .	114
A.2. Código generado . . . . .	115
A.2.1. sink_feedback.h . . . . .	115
A.2.2. sink_feedback.c . . . . .	117
A.3. Herramientas computacionales adicionales . . . . .	148



# Capítulo 1

## Introducción

Cuando observamos la Vía Láctea, vemos únicamente una parte de nuestra pequeña isla dentro del Universo, una galaxia espiral formada hace aproximadamente  $1.5 \times 10^{10}$  años y que contiene más de  $10^{11}$  estrellas acompañadas de cerca de  $10^9 M_{\odot}$ <sup>1</sup> de gas y polvo.

Nuestra Galaxia comprende un disco con radio  $\sim 25 - 32$  kpc<sup>2</sup> y un espesor efectivo de  $\sim 400 - 600$  pc acompañado de un sistema esférico compuesto de un bulbo de radio  $\sim 2 - 3$  kpc y un halo extendiéndose a más de 30 kpc a partir del centro (Binney & Merrifield, 1998). El Sol reside en el disco galáctico aproximadamente 15 pc arriba del plano medio (Cohen, 1995; Magnani *et al.*, 1996) y a  $\lesssim 8$  kpc del centro galáctico (Groenewegen *et al.*, 2008).

Las estrellas que pertenecen al disco rotan en órbitas casi circulares y su tasa de rotación angular es una función decreciente de su distancia radial, de manera que la velocidad tangencial es casi constante a lo largo de la dirección radial. A la distancia galactocéntrica del Sol, la velocidad de rotación es de  $\simeq 220$  km s<sup>-1</sup> (Kerr & Lynden-Bell, 1986), correspondiendo a un periodo orbital de cerca de 240 millones de años. La dispersión de velocidades de las estrellas en el disco es de  $\sim 10 - 40$  km s<sup>-1</sup> (Mihalas & Binney, 1981), lo cual causa que una estrella tenga pequeñas oscilaciones alrededor de una órbita circular, tanto en el plano galáctico (epiciclos) como en el plano vertical. En

---

<sup>1</sup> $1M_{\odot} = 1.99 \times 10^{33} g$

<sup>2</sup> $1 \text{ pc} = 3.08568 \times 10^{18} \text{ cm}$

contraste, las estrellas presentes en el bulbo y en el halo rotan lentamente y a menudo tienen órbitas muy excéntricas.

El Medio Interestelar (MI) es material formado en su mayoría por gas en estados ionizado, atómico y molecular, en rangos muy amplios de temperaturas, densidades y presiones característicos, además de partículas de polvo, rayos X y rayos cósmicos, inmersos dentro de un campo magnético.

Construir un modelo, incluso aproximado, de la Galaxia implica una gran dificultad. Sin embargo, podemos observar galaxias parecidas a la nuestra (p. ej. M31, la galaxia Andrómeda). Haciendo comparaciones entre observaciones, resultados analíticos y simulaciones numéricas es posible mejorar nuestros modelos con el fin de averiguar el por qué de la estructura y dinámica de la Galaxia o de sus componentes como las estrellas o el gas.

Nuestra Galaxia forma estrellas a una tasa de  $\sim 1 - 3 M_{\odot} \text{ año}^{-1}$  a partir del material gaseoso (en su mayoría hidrógeno) que se encuentra en el MI. Toda la formación estelar en la Galaxia se produce en las llamadas nubes moleculares, que son las regiones más densas del MI. Por otro lado, durante el tiempo de vida de las estrellas (que depende de su masa), la mayoría de su material es retornado al MI a través de vientos estelares y eventos explosivos. Estos procesos, junto con la emisión radiativa de la estrella, pueden afectar de diversas formas su nube madre, como dispersarla comprimiendo los alrededores, induciendo nueva formación estelar (FE). En la actualidad, existen dos líneas principales de pensamiento respecto este proceso llamado *retroalimentación estelar*. La primera considera que mantiene a las nubes en equilibrio y la segunda considera que las dispersa, evaporándolas.

El objetivo principal de esta tesis es la implementación de un algoritmo que permita describir la retroalimentación estelar al MI en el código numérico GADGET2, a fin de permitir la posterior investigación de los efectos de la retroalimentación energética por radiación de fotones de estrellas recién nacidas en su ambiente. En el cap. 2 se revisan primero algunos principios básicos sobre la física del MI y a continuación se revisan las propiedades de las principales componentes del MI. En el cap. 3 se revisan brevemente los principales problemas contemporáneos sobre la FE y los algoritmos numéricos para abordar el problema. En el cap. 4 se describe brevemente el código GADGET2 y las adaptaciones que contiene para describir la física del MI. En

el cap. 5 se describe el algoritmo de Dale *et al.* (2007) y su implementación. En el cap. 6 se presentan pruebas del funcionamiento del algoritmo y se discuten sus limitaciones. Finalmente, en el cap. 7 se presentan algunas conclusiones y se discute el trabajo a futuro.



# Capítulo 2

## El Medio Interestelar

El Medio Interestelar (MI) es el material que existe entre las estrellas dentro de la Galaxia, con una masa de  $10^{10}M_{\odot}$  (un 10 – 15 % de la masa total del disco). Es una mezcla extremadamente diluida de gas, en su mayoría hidrógeno (90.8 % por número, 70.4 % por masa) en estado atómico, molecular ( $H_2$ ) o ionizado; helio ( $\sim 9.1$  % por número,  $\sim 28.1$  % por masa), trazas de elementos más pesados ( $\sim 0.12$  % por número,  $\sim 1.5$  % por masa) y otras moléculas, además de polvo y partículas relativistas inmersos en un campo magnético (Ferrière, 2001).

En galaxias de disco como la nuestra, el MI se extiende más allá del disco estelar a distancias cercanas al doble del tamaño del radio estelar (Binner & Merrifield, 1998). Diferentes componentes del MI tienen diferentes escalas de altura (Ferrière, 2001) y su espesor está determinado por el potencial gravitacional de las estrellas y de cada componente. La presión en el MI es generada por el peso de la columna vertical del gas, disminuyendo con la altura de tal forma que podemos considerarlo como una *atmósfera estratificada* (Boulares & Cox 1990, Ferrière, 2001).

La materia interestelar se manifiesta a través del oscurecimiento, enrojecimiento y polarización de la luz estelar; por la formación de líneas de absorción en el espectro estelar y por varios mecanismos de emisión en el continuo y a longitudes de onda específicas.

## **2.1. Dinámica y termodinámica del Medio Interestelar**

El ciclo de la materia, partiendo del MI a las estrellas y de vuelta al MI, es de vital importancia para el entendimiento de la dinámica de la Galaxia. Las estrellas se forman a partir de un reservorio de material interestelar, cuyas componentes, lejos de estar uniformemente esparcidas a través del espacio interestelar, muestran un contraste de densidad y temperaturas muy grande. Las transiciones electrónicas, rotacionales y vibracionales de sus componentes, constituyen mecanismos de enfriamiento que le permiten perder energía a través de emisión de fotones. Además, en las regiones moleculares, que a su vez son más densas y frías, las parcelas de gas pierden parte de su soporte magnético, por el proceso conocido como “difusión ambipolar”, convirtiéndose en gravitacionalmente inestables y colapsando en estrellas nuevas.

Esta materia, una vez atrapada en el interior de las estrellas pasa por una sucesión de reacciones termonucleares que la enriquecen con elementos pesados. Una parte de esta materia regresa al MI, ya sea en forma continua a través de vientos estelares, o bien en forma de nebulosas planetarias o supernovas. En ambos casos, la inyección de masa al MI es acompañada de fuertes liberaciones de energía, generando movimientos turbulentos que calientan y agitan el MI, contribuyendo a mantener su estructura altamente heterogénea y favoreciendo en ciertos casos el nacimiento de nuevas regiones moleculares propensas a formación estelar.

El efecto neto de los procesos de enfriamiento y calentamiento radiativos en el MI es el de producir un comportamiento termodinámico casi isobárico, constituyendo así un factor importante no sólo al intercambiar materia y energía continuamente dentro de la Galaxia, sino controlando muchas de sus propiedades. La distribución espacial junto con sus características térmicas y químicas determinan la localización donde se forman nuevas estrellas, y éstas a su vez, afectan la estructura total, evolución y dinámica a gran escala de la Galaxia.

### 2.1.1. Ecuaciones

Al considerar al MI como un fluido compresible, magnetizado y sujeto a un campo gravitacional, su comportamiento estará regido por las ecuaciones de la magnetohidrodinámica (MHD), que expresan respectivamente, conservación de masa, momento, energía y flujo magnético. Estas ecuaciones están dadas por:

$$\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho = -\rho \nabla \cdot \mathbf{u}, \quad (2.1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{\nabla P}{\rho} - \nabla \phi + \frac{1}{4\pi\rho} (\nabla \times \mathbf{B}) \times \mathbf{B}, \quad (2.2)$$

$$\frac{\partial e}{\partial t} + \mathbf{u} \cdot \nabla e = -(\gamma - 1)e \nabla \cdot \mathbf{u} + \Gamma - n\Lambda, \quad (2.3)$$

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{u} \times \mathbf{B}), \quad (2.4)$$

donde  $\mathbf{u}$  es el campo de velocidad del gas,  $\rho$  es la densidad de masa,  $e$  es la energía interna por unidad de masa,  $P$  es la presión térmica, dada por la ecuación de estado de gas ideal,

$$P = nkT, \quad (2.5)$$

$k$  es la constante de Boltzmann,  $\mathbf{B}$  es la intensidad el campo magnético,  $n = \rho/(\mu m_H)$  es la densidad numérica,  $\mu$  es el peso promedio por partícula de gas,  $\gamma$  es el cociente entre los calores específicos a presión constante y a volumen constante del fluido,  $m_H$  es la masa del átomo de hidrógeno y  $\phi$  es el potencial gravitacional, que satisface la ecuación de Poisson,

$$\nabla^2 \phi = 4\pi G\rho. \quad (2.6)$$

La ecuación que relaciona la energía interna con la temperatura es

$$e = c_V T = \frac{1}{\gamma - 1} \frac{P}{\rho} \quad (2.7)$$

donde  $c_V$  es el calor específico a volumen constante. Finalmente, en la ec. (2.3),  $\Lambda$  es la función de enfriamiento y  $\Gamma$  es la función de calentamiento. Tanto  $\Gamma$  como  $n\Lambda$  están dadas en  $\text{erg s}^{-1} \text{g}^{-1}$

El lado izquierdo de las tres primeras ecuaciones representa la tasa de cambio *total*<sup>1</sup> de la respectiva propiedad del gas de un elemento de fluido, debido tanto a su cambio en la posición actual como al cambio debido al movimiento del elemento con velocidad  $\mathbf{u}$  a una nueva posición (segundo término, *advectivo*). Los lados derechos de las ecuaciones representan las *fuentes y sumideros* responsables del cambio de la propiedad del gas. Específicamente:

- En la ecuación de conservación de la masa, el término fuente es debido a un campo de velocidades convergente a la posición del elemento de fluido, incrementando la densidad.
- En la ecuación de conservación del momento, el fluido tiende a moverse a los sitios de baja presión y a los sitios en donde el potencial gravitacional es más negativo. Otro término fuente es el de la fuerza de Lorentz ejercida por el campo magnético.
- En la ecuación de conservación de energía, el primer término del lado derecho es el trabajo adiabático  $PdV$  sobre la parcela de fluido y el segundo, es el *enfriamiento neto* que resulta de los procesos de enfriamiento y calentamiento.

### Velocidad del sonido en el gas

La velocidad del sonido  $c_s$  es un concepto fundamental en la dinámica de flujos compresibles, como es el caso del MI, y es esencialmente una medida de la velocidad característica de las moléculas en el medio. Las ondas de sonido en un fluido se propagan esencialmente a esta velocidad, ya que es aquella a la que la información sobre perturbaciones en el medio se propaga por colisiones entre moléculas. En particular, la expansión libre de una parcela de gas ocurre a velocidades del orden de  $c_s$ . En general, la velocidad del sonido está dada por:

$$c_s = \sqrt{\gamma \frac{P}{\rho}} = \sqrt{\frac{\gamma kT}{\mu m_H}}. \quad (2.8)$$

<sup>1</sup>La derivada total de una función del tiempo y la posición está dada por  $d/dt = \partial/\partial t + \mathbf{u} \cdot \nabla$ .

Recordando que para un gas isotérmico  $\gamma = 1$ , tenemos:

$$c_{s, iso} = \sqrt{\frac{kT}{\mu m_H}}. \quad (2.9)$$

### 2.1.2. Termodinámica

El MI es un medio cuyas componentes se encuentran en distintos rangos de temperatura, presión y densidad; por tanto, es importante su estudio desde el punto de vista termodinámico, para lo cual nos enfocaremos ahora en la ecuación de la energía, (2.3).

#### Enfriamiento y Calentamiento

Las tasas de calentamiento  $\Gamma$  y enfriamiento  $\Lambda$  incluyen una gran cantidad de procesos que operan a nivel atómico y molecular (ver fig. 2.1) y cuyo resultado final es que la tasa de cambio neta de la temperatura del gas dependa fuertemente de su temperatura, densidad y metalicidad (ver fig. 2.2). Observemos entonces que el enfriamiento neto del gas se define entonces como:

$$\mathcal{L} \equiv n\Lambda - \Gamma \quad (2.10)$$

#### Tiempo de enfriamiento $\tau_{enf}$

Debido a que las nubes emiten radiación y por lo tanto se enfrían, es útil calcular el tiempo que les lleva enfriarse. Este tiempo en esencia depende de la energía interna del gas  $e$  y de la tasa a la que emite energía sujeto solamente al enfriamiento neto del gas. Por lo tanto, podemos definir el tiempo de enfriamiento como:

$$\tau_{enf} \approx \frac{e}{|n\Lambda - \Gamma|}. \quad (2.11)$$

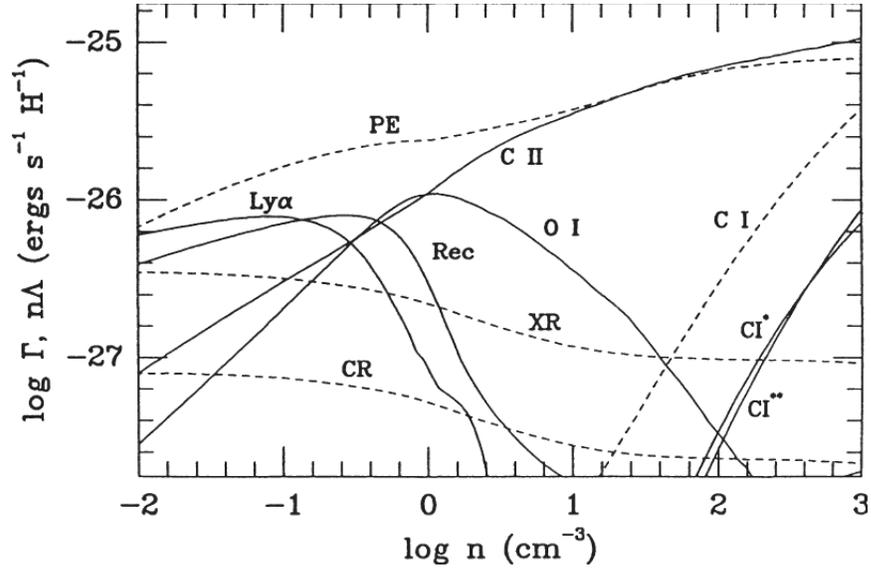


Fig. 2.1: Procesos de enfriamiento y calentamiento que actúan sobre el gas atómico (neutro) en el MI. Las líneas a rayas indican procesos de calentamiento, las continuas de enfriamiento. PE: calentamiento fotoeléctrico sobre granos de polvo y moléculas de hidrocarburos poliaromáticos (PAHs); XR: calentamiento por rayos X; CR: calentamiento por rayos cósmicos; CI: calentamiento por fotoionización del carbono neutro; Ly $\alpha$ : enfriamiento por emisión Ly $\alpha$ ; Rec: enfriamiento por recombinación sobre granos de polvo y PAHs; CI\*: enfriamiento por emisión de la línea de estructura fina a 609  $\mu\text{m}$  del CI; CI\*\*: enfriamiento por emisión de la línea de estructura fina a 370  $\mu\text{m}$  del CI; OI: enfriamiento por emisión de estructura fina del OI. Tomada de Wolfire *et al.* (1995).

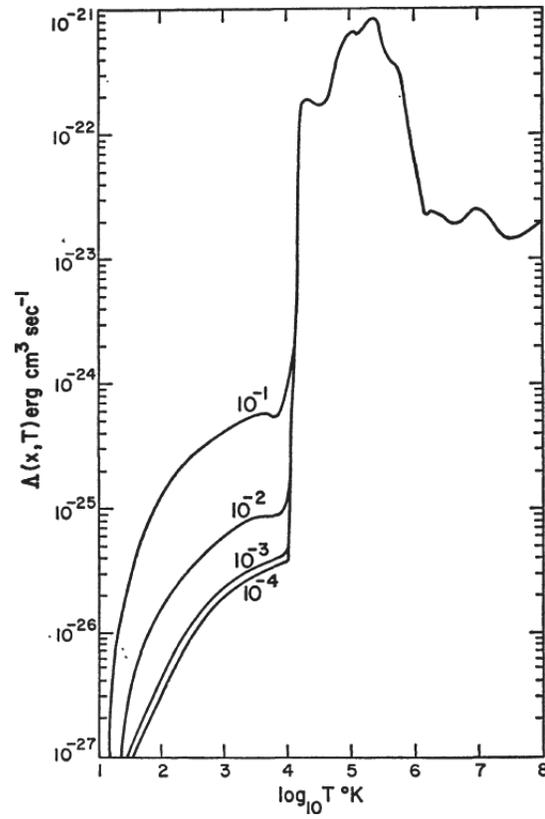


Fig. 2.2: Función de enfriamiento calculada por Dalgarno & McCray (1972), como función de la temperatura y de la fracción de ionización  $x \equiv n_e/n_H$  (números junto a las curvas).

### Inestabilidad térmica

Al observar la gráfica de los procesos de enfriamiento y calentamiento es notable que los procesos de enfriamiento son los mayormente responsables del comportamiento termodinámico del MI, pues varían fuertemente con la densidad. Una peculiaridad del medio atómico es el fenómeno conocido como *inestabilidad térmica*, que cualitativamente puede entenderse como la tendencia del MI a alejarse de un cierto rango de densidades y temperaturas al sufrir alguna perturbación.

Para explicar esto, consideremos un régimen en el que el fluido está en reposo y sin intercambio de energía. En este caso, la ecuación (2.3) se reduce a la condición de *equilibrio térmico*:

$$\Gamma = n\Lambda. \quad (2.12)$$

La ecuación de estado del gas ideal, ec. (2.5), relaciona a las variables termodinámicas  $\rho$ ,  $P$  y  $T$ , de forma que sólo dos de ellas son independientes. En una región donde se cumple que

$$\left(\frac{\partial\Lambda}{\partial T}\right)_\rho < 0, \quad (2.13)$$

un aumento en la temperatura  $T$  implica que la función de enfriamiento  $\Lambda$  disminuirá, con lo cual  $\Gamma > n\Lambda$ , causando un mayor aumento en  $T$  y disparando el sistema hacia una  $T$  aún mayor. Este es el *modo isocórico* de la inestabilidad térmica (IT) y sólo puede ocurrir a escalas de tamaño muy grandes, pues la región en la que esto ocurre se sobre- o des-presuriza y se necesita que el tiempo de cruce sónico para la región (el tiempo que el gas de la región tarda en expandirse [o contraerse] necesario para igualar presiones) sea mucho mayor que el tiempo de enfriamiento, a fin de que la región tenga tiempo de enfriarse antes de comprimirse o expandirse debido al cambio en su presión. En el MI, este modo es relevante en regiones muy calientes, con  $T \gtrsim 10^5 K$ , ya que como se observa de la fig. (2.2),  $\frac{\partial\Lambda}{\partial T} < 0$  para  $10^5 < T < 10^6 K$ .

Otro modo muy importante de la IT es el *modo isobárico*, que se verifica en regiones donde

$$\left(\frac{\partial\Lambda}{\partial\rho}\right)_P > 0. \quad (2.14)$$

Estos dos procesos descritos, producen la segregación del gas en dos fases, una caliente ( $T \gtrsim 10^6$  K) y una tibia ( $T \lesssim 10^5$  K).

### Origen de las fases

Un equilibrio multifase únicamente es posible si el gas es térmicamente inestable. Si para una presión dada existe únicamente una temperatura en la cual el calentamiento y el enfriamiento están en equilibrio, entonces el gas será homogéneo, es decir, tendrá una única fase.

Field, Goldsmith & Habing (FGH, 1969), en su *modelo de dos fases* del MI atómico, mostraron que es posible que dos fases térmicamente estables coexistan a la misma presión, separadas por una fase térmicamente inestable. En su modelo, sin embargo, consideraron el calentamiento del gas por rayos cósmicos como la principal fuente de calentamiento. Posteriormente se encontró que la habían sobre-estimado. Sin embargo, la radiación UV de fondo producida por estrellas masivas, que libera electrones energéticos por efecto fotoeléctrico en el polvo, sí es suficiente (Wolfire *et al.* 1995). Así pues, la existencia de dos fases estables y una inestable en un cierto intervalo de densidades y regiones, implica que el MI tenderá a “asentarse” en las fases estables y a alejarse de la inestable, produciendo espontáneamente un sustrato de gas tibio difuso, poblado por “islas” de gas frío y denso. En efecto, la densidad y temperatura de la fase inestable ( $T \sim 2000$  K,  $n \sim 1$  cm<sup>-3</sup>) corresponden a la densidad y temperatura promedio del MI en la Vía Láctea. Las fases estables corresponden a la fase fría ( $T \sim 10^2$  K,  $n \sim 50$  cm<sup>3</sup>, *Medio Neutro Frío*, MNF) y tibia ( $T \sim 10^4$  K,  $n \sim 0.5$  cm<sup>3</sup>, *Medio Neutro Tibio*, MNT) del hidrógeno neutro atómico en el MI.

Este modelo es para un MI atómico en reposo, sin considerar eventos violentos como supernovas y regiones de gas ionizado (*regiones HII*) que continuamente agitan el medio y lo hacen turbulento. Cox & Smith (1974) mostraron que los remanentes de supernovas, al inyectar energía dentro del medio, crean grandes volúmenes de gas caliente ( $T \sim 10^6$  K,  $n \sim 0.01$  cm<sup>-3</sup>) dando lugar a una tercera fase de gas caliente (McKee & Ostriker, MO, 1977). Este nuevo *modelo de tres fases* desde entonces se ha constituido como modelo referencia del MI. En resumen, podemos considerar al MI como un medio *multifase* en el que coexisten distintas fases, aproximadamente a la misma presión térmica,

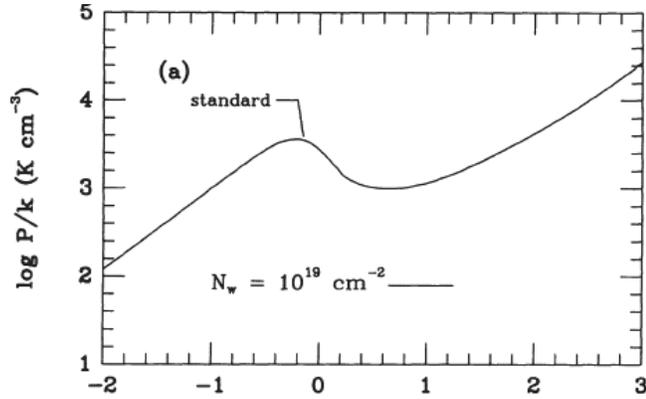


Fig. 2.3: La presión térmica  $P/k$  contra la densidad de hidrógeno  $n$ , correspondiente a una nube de gas con masa  $M \sim 1M_{\odot}$  en el MNT y con densidad columnar  $N = 10^{19} \text{ cm}^{-2}$ . Tomada de Wolfire *et al.* 1995.

desde el gas ionizado caliente hasta el gas atómico denso y frío.

Ambos modelos, el de dos y el de tres fases, son incompletos en el sentido de que omiten al gas molecular, una de las componentes del MI más importantes, pues es allí donde se forman las estrellas. La mayor parte de este gas en la Galaxia está en *Nubes Moleculares Gigantes* (NMGs), las cuales están gravitacionalmente ligadas y por lo tanto no están a la misma presión que el MI que las rodea. Los movimientos dentro de ellas son altamente supersónicos y por lo tanto, la presión térmica de las NMGs es una pequeña fracción del total. Sin embargo, Falgarone & Puget (1986) sugirieron la posibilidad de que exista en su interior un medio de dos fases, con HI frío y tibio, además de *grupos* moleculares coexistiendo a una presión térmica comparable a la del MI difuso.

### 2.1.3. Regiones HII

Pasamos ahora a discutir las regiones HII, que son las regiones ionizadas alrededor de las estrellas masivas, ya que estas regiones tienen un papel muy importante como fuentes de energía térmica y cinética en las nubes moleculares y son las regiones que se formarán en las simulaciones numéricas al

incluir el algoritmo que se describe en el cap. 5 al código GADGET2.

Después de que una estrella ha alcanzado la secuencia principal y emite fotones ultravioleta a una tasa constante, estos fotones tienen dos efectos en el medio. Primeramente, el gas alrededor de la estrella es ionizado y los fotones ultravioleta excedentes son absorbidos en una región pequeña (de unas cuantas veces el camino libre medio de los fotones en el hidrógeno neutro  $l \sim 10(\alpha_0 n)^{-1}$ ) llamada *frente de ionización* (FI) dentro de la cual el grado de ionización y en particular el flujo de fotones decrece rápidamente. Por lo tanto, el FI separa al gas en dos regiones: una interior de gas ionizado y una exterior de gas neutro. Dentro de la región ionizada, los iones y los electrones libres siguen recombinándose antes de ser separados otra vez por nuevos fotones ionizantes provenientes de la estrella.

El segundo efecto producido por los fotones es que este proceso es acompañado por un calentamiento neto del gas interestelar al transferir su exceso energía (con respecto al potencial de ionización) a los electrones, situando al gas ionizado a temperatura de equilibrio de  $10^4$  K dada por el balance entre el calentamiento fotoeléctrico y el enfriamiento radiativo que a su vez depende de la densidad y la metalicidad del gas. Este valor de la temperatura es un factor de 100 veces más alto que la temperatura del gas neutro de  $\sim 100$  K mientras que la densidad numérica de partículas aumenta en un factor de dos al separar los átomos en protones y electrones debido a la ionización de gas. Por lo tanto, la presión térmica del gas ionizado es 200 veces mayor que la del gas neutro de su ambiente dando origen a una expansión casi libre del gas. Sin embargo, como la tasa de recombinación depende de la densidad de protones y electrones, al expandirse el gas ionizado éstas disminuyen provocando que el flujo de fotones alcance a ionizar una masa mayor. El proceso continúa hasta un estado final de equilibrio de presión con sus alrededores (o, más comúnmente, hasta que la estrella explota como supernova).

Esto es considerado como la descripción clásica de la formación de regiones HII. Resolviendo numéricamente la ecuación de transporte para los fotones ionizantes del hidrógeno, junto con la condición de equilibrio ionización-recombinación en un medio atómico de densidad constante, Strömgren (1939) mostró que existen regiones de gas casi completamente ionizado alrededor de estrellas calientes tipo O, con una frontera delgada con el gas casi completamente neutro del exterior. Strömgren indicó que esos fotones tienen más

energía que la necesaria para ionizar el hidrógeno, y por lo tanto deben ser una fuente importante para el calentamiento del gas. Posteriormente, Spitzer & Savedoff (1950) mostraron que este exceso de energía causa que estas regiones sean calentadas a una temperatura de  $\sim 10^4$  K.

En la práctica, este problema se complica por varios hechos importantes: a) Las nubes de gas molecular, en lugar del gas atómico, son los principales sitios de formación estelar en la actualidad; b) las estrellas jóvenes poseen vientos poderosos que ayudan a dicho proceso de expansión a alejarse de su material natal, y c) el medio externo no es uniforme, sino que tiene importantes fluctuaciones de densidad y temperatura.

Como puede intuirse, para estudiar las regiones HII es importante determinar la fracción de ionización del hidrógeno en la región ionizada, así como su tamaño inicial y posteriormente analizar su evolución en el sentido de la expansión. A continuación se describen algunos conceptos importantes de la teoría de formación de regiones HII.

### La ionización del hidrógeno

Las estrellas masivas emiten grandes cantidades de fotones ionizantes que actúan sobre los átomos neutros del hidrógeno. Estos átomos pueden encontrarse en cualquier nivel excitado debido al proceso de recombinación, pero el tiempo de vida en este estado es muy corto ( $\sim \mu$ s). Por lo tanto, la probabilidad de ionizar a un átomo en este estado es despreciable, y podemos considerarlos prácticamente a todos en el estado base.

Con el propósito de determinar la fracción de ionización del hidrógeno seguiremos la discusión de Dyson (1980). Consideremos una nube de gas uniforme con densidad numérica inicial  $n_0$ . Sea  $dV$  un elemento de volumen del gas situado a una distancia  $r$  de una estrella, con  $J$  fotones ionizantes pasando a través de él. La tasa de ionización está dada por:

$$\dot{N}_I = \alpha_0 n_H J, \tag{2.15}$$

en donde  $n_H$  es la densidad numérica de átomos de hidrógeno neutro. La cantidad  $\alpha_0$  representa el área efectiva de un átomo para captura de un fotón incidente y es llamada la sección transversal de fotoionización. Estrictamente

hablando, ésta depende de la energía del fotón incidente y disminuye cuando la energía del fotón se incrementa. Sin embargo, para nuestro propósito es suficiente considerarla aproximadamente constante correspondiente al valor que tiene para un fotón de energía  $h\nu$  igual al potencial de ionización del hidrógeno,  $\alpha_0 = 6.8 \times 10^{-18} \text{ cm}^2$ .

Sea  $x$  la fracción de ionización del hidrógeno. Entonces, el número de electrones disponibles dependerá de la densidad total como  $n_e = xn$ , donde  $n = n_p + n_H$  y  $n_p$  es la densidad de protones. Para una nube neutra de hidrógeno,  $n_e = n_p$ , con lo cual:

$$n_H = (1 - x)n. \quad (2.16)$$

Por otra parte, la tasa de recombinación por unidad de volumen es proporcional al producto de las densidades numéricas de protones y electrones y también depende de la temperatura del electrón  $T_e$  por dos razones. La primera es que la probabilidad de las recombinaciones de un electrón en un nivel de energía  $n$  dependen de la energía del electrón. La segunda es que la tasa a la cual un protón encuentra un electrón depende de la distribución de velocidades del electrón, es decir, de la temperatura del electrón. Por lo tanto, la tasa de recombinación volumétrica puede expresarse como:

$$\dot{\mathcal{N}}_R = \sum_{n=2}^{\infty} \dot{\mathcal{N}}_n \equiv n_e n_p \beta_2(T_e), \quad (2.17)$$

en donde  $\beta_2(T_e)$  es el coeficiente de recombinación al nivel  $n = 2$ . Aquí no se considera el nivel base debido a que se supone que una recombinación directa a dicho nivel es balanceada por una ionización inmediata. Una buena aproximación numérica de este coeficiente es  $\beta_2 = 2 \times 10^{-16} T_e^{-3/4}$ .

Igualando las tasas de recombinación (ec. 2.17) y de ionización (ec. 2.15) por unidad de volumen, resulta la *ecuación de balance de ionización*:

$$\dot{\mathcal{N}}_R = \dot{\mathcal{N}}_I, \quad (2.18)$$

es decir,

$$x^2 n^2 \beta_2(T_e) = \alpha_0 (1 - x) n J, \quad (2.19)$$

o, equivalentemente,

$$\frac{x^2}{1 - x} = \frac{J}{n} \frac{\alpha_0}{\beta_2(T_e)}. \quad (2.20)$$

## 2.1 Dinámica y termodinámica del Medio Interestelar *El Medio Interestelar*

Para estimar  $J$  en una región típica de la nube alrededor de la estrella, suponemos que su dilución geométrica sigue una ley de inverso cuadrado:

$$J = \frac{S_*}{4\pi r^2}, \quad (2.21)$$

en donde  $S_*$  es la tasa a la cual la estrella emite fotones ionizantes (ver tabla (2.1)).

Tipo estelar	$T_{effec}$ [K]	$M/M_\odot$	$L/L_\odot$	$S_*$ [s <sup>-1</sup> ]	$J$ [cm <sup>-2</sup> s <sup>-1</sup> ]
O5	42000.0	60.0	51.0000	499000.0	$4.2624 \times 10^{11}$
O6	39500.0	37.0	17.4000	324000.0	$1.4542 \times 10^{11}$
O8	36500.0	23.0	3.9000	147000.0	$3.2595 \times 10^{10}$
B0	30000.0	17.5	0.4300	32500.0	$3.5938 \times 10^9$

Tabla 2.1: Valores de característicos para distintos tipos de estrellas.  $J$  es calculado a  $r = 1$  pc.

Si consideramos el valor típico de la densidad en una nube molecular ( $n = 100 \text{ cm}^{-3}$ ) y el número de fotones ionizantes por unidad de tiempo para una estrella tipo O8 en secuencia principal ( $S_* = 3.9 \times 10^{48} \text{ s}^{-1}$ ), a una distancia  $r = 1$  pc, con  $T_e = 10^4$  K, obtenemos  $J \approx 8.36 \times 10^{10} \text{ cm}^{-2} \text{ s}^{-1}$  y  $\beta_2 = 2 \times 10^{-13} \text{ cm}^3 \text{ s}^{-1}$ . Sustituyendo en la ecuación de balance de ionización ec. (2.18) resulta:

$$\frac{x^2}{1-x} \approx 2.84 \times 10^4. \quad (2.22)$$

Resolviendo la ecuación cuadrática bajo la condición ( $0 \leq x \leq 1$ ) se obtiene la solución única:

$$x = \frac{1}{2} \left( \sqrt{c_x^2 + 4} - c_x \right) \approx 1, \quad (2.23)$$

donde  $c_x$  es el valor en el lado derecho de la ec. (2.22), con lo cual hemos llegado a la conclusión de que el gas dentro de la región HII está casi completamente ionizado. Por lo tanto, en la ecuación de balance de ionización podremos suponer  $x \approx 1$ , es decir, las regiones HII están casi completamente ionizadas.

### El tamaño de la región HII

El volumen de gas que la estrella puede ionizar está limitado al volumen en el cual la tasa de recombinación total es igual a la tasa a la cual la estrella emite fotones ionizantes. Es decir, si la región HII tiene un radio  $R_{ini}$ , la condición del balance de ionización para la región está dada por:

$$\left(\frac{4}{3}\pi R_{ini}^3\right) (n_H n_e \alpha) = N_*, \quad (2.24)$$

donde  $N_*$  es el número de fotones ionizantes (aquéllos con frecuencias más allá del límite de Lyman) por unidad de tiempo,  $n_H$  y  $n_e$  son las densidades numéricas de protones y electrones libres en la región HII,  $\alpha$  es el coeficiente de recombinación al nivel 2 y superiores para un gas de hidrógeno libre de polvo y es tomado como constante ( $\alpha = 3 \times 10^{-13} \text{ cm}^{-3}$ ) si la temperatura de la región HII permanece fija. Por lo tanto, el tamaño de la región HII es:

$$R_S = \left(\frac{3N_*}{4\pi n_H n_e \alpha}\right)^{\frac{1}{3}}, \quad (2.25)$$

y es conocido como *radio de Strömgren*.

#### 2.1.4. Evolución de las regiones HII

Conforme la estrella ioniza el gas neutro el FI se propaga hacia afuera hasta un radio de equilibrio (entre las tasas de ionización y recombinación) igual al radio de Strömgren inicial dado por la ec. (2.25).

Si suponemos que el gas que rodea a la estrella está inicialmente en reposo con respecto a un marco de referencia situado en la estrella, es claro que la evolución de la región HII está dividida en dos fases: la fase de formación y la fase de expansión. A continuación se menciona en forma detallada su física.

#### La fase de formación

Una propiedad básica del FI es la velocidad a la cual se mueve. Para determinarla, consideremos a  $J$  como el número de fotones en el continuo de

## 2.1 Dinámica y termodinámica del Medio Interestelar *El Medio Interestelar*

Lyman que cruzan el área unitaria del FI por segundo y supongamos que en algún tiempo  $t$  el FI está situado a una distancia  $R$  de la estrella, con el gas completamente ionizado detrás del FI y neutro delante de éste. Cuando el FI se ha movido desde una distancia  $R$  hasta  $R+dR$ , ya han arribado suficientes fotones para ionizar todos los átomos neutros entre estas dos posiciones, esto es,  $Jdt = n_0 dR$ . Por lo tanto, la velocidad del FI respecto de las partículas del gas neutro es:

$$\frac{dR}{dt} = \frac{J}{n_0}. \quad (2.26)$$

Debido a que el frente de ionización es esférico, debe existir un factor de dilución geométrico, de lo contrario, las recombinaciones continuas producirían átomos neutros y por lo tanto no podríamos emplear la hipótesis implícita de que únicamente un fotón es necesario para ionizar cada átomo neutro. Así, en términos de la conservación de fotones, la tasa a la cual los fotones son emitidos por la estrella es igual a la tasa a la cual arriban al frente mas la tasa a la cual son absorbidos por el gas neutro:

$$S_* = 4\pi R^2 J + \frac{4}{3}\pi R^3 n_0^2 \beta_2. \quad (2.27)$$

Despejando  $J$  y sustituyendo en la expresión para la velocidad del FI (ec. 2.26) obtenemos:

$$\frac{dR}{dt} = \frac{S_*}{4\pi R^2 n_0} - \frac{1}{3} R n_0 \beta_2. \quad (2.28)$$

Como puede verse, esta velocidad disminuye conforme  $R$  aumenta, pero la expansión continúa mientras la presión del gas ionizado sea mayor que la del gas neutro. Por otra parte, si consideramos que la velocidad del sonido ( $c_s$ ) es la velocidad a la cual el gas reacciona a las perturbaciones en el medio, entonces la velocidad del FI no podrá ser mayor a ésta, con lo cual:

$$\dot{R} = \min \left\{ c_i, \frac{1}{4\pi n_H R_i^2} \left[ N_* - \frac{4\pi}{3} R_i^3 n_H^2 \alpha \right] \right\}, \quad (2.29)$$

será la velocidad física permisible del FI.

Para determinar el comportamiento del FI durante esta etapa, podemos expresar las ecuaciones en términos de variables adimensionales con el fin de

encontrar sus valores característicos. Esto se logra empleando las relaciones:

$$\begin{aligned} \lambda &= \frac{R}{R_S} & V_R &= \frac{R_S}{t_R}, \\ \tau &= \frac{t}{t_R} & \dot{\lambda} &= \frac{(\frac{dR}{dt})}{V_R}, \end{aligned} \quad (2.30)$$

en donde  $t_R = (n_0\beta_2)^{-1}$  el tiempo de recombinación para el hidrógeno. La ec. (2.28) se convierte entonces en:

$$\dot{\lambda} = \frac{1}{3}(1 - \lambda^3)/\lambda^2, \quad (2.31)$$

cuya solución es:

$$\lambda = (1 - e^{-\tau})^{1/3}. \quad (2.32)$$

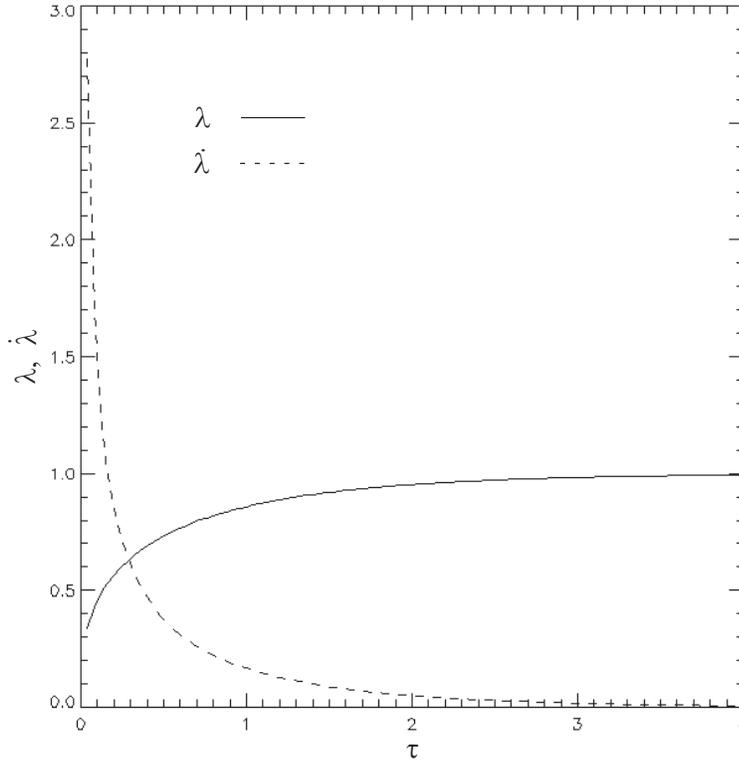


Fig. 2.4: Evolución del FI hasta  $\tau = 4t_R$

De aquí se desprenden algunas conclusiones importantes (ver fig. 2.4):

- I El FI alcanza la posición del radio de Strömgren (que depende de la densidad del medio), únicamente después de un tiempo grande. Sin embargo,  $R$  está dentro de unos pocos porcientos de  $R_S$  a tiempos mayores que  $t_R$ .
- II Hasta que  $R$  es muy cercano a  $R_S$ , el radio de la región ionizada aumenta a una velocidad mucho mayor que la velocidad característica del gas ( $c_i$ ). Por lo tanto, el gas ionizado no puede moverse apreciablemente durante esta fase y la densidad del gas permanece prácticamente sin cambio ( $n_2 \approx n_0$ ).
- III Para  $t \gtrsim t_R$ , el FI rápidamente reduce su velocidad.

### La fase de expansión

Como hemos visto, la expansión de la región III, en principio, podría cesar cuando el gas ionizado caliente alcance la presión de equilibrio con el gas neutro frío de sus alrededores. Es decir, la *condición de equilibrio de presión* en el estado final de la evolución es:

$$2n_f k T_i = n_0 k T_n, \quad (2.33)$$

donde los subíndices  $i, n, f$  indican ionizado, neutro y final respectivamente. El factor 2 del lado izquierdo se debe a que el número de partículas se duplica por el efecto de la fotoionización. De aquí, la densidad final del gas será:

$$n_f = \left( \frac{T_n}{2T_i} \right) n_0. \quad (2.34)$$

Por otra parte, la esfera de gas ionizado seguirá absorbiendo todos los fotones producidos por la estrella y el radio final de la esfera será:

$$R_f = \left( \frac{2T_i}{T_n} \right)^{2/3} R_S. \quad (2.35)$$

Así, el cociente de la masas contenido en la esfera resulta ser:

$$\frac{M_f}{M_S} \equiv \frac{R_f^3 n_f}{R_S^3 n_0} = \frac{2T_i}{T_n}. \quad (2.36)$$

Por lo tanto, empleando los valores característicos usados anteriormente,  $n_f/n_0 \approx 0.005$ ,  $R_f/R_S \approx 34$ ,  $M_f/M_S \approx 200$ . Evidentemente, la esfera de Strömgen inicial contiene únicamente una pequeña fracción de la masa que la estrella puede ionizar en la etapa final de la evolución de la región HII. Debido a que la velocidad de expansión de la esfera ionizada es altamente supersónica con respecto a la velocidad del sonido inicial ( $c_i/c_n \gtrsim 10$ ), al expandirse crea una onda de choque que comprime el gas neutro delante del FI, incrementando su densidad y desplazándolo de su posición original, por lo que el FI reduce su velocidad significativamente durante la evolución y de igual manera reduce significativamente la densidad del gas dentro de la esfera.

A continuación describimos la expansión de la región HII, ilustrada en la fig. (2.5).

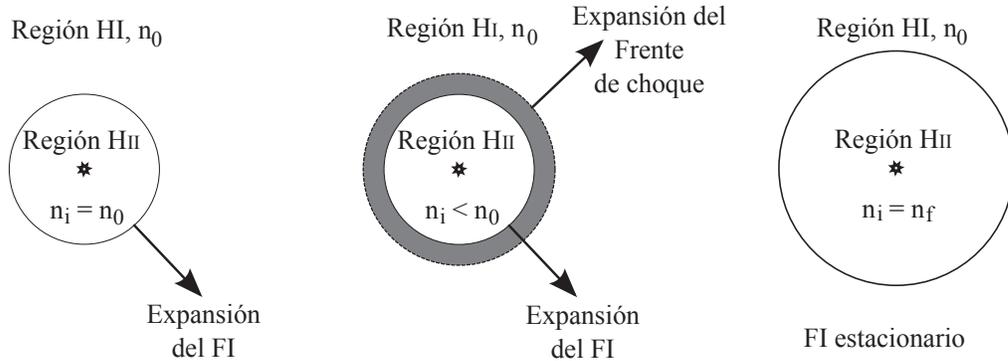


Fig. 2.5: Esquema de la evolución de una region HII en expansión. a) El estado inicial, b) la expansión con choque en el gas neutro, c) el estado final de equilibrio.

En el primer estado, cuando  $R \lesssim R_S$ , el gas ionizado está en reposo a la misma densidad ( $n_0$ ) de sus alrededores y el movimiento está descrito por las ecuaciones (2.31) y (2.32). En la segunda fase ( $R \gtrsim R_S$ ), se ha generado una onda de choque por la expansión de la burbuja de gas caliente que se mueve directamente dentro del material neutro.

Para hacer el análisis de la expansión, es conveniente hacer diversas hipótesis:

- I) La capa del gas neutro chocado es delgada debido a la compresión a través del choque.  $R$  y  $dR/dt$  se refieren a la onda de choque y al FI (es decir, éstos están acoplados). Es importante notar que la velocidad del FI respecto al gas neutro en la onda de choque es pequeña comparada con su velocidad respecto de un marco de referencia fijo.
- II) La presión detrás de la onda de choque se supone uniforme, pero dependiente del tiempo en ambos gases (ionizado y neutro). Para que esto se cumpla, el tiempo de viaje del sonido a través de las regiones neutra y ionizada debe ser menor que la escala de tiempo para que ocurra un cambio sustancial en la configuración total. Esto es una buena aproximación para una capa neutra delgada, pero no para una zona ionizada, al menos en las primeras etapas de la evolución, en las que la esfera se expande con una velocidad comparable a  $c_i$ . Sin embargo, ésta aproximación se hace progresivamente mejor. Debido a que la temperatura del gas ionizado varía poco, esta hipótesis implica una densidad uniforme  $n_i$  en el gas ionizado.
- III) La onda de choque es fuerte, es decir, se mueve de manera altamente supersónica dentro del material neutro. Sin embargo, se debilita con el tiempo.
- IV) La tasa de recombinación en el gas ionizado balancea la tasa de emisión de fotones UV. Estrictamente hablando, este balance nunca puede ser satisfecho exactamente, pues una fracción de los fotones deben ser usados en la ionización “fresca” del gas debido a que la masa ionizada está creciendo continuamente con el tiempo. Sin embargo, se puede demostrar que esta fracción es pequeña.
- V) El gas neutro delante del choque está en reposo. Entonces,  $dR/dt$  es la velocidad del choque relativo al gas neutro y al marco de referencia fijo de la estrella.

Tomando en cuenta éstas hipótesis, podemos establecer un modelo simple para la expansión de la región HII en el que la presión detrás de la onda de choque,  $P_s$ , es igual a la presión  $P_i$  en el gas ionizado:

$$P_i = 2n_i k T_e = n_i m_H c_i^2. \quad (2.37)$$

Por otra parte,  $P_S$  está relacionada con la velocidad del choque,  $V_s$ , relativa al gas que se encuentra adelante:

$$P_s = \epsilon \rho_0 V_s^2, \quad (2.38)$$

donde  $\epsilon = 3/4$  si el choque es adiabático y  $\epsilon = 1$  si es isotérmico. El enfriamiento en el gas neutro chocado puede ser extremadamente efectivo por lo que podemos suponer que el choque es isotérmico, con lo cual:

$$P_s = P_i = n_0 m_H \dot{R}^2, \quad (2.39)$$

donde  $\dot{R} = dR/dt = V_s$ . Por lo tanto, de las ecuaciones (2.37) y (2.39)

$$\dot{R}^2 = \left( \frac{n_i}{n_0} \right) c_i^2. \quad (2.40)$$

La cuarta hipótesis implica que

$$S_* = \frac{4}{3} \pi n_i^2 \beta^2 R^3. \quad (2.41)$$

Usando las ecuaciones (2.25), (2.40) y (2.41), obtenemos

$$R^{3/2} \dot{R}^2 = c_i^2 \left( \frac{3S_*}{4\pi n_0^2 \beta^2} \right)^{1/2} = c_i^2 R^{3/2}. \quad (2.42)$$

Luego, usando las variables adimensionales:

$$\lambda = \frac{R}{R_S}, \quad N = \frac{c_i t}{R_S}, \quad (2.43)$$

podemos reescribir la ecuación anterior en una forma más simple

$$\lambda^{3/4} \frac{d\lambda}{dN} = 1, \quad (2.44)$$

y suponiendo que el tiempo para formar la esfera inicial de Strömngren es mucho menor que el tiempo de vida de la región HII, es decir,  $N = 0$  cuando  $\lambda = 1$ . Usando esta condición, obtenemos la solución

$$\lambda = \left( 1 + \frac{7}{4} N \right)^{4/7}, \quad (2.45)$$

o bien,

$$\frac{d\lambda}{dN} = \dot{\lambda} = \left(1 + \frac{7}{4}N\right)^{-3/7}. \quad (2.46)$$

Esto implica que la velocidad de FI es  $\dot{R} = c_i$  en  $N = 0$ .

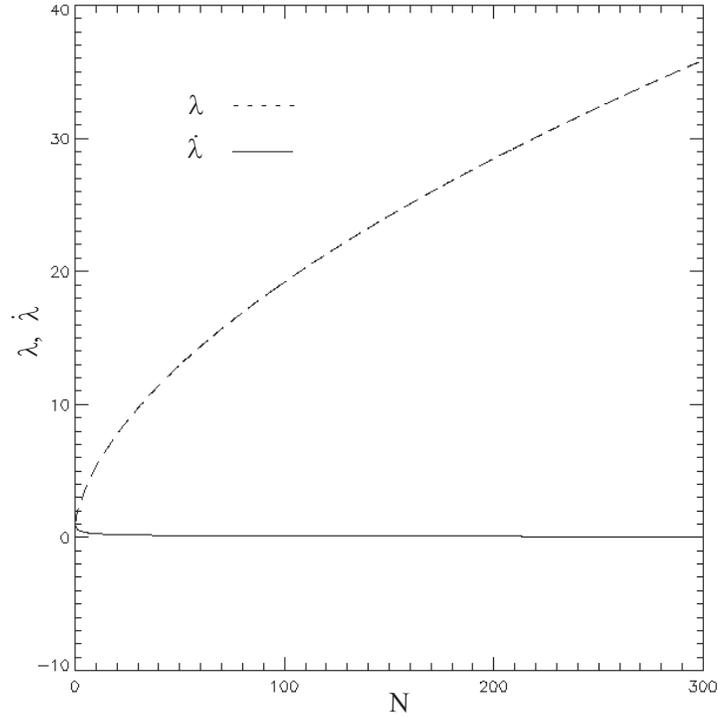


Fig. 2.6: Evolución del FI durante la fase de expansión.

Revirtiendo el cambio de variable para obtener la tasa de crecimiento  $R_i$  de la región HII, obtenemos

$$R_i = R_{ini} \left[1 + \frac{7 c_s (t - t_0)}{4 R_{ini}}\right]^{4/7}, \quad (2.47)$$

donde  $c_s$  es la velocidad del sonido isotérmica del gas ionizado y  $t_0$  es el tiempo en el cual el FI alcanzó el radio  $R_{ini}$ .

Si consideramos el tiempo de la región HII en la fase de expansión  $t_e$ , a partir de la fig. (2.6) podemos observar que  $t_e \gtrsim R_S/c_i$ , entonces  $t_e/t_R \gtrsim 0.4n_0^{1/3}$ , por lo tanto, usando los valores característicos de  $R_S$  y  $t_R$  obtenemos  $t_e/t_R \gg 1$ . Es decir, el tiempo que la región HII pasa en la fase de formación es mucho menor que el que pasa en la fase de expansión.

### **La fase de recombinación**

La evolución de las regiones HII durante la fase de recombinación depende de los detalles de cómo el flujo de radiación ionizante cambia con el tiempo una vez que la estrella sale de la secuencia principal. La reducción de fotones ionizantes causa que el FI ceda, supersónicamente si la caída en el flujo es suficientemente rápida. El gas neutro, nuevamente recombinado, sigue expandiéndose como resultado de su inercia. Durante esta fase, la región HII tiene la apariencia de una esfera de gas ionizado expandiéndose, rodeada por un cascarón geoméricamente grueso de gas neutro expandiéndose ( $u \leq 10 \text{ km s}^{-1}$ ) a casi la misma densidad.

#### **2.1.5. Turbulencia**

Recientemente se ha considerado a la turbulencia como un ingrediente fundamental de la evolución de NMs en la determinación de su morfología, tiempos de vida, tasa de formación estelar, etc. (ver p. ej. Vázquez-Semadeni *et al.* 2000, MacLow & Klessen 2004; Ballesteros-Paredes *et al.* 2007; McKee & Ostriker 2007).

Se entiende por turbulencia al régimen de flujo de fluidos caracterizado por grandes fluctuaciones de velocidad, desordenadas, que frecuentemente están estructuradas de manera anidada o jerárquica. La turbulencia involucra movimientos de gas en todas las escalas, los cuales proveen una presión hidrodinámica, que generalmente se considera como un agente de soporte en contra de la auto-gravedad. Sin embargo, estos movimientos son macroscópicos y por lo tanto también mezclan el gas y pueden producir compresión en lugar de soporte (Elmegreen 1993; Ballesteros-Paredes, Vázquez-Semadeni & Scalo 1999, Vázquez-Semadeni *et al.* 2008).

Existe un parámetro adimensional de la turbulencia llamado *número de Reynolds*, definido por

$$\mathcal{R} \equiv \frac{|\mathbf{u} \cdot \nabla \mathbf{u}|}{|\nu \nabla^2 \mathbf{u}|} \sim \frac{u_l l}{\nu}, \quad (2.48)$$

en donde  $u_l$  es la velocidad característica a la escala  $l$ , y  $\nu$  es la viscosidad cinemática,  $\nu = c_s \lambda$  (donde  $\lambda = 1/n^{1/3}$  es el camino libre medio entre las partículas del gas). Así,  $\mathcal{R}$  cuantifica qué tan turbulento es el flujo. La transición de un flujo laminar a uno turbulento se da cuando  $\mathcal{R}$  excede cierto valor crítico típicamente de algunas veces  $10^3$  (Frisch 1995). En el MI,  $\mathcal{R} \sim 10^5 - 10^7$  (Elmegreen & Scalo 2004).

### Turbulencia incompresible

Mucha de la teoría de turbulencia es hecha en el espacio de Fourier, pues permite un tratamiento directo de las longitudes de escala, lo cual lleva naturalmente a la definición del *espectro de energía*  $E(k)$ , definido como la energía cinética por unidad de masa contenida en los modos con números de onda en cierto intervalo ( $k, k + dk$ ):

$$E = \frac{1}{2} \int |\mathbf{u}_k|^2 d^3x = \frac{1}{2} \int_0^\infty E(k) dx. \quad (2.49)$$

En 1941, Kolmogorov y Obukhov obtuvieron la forma de  $E(k)$  basándose en las siguientes hipótesis:

- I) El número de Reynolds es suficientemente grande para que un rango muy grande de escalas esté activo en el flujo.
- II) La energía es inyectada principalmente a grandes escalas y disipada a pequeñas escalas.
- III) La transferencia de energía es local (ocurriendo principalmente entre números de onda similares) en el espacio de Fourier.
- IV) En sistema está en equilibrio estadístico, de tal forma que las tasas de inyección, transporte y disipación de energía son iguales.

Las primeras tres hipótesis, implican la existencia de un *rango inercial* de números de onda en los cuales la energía se transfiere en cascada únicamente desde escalas grandes hacia escalas menores. Una consecuencia de la última hipótesis es que la tasa de transferencia de energía en el rango inercial es independiente del número de onda  $k = 2\pi/l \sim l^{-1}$ .

Con estas hipótesis, y considerando la tasa de transferencia de energía en el rango inercial  $\epsilon = u_l^2/\tau_l$  para los vórtices del flujo turbulento de escala de tamaño  $l$ , con escala de tiempo característica  $\tau_l = l/u_l$ , donde  $u_l$  es la diferencia de velocidades característica a través del vórtice, obtuvieron:

$$u_l = (\epsilon l)^{1/3} \sim l^{1/3}. \quad (2.50)$$

Por otro lado,  $u_l$  se relaciona con el espectro de energías mediante:

$$u_l^2 = 2 \int_{2\pi/l}^{\infty} E(k) dk, \quad (2.51)$$

finalmente encontraron la relación:

$$E(k) \sim \frac{u_k^2}{k} \sim k^{-5/3}, \quad (2.52)$$

conocida como la ley  $-5/3$  de Kolmogorov y verificada experimentalmente (Grant *et al.* 1962), y observacionalmente en el MI difuso (Armstrong *et al.* 1995).

## **Turbulencia compresible**

El estudio de la turbulencia en el MI (compresible) resulta mucho más complejo que la turbulencia incompresible, aunque la comprensión de esta última permite avanzar en el estudio de la primera. Adicionalmente, existe otro parámetro llamado *número de Mach*, definido como el cociente entre la velocidad del gas  $u$  y la velocidad del sonido adiabática  $c_s$ ,

$$\mathfrak{M} \equiv \frac{u}{c_s}, \quad (2.53)$$

el cual permite clasificar el régimen del flujo según:

- $\mathfrak{M} < 1$  Subsónico,
- $\mathfrak{M} \sim 1$  Transónico,
- $\mathfrak{M} > 1$  Supersónico,
- $\mathfrak{M} \gg 1$  Hipersónico.

Los flujos subsónicos se comportan de manera esencialmente incompresible, y su compresibilidad aumenta según aumenta  $\mathfrak{M}$  (Landau & Lifshitz 1982).

Empleando un análisis dimensional parecido al empleado para el número de Reynolds, es posible determinar el orden de magnitud de las fluctuaciones de densidad en un flujo isotérmico, para el cual  $\nabla P \propto c_s^2 \nabla \rho$ . Si las magnitudes de la presión y el término advectivo de la ecuación de conservación de momento son comparables, se tiene que  $|\nabla P| \sim \rho |\mathbf{u} \cdot \nabla \mathbf{u}|$ , donde  $u$  es la amplitud típica de las fluctuaciones de velocidad. Si suponemos que la velocidad media del flujo es cero, se pueden emplear las aproximaciones  $\nabla \rho \sim \delta \rho / l$  y  $\nabla \mathbf{u} \sim \mathbf{u} / l$  para obtener una estimación de las fluctuaciones de densidad para un gas isotérmico (ver p. ej. Shu 1992, Cap. 16):

$$\frac{\delta \rho}{\rho} \sim \mathfrak{M}^2. \quad (2.54)$$

Observemos que la relación (2.54) es similar a la relación para un salto de densidad asociado con el flujo en un choque isotérmico.

### **Espectro de energía de la turbulencia compresible**

En el caso altamente compresible, debe tomarse en cuenta la presencia de choques en la determinación del espectro de energía (ver p. ej. Saffman 1970). Burgers (1974), introdujo un modelo simplificado de turbulencia sin presión, para investigar los efectos del término advectivo no lineal en la dinámica de fluidos empleando la ecuación de onda conocida como la *ecuación de Burgers*, y cuya solución es conocida como el espectro de Burgers,  $k^{-2}$ . Este es usado frecuentemente para modelar la dinámica del MI turbulento (ver p. ej. Katz & Green 1986, Scalo *et al.* 1988).

En el caso compresible, la pendiente espectral pareciera depender del grado de compresibilidad del flujo (contrariamente al caso incompresible, para

el cual el espectro de Kolmogorov se considera universal). En el régimen débilmente compresible, las simulaciones numéricas en 3D indican que ambas componentes (rotacional y compresible) del campo de velocidades desarrollan un espectro de equilibrio estadístico, es decir, de Kolmogorov (Porter *et al.* 1994, 1995). A una compresibilidad muy alta, ambas componentes se aproximan a un espectro  $k^{-2}$  (Passot *et al.* 1995; Kritsuk *et al.* 2007).

La turbulencia en el MI difuso tibio es transónica, con la velocidad del sonido y los movimientos térmicos ambos  $\sim 10 \text{ km s}^{-1}$  (ver p. ej. Heiles & Troland 2003), mientras que dentro de las NMs es altamente supersónica, con números de Mach  $\mathfrak{M} \approx 5 - 20$  (ver p. ej. Zuckerman & Palmer 1974). Las NMs son, por lo tanto, muy compresibles. Hasta hace poco tiempo (ver p. ej. MacLow & Klessen 2004) se pensaba que, a escalas globales, la turbulencia previene el colapso monolítico de las nubes y, a pequeña escala, promueve el colapso local por medio de compresiones aunque hoy en día existen sugerencias de que, por el contrario, la turbulencia es alimentada por el colapso gravitacional (Field *et al.* 2008; Klessen & Hennebelle 2010). La interacción entre la propia gravedad de las nubes y la turbulencia controlan la formación y el comportamiento de las NMs, así como los procesos de formación de núcleos y estrellas dentro de ellas.

Además, en el MI existe una gran variedad de fuentes de energía térmica y cinética (como jets, vientos estelares, expansión de regiones HII, explosiones de supernovas, etc.), por lo que podemos decir que la turbulencia es entonces un fenómeno generalizado.

## 2.2. Componentes del medio interestelar

El MI está sometido a procesos de inyección de energía que calientan y agitan el gas, así como a mecanismos de enfriamiento por emisión de fotones mediante transiciones electrónicas, rotacionales y vibracionales de sus componentes (ver p. ej. Spitzer 1988, Osterbrock 1989). Por otra parte, está sometido a su propia autogravedad y la de las estrellas de la Galaxia. El balance de estos mecanismos determina las condiciones físicas del gas (densidad, temperatura, velocidad, intensidad del campo magnético, composición química) que lo mantienen en un amplio rango de regímenes. Estos regímenes, tam-

bién llamados *fases* o componentes, son un conjunto de rangos de densidad y temperatura en los que el gas tiende a establecerse con facilidad, aunque probablemente existe todo un continuo de temperaturas y densidades intermedias menos abundantes (Vázquez-Semadeni *et al.* 2003; Vázquez-Semadeni *et al.* 2009). Estas fases pueden coexistir a la misma presión debido a la inestabilidad térmica descrita en la sección 2.1.2. En general, el MI puede ser considerado como un fluido que, bajo el efecto de los procesos de enfriamiento y calentamiento, tiene un comportamiento termodinámico casi isobárico que da lugar a dichas fases. Las excepciones a este comportamiento casi isobárico son aquellas regiones que se encuentran a alta presión, ya sea por calentamiento local aplicado por fuentes estelares (remanentes de supernova o regiones HII), o las nubes moleculares, que se encuentran a presiones muy superiores a la promedio del MI, lo que se entiende como una consecuencia de que son fuertemente autogravitantes. La tabla 2.2 muestra los principales parámetros descriptivos de estas componentes. El volumen ocupado se refiere al plano medio galáctico.

Componente	Volumen %	Altura [pc]	Temperatura [K]	Densidad [cm <sup>-3</sup> ]	Detección
gas ionizado caliente	20 – 70	1000 – 3000	~ 10 <sup>6</sup>	~ 0.0065	Líneas altamente ionizadas
gas ionizado tibio	20 – 50	1000	~ 8000	0.2 – 0.5	H <sub>α</sub>
gas neutro tibio	10 – 20	300 – 400	6000 – 10000	0.2 – 0.5	HI 21cm
gas neutro frío	1 – 5	100 – 300	50 – 100	20 – 50	HI, 21cm en absorción
nubes moleculares	< 1	70	10 – 20	10 <sup>2</sup> – 10 <sup>6</sup>	CO y otras moléculas

Tabla 2.2: Parámetros descriptivos de las componentes del MI

### 2.2.1. Gas ionizado caliente

Esta componente está formada por gas ionizado colisionalmente, que parece ser generado por explosiones de supernovas y, en una proporción menor, por vientos estelares (McCray & Snow 1979; Spitzer 1990). Las explosiones de supernova generan ondas de choque que se propagan rápidamente en el MI, abriendo cavidades llenas de gas rarificado y caliente rodeado por un cascarón denso de material chocado. Cox & Smith (1974) señalaron que estas cavidades se expanden por su mayor presión térmica, enrareciendo altamente el gas que contienen, y por lo tanto, su tiempo de enfriamiento radiativo es suficientemente grande para permanecer en este estado por millones de años, ocupando una fracción significativa del volumen del MI (su llamado *factor de llenado*). Sin embargo, se ha demostrado que la presión magnética interestelar reduce substancialmente su factor de llenado (McKee, 1990) y en la actualidad se piensa que no es mayor de 20 % en el plano medio de la Galaxia (de Avezil & Breitschwerdt, 2004).

### 2.2.2. Gas ionizado tibio

Las estrellas más masivas y calientes (tipos O y B) emiten una intensa radiación UV, la cual es suficientemente energética para ionizar los átomos de hidrógeno. Como resultado, estas estrellas están rodeadas por las llamadas regiones HII, en las cuales el hidrógeno está casi completamente ionizado y cuya frontera con el MI del entorno es abrupta (ver sección 2.1.3).

Sin embargo, la mayoría del gas interestelar ionizado tibio se encuentra fuera de las regiones HII en forma de gas difuso extendido, cuya temperatura inferida a partir del ancho de las líneas de emisión de  $H_{\alpha}^2$ ,  $[S_{II}]$  6716 Å,  $[N_{II}]$  6583 Å es de  $\sim 8000$  K. Se piensa que los fotones que ionizan esta componente son fotones “escapados” de las regiones HII (Reynolds 1985; Haffner *et al.* 1999)

---

<sup>2</sup>Balmer,  $n = 3 \rightarrow 2$

### 2.2.3. Gas neutro tibio

Esta componente está constituida por hidrógeno atómico neutro, detectable en la línea de emisión de 21 cm. Bajo la mayoría de las condiciones interestelares, las colisiones entre partículas son tan poco frecuentes que casi todos los átomos de hidrógeno tienen su electrón en su nivel base de energía. Todas las transiciones electrónicas entre este nivel y un estado excitado (serie de Lyman) caen en el espectro UV, empezando con la transición  $L_\alpha$  ( $\lambda = 1216 \text{ \AA}$ ), por lo que esta transición no es observable en el óptico. Por otro lado, los electrones y protones tienen espín (giro), el cual produce un campo magnético tal que cuando sus respectivos espines en el átomo de hidrógeno están alineados en el estado base, tienen un nivel de energía ligeramente más alto que cuando están opuestos. Esta diferencia de energía corresponde a la *estructura hiperfina* del estado base del hidrógeno y resulta en la emisión de la línea de 21 cm, pues la colisión entre dos átomos produce la excitación del electrón haciéndolo pasar a un nivel de energía superior (espines paralelos); como éste no es el nivel de energía mínima, espontáneamente pasa al nivel inferior emitiendo un fotón ( $\lambda \approx 21.1 \text{ cm}$ ). Esta transición es altamente prohibida, por lo cual dichos fotones penetran profundamente en el MI, haciendo a la Galaxia ópticamente transparente en su mayoría. Por otra parte, son intrínsecamente tan raras ( $A_{21} = 2.85 \times 10^{-15} \text{ s}^{-1}$ ) que se requiere de nubes de gas con profundidades ópticas muy grandes para hacer posible su detección. Esto es muy conveniente, ya que el HI se encuentra permeando densamente el disco galáctico.

### 2.2.4. Gas neutro frío

El gas de esta componente está aglomerado en nubes HI difusas, que generalmente se encuentran hasta una altura de 100 pc sobre el disco galáctico. Sin embargo se han observado nubes a varios kpc por encima y cayendo a muy altas velocidades (ver p. ej. Dieter 1964, Oort 1966).

Estas nubes constituyen la fase densa fría del gas atómico (Field *et al.* 1969; Wolfire *et al.* 1995), y en ocasiones se observan en absorción de HI contra el fondo brillante de HI tibio (Gibson *et al.* 2000). Recientemente se ha propuesto que estas nubes pueden ser las progenitoras de las nubes

moleculares (Vázquez-Semadeni *et al.* 2006).

### 2.2.5. Gas molecular

Las moléculas que constituyen esta componente son formadas mediante cadenas de reacciones químicas muy complejas. Siendo el hidrógeno el átomo más abundante en el universo, la molécula de  $\text{H}_2$  resulta ser la más abundante y se piensa que se forma por recombinación en la superficie de granos de polvo interestelares (Hollenbach & Salpeter, 1971). La siguiente molécula en abundancia es la de CO, identificada en un espectro estelar ultravioleta (Smidth & Stecher, 1971). A pesar de estarse formando continuamente, las únicas regiones donde las moléculas pueden sobrevivir en grandes cantidades a la disociación por radiación, son los interiores de nubes densas y los interiores profundos de nubes difusas. Las temperaturas observadas de las regiones moleculares son explicadas como el resultado del balance térmico entre el calentamiento por rayos cósmicos (y en los bordes de las nubes, colisiones con fotoelectrones provenientes de granos de polvo y con  $\text{H}_2$  radiativamente excitadas) y el enfriamiento por líneas de emisión molecular (principalmente de CO) cuya tasa incrementa rápidamente con la temperatura (de Jong *et al.*, 1980; Goldsmith, 1987; Hollenbach & Tielens, 1999). Las colisiones con granos de polvo también entran en el balance térmico dependiendo de la temperatura del polvo con respecto a la del gas (Burke & Hollenbach, 1983).

### 2.2.6. Regiones HII

Estas regiones, discutidas en la sec (2.1.3), se encuentran fuera de equilibrio, y contienen una fracción muy pequeña de la masa. Sin embargo, son importantes como agentes que inyectan energía al MI, agitando la turbulencia y dispersando las nubes en que se encuentran.

## 2.3. Nubes Moleculares

La *nubes moleculares* (NM) son de particular relevancia para el presente trabajo ya que son los sitios de formación estelar en la Galaxia (ver p. ej. las revisiones por Blitz, 1993; Williams, Blitz & McKee, 2000) que pretendemos modelar. Estas son nubes de gas molecular, en su mayoría  $\text{H}_2$ , y en menor proporción CO y muchas otras moléculas más complejas, conformando un fluido autogravitante, casi isotérmico (a  $10 - 20 \text{ K}$ ), magnetizado y turbulento. En sus regiones más densas, la presión térmica es muy superior a la del resto del MI, que es  $P/k_B \sim 3000 \text{ K cm}^3$ , por lo cual se piensa que la autogravedad tiene un papel predominante en su dinámica y confinamiento. Específicamente, si las nubes moleculares se están contrayendo gravitacionalmente y tienen un comportamiento isotérmico (de modo que  $P = c_s^2 \rho$ ), entonces su presión se incrementa linealmente con su densidad.

La molécula de  $\text{H}_2$  carece de momento eléctrico dipolar permanente y tiene un momento de inercia muy pequeño. Todas sus transiciones permitidas caen fuera del dominio del radio (Field *et al.*, 1966), lo cual la hace imposible de ser observada directamente en las regiones interestelares frías y generalmente oscurecidas en donde se forma y sobrevive, debido a las altas densidades ( $\sim 10^3 - 10^6 \text{ cm}^{-3}$ ) que permiten su formación sobre granos de polvo. En contraste, la molécula del CO posee transiciones de baja frecuencia rotacional que son fácilmente observadas incluso en el gas molecular muy tenue. Por consiguiente, y debido a su relativamente alta abundancia ( $[\text{CO}/\text{H}_2] \approx 1.8 \times 10^{-4}$ ), la emisión de CO es un excelente rastreador de hidrógeno molecular, aunque la correspondencia entre CO y  $\text{H}_2$  dista de ser perfecta. Además, existen otras moléculas rastreadoras en el gas más denso de las NM, haciendo posible conocer sus condiciones físicas.

Dentro de la Galaxia, las nubes moleculares representan menos de 1% del volumen del MI, pero contienen  $\sim 50\%$  de su masa total (ver p. ej. Cox 2005). La mayor parte del gas molecular está contenido en un anillo entre  $3.5 - 7.5 \text{ kpc}$  desde el centro galáctico. Los mapas a gran escala de CO han mostrado que su posición se correlaciona con los brazos espirales (Dame *et al.* 1987; Ferrière 2001), lo cual hace pensar que se forma y disocia en una escala de tiempo menor a  $10 \text{ Myr}$ , tiempo que le toma al material pasar a través de la región de los brazos espirales (Dame *et al.* 1987).

Verticalmente, el gas molecular se encuentra en una capa en el plano medio del disco Galáctico (de altura  $\sim 50 - 75$  pc), mucho más delgada que la del gas atómico tibio neutro y ionizado. Esta descripción uniforme del gas molecular es un promedio para volúmenes grandes; sin embargo, a pequeña escala es altamente irregular, con la mayoría del gas concentrado en nubes discretas y complejos de nubes.

### 2.3.1. Estructura de las nubes moleculares

Como veremos posteriormente (sec. 3.1.3), una nube de gas extendida e isotérmica en contracción gravitacional se fragmenta al aumentar la densidad, produciendo regiones más pequeñas como filamentos, grumos y núcleos (Heitsch & Hartmann 2008, Vázquez-Semadeni *et al.* 2009) y finalmente llevando a la formación de estrellas. La fragmentación y formación de estructura involucran una gran variedad de procesos físicos en cada uno de sus pasos intermedios.

De acuerdo con Vázquez-Semadeni (2003), podemos definir una *estructura* como una región que tiene valores específicos (o rangos de valores) de variables físicas, con al menos una de ellas claramente distinguible de sus valores en los alrededores de la región. Entonces, podemos definir la fragmentación como la formación de estructuras de pequeña escala dentro de una de escala mayor, lo que implica que la fragmentación es un proceso *multiescala*. En particular, las nubes interestelares, ya sea moleculares o de HI, son esencialmente estructuras de alta densidad inmersas dentro de otras de más gran escala con densidad menor. Por supuesto, es posible que mediante el mismo mecanismo o uno diferente, se fragmenten las estructuras de pequeña escala en estructuras incluso más pequeñas. Este proceso se conoce como *fragmentación jerárquica*.

En el medio interestelar, podemos reconocer estructuras jerárquicas pasando desde brazos espirales ( $L \sim 10$  kpc), supernubes ( $L \sim 1$  kpc,  $n \sim 1$  cm<sup>3</sup>) (Elmegreen & Elmegreen 1983), nubes moleculares gigantes ( $L \sim 50$  pc,  $n \sim 50$  cm<sup>3</sup>), nubes moleculares ( $L \sim 5$  pc,  $n \sim 10^3$  cm<sup>3</sup>), grumos ( $L \sim 0.5$  pc,  $n \sim 10^4$  cm<sup>3</sup>) y núcleos densos ( $L \sim 0.1$  pc,  $n > 10^4$  cm<sup>3</sup>) (Blitz 1993; Elmegreen 1993). Estas categorías parecen emerger una dentro de la otra, pues el MI

es más bien *un continuo* con estructuras en todo un rango de las escalas, lo cual contrasta con la visión tradicional de un MI con fases bien diferenciadas entre sí.

Se cree que la inestabilidad gravitacional apoyada por el enfriamiento, así como la inestabilidad de Parker, son responsables de la formación de las estructuras más grandes como supernubes atómicas (Elmegreen 1993, Franco *et al.* 2002). Dentro de éstas, las NMGs pueden ser originadas por a) inestabilidad gravitacional, b) inestabilidad de magneto-Jeans (Kim & Ostriker 2001) y c) movimientos turbulentos a gran escala producidos por la actividad estelar (Ballesteros-Paredes *et al.* 1999, de Avillez 2000, Mac Low 2002). Sin embargo, en años recientes (Burkert & Hartmann 2004; Hartmann & Burkert 2007; Vázquez-Semadeni *et al.* 2007, 2009) se ha producido un retorno a la noción de que las NMGs se encuentran en estado de contracción gravitacional (Goldreich & Kwan, 1974) y que la baja tasa global de formación estelar observada se debe a la interrupción de este proceso debido a la retroalimentación estelar (Hartmann *et al.* 2001; Vázquez-Semadeni *et al.* 2010).

Larson (1981) encontró relaciones básicas de escalamiento entre la masa, tamaño y dispersión de velocidades observadas en las NMs y su subestructura:

1. Las nubes parecen estar en equilibrio virial aproximado  $\sigma^2 \propto GM/R$ , en donde  $\sigma$  es la dispersión de velocidades en una dimensión,  $M$  es la masa y  $R$  es el tamaño de la nube.
2. El ancho de línea se incrementa con el tamaño,  $\sigma \propto R^p$ , con  $p \sim 0.5$ .
3. La densidad media  $\bar{n}_H \propto R^{-q}$  decrece al aumentar el tamaño, de tal modo que la densidad de columna  $N_H = \frac{4}{3}\bar{n}_H R$  es aproximadamente constante. ( $q \sim 1$ )

Estas son conocidas como *las relaciones de Larson*. Obsérvese que únicamente dos de ellas son independientes y que estas relaciones se extienden hasta seis órdenes de magnitud en masa, siendo aplicables desde la escala de los grumos dentro de las nubes hasta la escala de las NMGs.

### 2.3.2. Evolución

El proceso de cómo se forman las estrellas a partir de las NMs empieza con el estudio de tales objetos y de cómo dentro de ellas se condensan los núcleos densos gravitacionalmente. La acreción del gas a partir del medio tibio difuso es una parte integral de la dinámica y la evolución de las nubes (Hennebelle & Pérault 1999, 2000; Heitsch *et al.* 2005, 2006; Vázquez-Semadeni *et al.* 2006, 2007, 2009. Vázquez-Semadeni 2010) y, por lo tanto, éstas no puede ser consideradas como aisladas.

#### Formación

Las NMs parecen formarse a partir del gas difuso del MI durante compresiones a gran escala, generadas por supernovas, inestabilidades magneto-rotacionales o inestabilidad gravitacional en discos galácticos de estrellas y gas (Ballesteros-Paredes *et al.*, 2007; Hennebelle, MacLow & Vázquez-Semadeni *et al.* 2009).

Estas compresiones tienen lugar en forma sucesiva dentro de la Galaxia, de tal manera que primeramente comprimen el *Medio Neutro Tibio* (MNT) formando nubes a partir del *Medio Neutro Frío* (MNF) más densas y frías ( $\langle n \rangle \sim 100 \text{ cm}^3$ ,  $T = 50 - 100 \text{ K}$ ) que posteriormente entran en contracción gravitacional con anchos de línea supersónicos, sobrepresurizadas a más de 10 veces la presión térmica media del MI (Blitz & Williams, 1999) y finalmente formando NMGs. Dentro de ellas se encuentran regiones más densas ( $\langle n \rangle \sim 1000 \text{ cm}^3$ ) llamadas grumos (“clumps” en inglés). Los grumos tienden a ser autogravitantes (aunque véase también Bertoldi & McKee, 1992), con una equipartición entre sus energías gravitacional, cinética y magnética, que se interpreta como *virialización*. Dentro de ellos existen grumos más densos y pequeños llamados núcleos densos. La formación de estos fragmentos se atribuye, o bien a la contracción gravitacional (Hoyle 1953; Vázquez-Semadeni *et al.* 2009) o a la turbulencia en las nubes (ver p. ej. MacLow & Klessen 2004 y referencias ahí citadas).

Un parámetro fundamental es la *densidad de columna* de las nubes, pues además de tener una relación con la densidad volumétrica total, al ser sufi-

cientemente grande, la capa exterior de la nube actúa como un escudo protector para el gas interno en la misma nube, impidiendo que la radiación interestelar disocie las moléculas, permitiendo así la formación de gas molecular. Se encuentra que el gas comienza a volverse molecular para  $N \gtrsim 10^{21} \text{ cm}^{-2}$  (Franco & Cox 1986; Hartmann *et al.* 2001).

### Formación estelar

Una pregunta fundamental planteada por las observaciones de las NMs es su aparente estabilidad contra el colapso gravitacional. Se piensa que el tiempo característico de la formación estelar  $t_{FE}$ , debe exceder en gran medida al tiempo de caída libre  $t_{cl}$ , para que la *tasa de formación estelar* (TFE  $\sim M_{mol}/t_{FE}$ , en donde  $M_{mol}$  es la masa total del gas molecular en la galaxia), sea tan baja como la observada  $\text{TFE} \simeq 1 - 3M_{\odot} \text{ yr}^{-1}$  (Zuckerman & Palmer, 1974).

A pesar de ser las regiones de formación estelar dentro de la Galaxia, las NMGs forman estrellas con una baja eficiencia, definida como:

$$\text{EFE} = \frac{M_*}{M_d + M_*}, \quad (2.55)$$

donde  $M_*$  es la masa en estrellas dentro de la nube y  $M_d$  es la masa del gas denso de la nube. Myers *et al.* (1986) mostraron que para NMGs, la  $\text{EFE} \sim 2\%$ , lo cual constituye uno de los principales retos actuales en la investigación sobre la formación estelar.

Tradicionalmente se considera que el proceso responsable de este soporte se manifiesta en los anchos de línea altamente supersónicos observados. Debido a que las nubes no podrían estar en estado de colapso, pues  $\frac{M_{mol}}{\tau_{ff}} \gg \frac{M_{mol}}{\tau_{FE}}$ , se ha considerado que estos anchos de línea deben corresponder a movimientos turbulentos (Zuckerman & Evans, 1974). Sin embargo, recientemente se ha reinterpretado a estos movimientos como contracción gravitacional (Hartmann & Burkert 2007; Vázquez-Semadeni *et al.* 2008; Ballesteros-Paredes *et al.* 2010).

### **Destrucción**

Dependiendo del tamaño de las NM y sus subestructuras, es posible que ellas sean destruidas por los procesos de retroalimentación estelar, si éstos tienen suficiente energía como para vencer el pozo de potencial gravitacional de las nubes. Es decir, las NMGs podrían ser desgarradas, sobreviviendo por grandes periodos de tiempo antes de ser completamente destruidas, mientras que pequeños grumos dentro de NMs serían completamente destruidos incluso antes de que inicien la formación estelar. Esto es consistente con simulaciones recientes de Colín *et al.* (2010). Y a su vez, también son consistentes con simulaciones de Krumholz, Matzner & McKee (2006), con la única diferencia fundamental de que las primeras toman en cuenta la acreción de material del medio difuso que las rodea, además de que la acreción compite con la formación estelar y la retroalimentación estelar, regulando la masa de la nube y la FE. Los núcleos de NMs serían entonces objetos dinámicos, no cuasiestáticos, con tiempos de vida relativamente cortos que no exceden unos pocos millones de años (Vázquez-Semadeni *et al.* 2005a; Galván-Madrid *et al.* 2007).

# Capítulo 3

## Formación estelar

Las estrellas se forman cuando algún fragmento de una nube se vuelve gravitacionalmente inestable, de modo que la gravedad domina sobre los demás agentes que se oponen a ella. En este capítulo, se describen los principios físicos de esta competencia.

### 3.1. Principios básicos

#### 3.1.1. Inestabilidad gravitacional

El principio más importante para la formación estelar, y en general de cualquier objeto que se forme por un colapso gravitacional, es el de la inestabilidad gravitacional de Jeans, que describe la competencia entre la autogravedad de una parcela de gas y la oposición a ella por el gradiente de presión térmica. Si consideramos una nube de gas esférica isotérmica de masa  $M$ , con densidad numérica  $n$  y densidad de masa  $\rho = \mu n$  uniforme, donde  $\mu$  es la masa por partícula, con radio  $R = [3M/4\pi\rho]^{1/3}$ , en ausencia de movimientos macroscópicos y campos magnéticos cuyas energías térmica y gravitacional están en equilibrio ( $E_{term} = |E_{grav}|$ ), tenemos que:

$$\frac{kT}{\mu} = \frac{1}{5} \frac{GM}{R}. \quad (3.1)$$

Podemos entonces definir la *masa de Jeans* como:

$$M_J = \left[ \frac{375}{4\mu} \right]^{1/2} \left( \frac{kT}{G} \right)^{3/2} \rho^{-1/2} \simeq 10^6 \pi T_2^{3/2} n_2^{-1/2} M_\odot, \quad (3.2)$$

donde  $T_2 = T/10^2$  K y  $n_2 = n/10^2$  cm<sup>-3</sup>. Esta es una masa crítica arriba de la cual el gradiente de presión térmica es incapaz de soportar a la nube en contra de su propia gravedad. Así, una nube con  $M > M_J$  procede a “colapsarse” gravitacionalmente, disminuyendo aceleradamente su radio y aumentando su densidad. Este proceso sólo se interrumpe si la ecuación de estado cambia, de modo que  $\gamma$  llegue a ser mayor que 4/3 (Chandrasekhar). De lo contrario, dentro del límite clásico, se produce una singularidad.

De manera similar, encontramos que el tamaño de una nube a punto de colapsar está dado por la *longitud de Jeans*:

$$L_J = \sqrt{\frac{15kT}{4\pi G\mu\rho}}, \quad (3.3)$$

que se obtiene sustituyendo  $M$  en la ec. (3.1) y considerando  $R = L$ . Si la nube tiene un tamaño mayor a la longitud de Jeans ( $L \geq L_J$ ) colapsará, calentándose e incrementando su densidad. Si por el contrario,  $L < L_J$ , se expandirá, enfriándose y disminuyendo su densidad.

### 3.1.2. Tiempo de caída libre

Si la presión interna de la nube es incapaz de detener el colapso, el gas se moverá hacia el centro de la nube en un régimen cercano a la caída libre. Suponiendo una nube esférica de radio inicial  $R$ , la velocidad de caída de la cáscara exterior cuando se ha contraído a un radio  $r$  será:

$$\frac{\partial r}{\partial t} = - \left[ 2GM \left( \frac{1}{r} - \frac{1}{R} \right) \right]^{1/2}, \quad (3.4)$$

de donde, con la ayuda del cambio de variable  $x = r/R$  y la densidad promedio inicial  $\rho_0 = 3M(4\pi R^3)^{-1}$ , definimos el *tiempo de caída libre* como el tiempo total desde el inicio de la contracción hasta la singularidad:

$$t_{cl} = \int_0^{t_{cl}} dt = - \int_0^{t_{cl}} \left[ \frac{R^3 x}{2GM(1-x)} \right]^{1/2} dx = \left( \frac{3\pi}{32G\rho_0} \right)^{1/2}, \quad (3.5)$$

donde la última igualdad se obtuvo sustituyendo  $\rho_0$ , haciendo el cambio de variable adicional  $x = \cos^2(\xi)$  y considerando los valores de  $r$  en los límites de integración.

Notese también que el tiempo de caída libre es independiente del radio inicial de la nube. Por consiguiente, mientras la densidad inicial de toda la nube sea uniforme, todas las partes de la nube tardarán el mismo tiempo en colapsar y la densidad aumentará a la misma tasa en todas partes de la nube. Este comportamiento es conocido como “colapso homólogo”. Si la nube estuviera condensada centralmente cuando el colapso comienza, entonces el tiempo de caída libre del material cerca del centro sería más corto que el tiempo de caída libre para el material más externo. Por lo tanto, la densidad central incrementaría más rápidamente que en otras regiones de la nube.

### 3.1.3. Fragmentación de Hoyle

A pesar de que todavía no existe una teoría general y detallada del proceso de transformación del gas en estrellas, los anteriores conceptos nos dan un panorama de las primeras etapas de dicho proceso. De acuerdo con la ec. (3.2), si consideramos una nube isotérmica en colapso gravitacional, su densidad aumentará a medida que ella se contrae, causando que  $M_J$  disminuya, por lo cual, fragmentos de masas menores a los de la nube se volverán inestables y empezarán a condensarse y a colapsar por su cuenta. Dicho proceso es llamado *fragmentación* de Hoyle (1953) y puede continuar hasta que la opacidad del gas sea tan alta que la radiación no pueda escapar fácilmente, cambiando la ecuación de estado efectiva del gas, y deteniendo así la contracción y la fragmentación.

Como la tasa de pérdida de energía gravitacional de la nube es aproximadamente

$$\dot{E}_{grav} \sim \frac{\left(\frac{GM^2}{R}\right)}{t_{cl}}, \quad (3.6)$$

y considerando que durante el colapso el proceso de liberación de energía tiene una eficiencia  $\eta$ , podremos expresar la luminosidad radiada como

$$L_{rad} = (4\pi R^2)(\eta\sigma T^4). \quad (3.7)$$

donde  $\sigma$  es la constante de Boltzmann,  $T$  es la temperatura y  $0 < \eta < 1$ . Si la nube es ópticamente gruesa y la energía emitida por algunas partes de la nube es absorbida y después reemitida por otras partes de la nube, entonces el fragmento radiaría como cuerpo negro ( $\eta = 1$ ,  $L = L_{rad}$ )

$$L = 4\pi\sigma R^2 T^4. \quad (3.8)$$

Así, al igualar (3.6) con (3.8) obtenemos el límite inferior para la masa de los fragmentos:

$$M_{Frag} \geq \pi^2 \left(\frac{32}{9}\right)^{1/4} \left(\frac{k}{\mu}\right)^{9/4} \frac{T^{1/4}}{G^{3/2}\sigma^{1/2}} \simeq 0.01 T_1^{1/4} M_\odot, \quad (3.9)$$

donde  $T_1 = T/10$  K. En realidad, el mecanismo de fragmentación es influenciado por rotación, turbulencia y campos magnéticos, por lo cual éste análisis representa una simplificación.

### 3.1.4. Campo magnético. Soporte y relación masa-flujo

Consideraremos ahora otro mecanismo de soporte contra la autogravedad de una parcela de gas: el proporcionado por la presión magnética, aunque, como en la sección anterior, haremos el tratamiento en términos de balance de energías.

La energía almacenada en un campo magnético uniforme  $\mathbf{B}$  en el volumen  $V$  de una esfera es (ver p. ej. Shu 1992):

$$E_{mag} = \frac{B^2}{8\pi} V = \frac{B^2 R^3}{6}, \quad (3.10)$$

y el flujo magnético que pasa por una sección transversal  $A$  cilíndrica de radio  $R$  es

$$\Phi = BA = \pi BR^2. \quad (3.11)$$

En ausencia de procesos de disipación o difusión, el flujo se conserva (ver p. ej. Shu 1992). A esta propiedad se le denomina congelamiento del flujo.

Entonces, el cociente de la energía gravitacional a la energía magnética es

$$\frac{|E_{grav}|}{E_{mag}} = \frac{18 GM^2}{5 B^2 R^4} = \frac{18\pi^2 G}{5} \left(\frac{M}{\Phi}\right)^2, \quad (3.12)$$

en donde el término entre paréntesis recibe el nombre de cociente masa-flujo.

Luego, la condición para colapso  $|E_{grav}| > E_{mag}$  implica que

$$\frac{M}{\Phi} > \left(\frac{M}{\Phi}\right)_{crit} \equiv \sqrt{\frac{5}{18\pi^2 G}}, \quad (3.13)$$

en donde el valor numérico varía dependiendo de la geometría. Nakano & Nakamura (1978), hicieron un calculo más preciso para geometría cilíndrica, obteniendo:

$$\left(\frac{M}{\Phi}\right)_{crit} \equiv \sqrt{\frac{1}{4\pi^2 G}}. \quad (3.14)$$

Así, podemos definir la criticalidad magnética de una nube como

- $\frac{M}{\Phi} < \left(\frac{M}{\Phi}\right)_{crit}$  magnéticamente subcrítica,
- $\frac{M}{\Phi} > \left(\frac{M}{\Phi}\right)_{crit}$  magnéticamente supercrítica.

Una nube magnéticamente subcrítica puede ser soportada por el campo magético en contra de su autogravedad, mientras que una nube supercrítica no. Este criterio es usado observacionalmente para determinar si el campo magnético es suficiente, o no, para soportar las nubes y sus núcleos (ver p. ej. Crutcher *et al.* 2003).

## 3.2. Problemas contemporáneos

La investigación actual de la formación estelar se enfoca en los problemas a nivel colectivo más que a nivel individual. A continuación mencionamos algunos de éstos y la forma en que han sido atacados.

### 3.2.1. Modelo “estándar” de FE regulada magnéticamente

Shu, Adams & Lizano (1987) establecieron el *modelo estándar* de la formación estelar (ver figura 3.1), en el que consideraron una nube molecular,

cuyo gas está en un estado cercano al isotérmico y rotando muy lentamente. En él, identificaron cuatro etapas de la formación estelar: a) dentro de los gru-

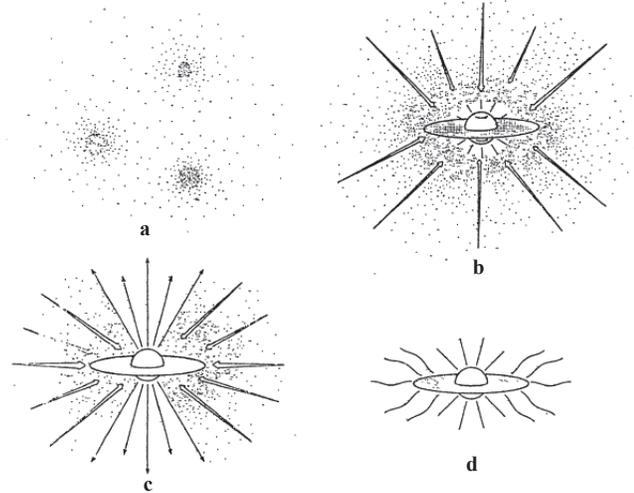


Fig. 3.1: Etapas de la formación estelar según el modelo de Shu, Adams & Lizano, 1987.

mos se forman núcleos cuya rotación es lenta. En los grumos magnéticamente “subcríticos” (sec. 3.1.4), esto ocurre a través del “escurrimiento” lento de las partículas neutras entre las ionizadas, mecanismo conocido como *difusión ambipolar*, causando que las regiones más densas de los grumos escapen al soporte magnético y colapsen, mientras que las envolturas permanecen soportadas. En los grumos magnéticamente “supercríticos”, el colapso involucra toda la masa del grumo, ya que todo el conjunto carece de suficiente soporte magnético para evitar el colapso. b) Cuando los núcleos en proceso de colapso sobrepasan el límite de inestabilidad, tiene lugar la formación de una protoestrella central y un disco, profundamente inmersos en una envoltura de gas y polvo. Conforme la acreción tiene lugar, finalmente se inicia el quemado de deuterio en las regiones centrales de la estrella, volviéndola casi completamente convectiva. Si la convección y la rotación diferencial de la estrella se combinan para producir un dínamo, la estrella puede evolucionar naturalmente a un estado de expulsión de material desde su superficie en forma de vientos estelares. Sin embargo, al principio, la presión del material que cae directamente en la superficie detendrá esta expulsión. Debido a que el material cae preferencialmente al disco, los vientos estelares gradualmente

se hacen más fuertes en las regiones de menor resistencia (los polos rotacionales), dando lugar a flujos colimados y flujos bipolares. c) Conforme pasa el tiempo, más y más material en rotación sigue cayendo preferentemente al disco en lugar de la estrella, con lo cual los vientos estelares ampliarán su ángulo de apertura, eventualmente abarcando toda la esfera. d) La caída de material se inhibe gradualmente hasta revertirse, revelando una estrella recién formada, con un disco circunestelar.

Ahora sabemos que la FE se produce por colapso gravitacional de una parcela de gas, dentro de los núcleos densos de las NMGs. Es a este nivel de la fragmentación jerárquica que alcanzamos la conexión a la formación estelar. Sin embargo, si los grumos y los núcleos tienen fluctuaciones de densidad producida por turbulencia supersónica en las NMs, es natural preguntarse si el origen de tal turbulencia es compatible con el modelo estándar de la formación estelar.

En años recientes, ha surgido un nuevo modelo que intenta entender tales asuntos como propiedades de nubes moleculares, la eficiencia de formación estelar e incluso la función de masa inicial de las estrellas, como una consecuencia de parámetros globales de la turbulencia en las NMs. En este modelo, la baja eficiencia de formación estelar es una consecuencia del soporte turbulento *global* (a gran escala) de las nubes, junto con el colapso *local* (a pequeña escala) inducido por la turbulencia compresible en un tipo de “coladera” turbulenta (Vázquez-Semadeni *et al.* 2003). Sin embargo, el comportamiento colectivo del gas en el MI y su relación con las propiedades estadísticas de la FE continúa siendo tema de debate e investigación.

A continuación se describen algunos de los procesos más importantes en el estudio de la FE, así como también algunos conceptos mencionados en el modelo estándar.

### 3.2.2. La tasa y la eficiencia de la formación estelar

La *eficiencia de formación estelar* (EFE) es la fracción de gas molecular convertido en estrellas al cabo de un cierto tiempo  $\Delta t$ , y se define como:

$$\text{EFE}(t) = \frac{1}{M_{\text{tot}}(t)} \int_0^{\Delta t} \text{TFE}(t) dt, \quad (3.15)$$

donde  $\text{TFE}(t)$  es la *tasa de formación estelar*.

Observacionalmente, se ha encontrado que la EFE es pequeña y fluctúa alrededor de  $\sim 2\%$  para complejos moleculares gigantes (Myers *et al.* 1986), elevándose a  $10 - 30\%$  para núcleos densos que forman cúmulos estelares (Lada & Lada, 2003), así como para galaxias “starburst”, cuyo gas es principalmente molecular (Kennicutt, 1998).

La baja EFE a la escala de las NMGs sigue siendo un asunto de fuerte debate, dentro del cual existen dos escenarios principales que intentan explicarla, descritos a continuación.

#### Escenarios de formación estelar

Si la TFE es aproximadamente constante, la ec. (3.15) puede escribirse como

$$\text{EFE}(t) = \frac{\text{TFE}}{M_{\text{tot}}} \Delta t. \quad (3.16)$$

Así, surgen dos escenarios de formación estelar (FE):

- 1) una TFE grande y  $\Delta t$  pequeño, conocido como escenario de *FE rápida*,
- o
- 2) una TFE pequeña y  $\Delta t$  grande, conocido como escenario de *FE lenta*.

En el escenario de la FE rápida, las estrellas se forman rápidamente, y acto seguido desgarran a su nube madre por dispersión y/o fotoionización antes de que la masa gaseosa de la nube sea completamente convertida en estrellas (Withworth 1979; Elmegreen 1983; Franco *et al.* 1994, Williams & McKee 1997; Hartman *et al.* 2001). En este caso, la TFE de regiones activas de

formación estelar puede ser grande por periodos breves, y después detenerse debido a la gran cantidad de estrellas formadas.

En el escenario de la FE lenta, la nube forma estrellas a una baja tasa durante largos periodos de tiempo (implicando una larga duración de la nube). En el modelo estándar de la FE (Shu *et al.* 1987; Mouschovias 1991), esto ocurría porque las nubes eran consideradas magnéticamente subcríticas, y formaban estrellas sólo en las pequeñas regiones que se tornaban supercríticas por la acción de la DA. Sin embargo, observaciones más recientes (Bourke *et al.* 2001; Troland & Cutcher 2008; Crutcher *et al.* 2009) sugieren que las nubes son magnéticamente supercríticas, por lo que el modelo estándar ya no parece viable, al menos para explicar la formación de la mayoría de las estrellas.

En la visión más moderna, el escenario de FE lenta también se basa en el efecto de la retroalimentación estelar. Sin embargo, en este caso, su papel es el de producir movimientos turbulentos azarosos dentro de las nubes, oponiéndose a la autogravedad y permitiendo que ellas permanezcan en un equilibrio casi estacionario por tiempos significativamente más grandes que su tiempo de caída libre ( $t_{cl}$ ) (Krumholz & McKee 2005, 2006; Krumholz, Matzner & McKee 2006; Li & Nakamura 2007). En este caso, la baja EFE podría deberse a un papel dual de la turbulencia supersónica en las nubes autogravitantes, oponiéndose al colapso global de la nube mientras que promueve el colapso local de fluctuaciones turbulentas de densidad, las cuales contienen fracciones pequeñas del total de la masa de la nube (Klessen, Heitsch & Mac Low 2000; Vázquez-Semadeni, Ballesteros-Paredes & Klessen 2003; Mac Low & Klessen 2004; Ballesteros-Paredes *et al.* 2007).

Otra controversia, relacionada con el control de la EFE, se refiere a la naturaleza de los movimientos que originan los anchos de línea observados en NMGs y su subestructura. Inicialmente se propuso que ésta última corresponde a contracción gravitacional (Goldreich & Kwan 1974), pero Zuckerman & Palmer (1974) notaron que esto implicaría que la TFE que sería del orden de la masa molecular total del gas en la Galaxia dividida por el tiempo de caída libre típico para la NMG,  $\sim 250 M_{\odot} \text{ yr}^{-1}$ , lo que resulta ser dos órdenes de magnitud mayor que la TFE observada. Zuckerman & Evans (1974) sugirieron que los perfiles de línea observados correspondían a movimientos turbulentos aleatorios a pequeñas escalas, noción que prevalece hasta nues-

tros días. Sin embargo, diversos estudios han mostrado que los movimientos en las NMs son consistentes con movimientos a gran escala en las nubes en lugar de pequeña escala (Ossenkopf & Mac Low 2002, Brunt 2003, Padoan *et al.* 2009). Esto es más consistente con un escenario en el que las nubes moleculares están en un proceso de contracción (Burkert & Hartmann 2004; Hartmann & Burkert 2007; Vázquez-Semadeni *et al.* 2007, 2009). En este caso, el control de la EFE puede recaer sobre la retroalimentación estelar (Hartmann *et al.* 2001; Vázquez-Semadeni *et al.* 2010).

### 3.2.3. Función de masa inicial

La *función de masa inicial* (FMI) describe las abundancias relativas de los diferentes tipos de estrellas, es decir, dada la masa de gas que se transforma en estrellas, nos dice cómo se distribuye esta masa en diferentes tipos de estrellas.

La FMI estelar es un diagnóstico que fundamenta del proceso de formación estelar y entender su origen físico es una de las principales metas de cualquier teoría de formación estelar. En términos generales, la FMI es la distribución de masas de las estrellas al formarse y observacionalmente se encuentra que está descrita por una ley de potencias (Salpeter, 1955):

$$dN \propto M^{-\alpha} dM. \quad (3.17)$$

En términos de intervalos logarítmicos de masa, se tiene que

$$dN \propto M^{\Gamma} d(\log M), \quad (3.18)$$

donde  $\Gamma = -(\alpha - 1)$ . Tanto  $\alpha$  como  $\Gamma$  son constantes que dependen del intervalo de masas considerado. En ocasiones se habla de que la FMI es una distribución lognormal (Miller & Scalo 1979).

Una descripción actual de la FMI es la de Kroupa (2001), que, expresada en la forma (3.17), es

$$\begin{aligned} dN &\propto M^{-2.3} dM, & (M \geq 0.5M_{\odot}) \\ dN &\propto M^{-1.3} dM, & (0.08 \leq M < 0.5M_{\odot}) \\ dN &\propto M^{-0.3} dM, & (M < 0.08M_{\odot}). \end{aligned}$$

Como hemos visto, los procesos físicos como la gravedad, acreción, turbulencia, campos magnéticos y retroalimentación estelar, entre otros, juegan un papel importante en el proceso de formación estelar y por lo tanto afectan la distribución de masas estelares resultante. Las observaciones indican que la formación estelar existe en un intervalo amplio de masas. El resultado más importante del estudio de la FMI es que las estrellas de baja masa son mucho más abundantes que las de alta masa, lo cual debe ser explicado por cualquier teoría que intente describir la FE.

### 3.3. Mecanismos de regulación de la FE

#### 3.3.1. Difusión ambipolar

La difusión ambipolar (DA) es un proceso importante en el estado inicial de la formación estelar, mediante el cual los grumos de una NM se desacoplan del campo magnético interestelar y del soporte proporcionado por éste. El campo magnético permea el espacio interestelar y actúa sobre los iones (mayormente protones) y electrones. Estas partículas cargadas, a su vez, colisionan con átomos neutros y, en la mayoría de los casos, los acoplan al campo. Sin embargo, en una NM, donde la fracción de ionización es muy baja ( $\sim 10^{-5}$ ), las partículas neutras raramente encuentran partículas cargadas y, por lo tanto, no están bien acopladas. Así, las partículas neutras pueden “resbalar” por entre los iones, permitiendo que el flujo magnético “escape” de la región en contracción gravitacional.

#### 3.3.2. Retroalimentación

Como hemos visto, las estrellas nacen en los núcleos densos de nubes moleculares. Las estrellas recién nacidas producen flujos y/o radiación ionizante UV, que inyectando (retroalimentando) energía térmica y cinética dentro de sus nubes madre, afectan:

1. la formación estelar subsecuente dentro de las nubes,

- la tasa de formación estelar (TFE),
  - la eficiencia de formación estelar (EFE),
  - la distribución de masa (FMI) de productos estelares,
2. la subestructura, condiciones físicas, y la evolución de la nube,
- generación de turbulencia,
  - estructura de densidad, y
  - la dinámica de la nube, pudiendo proporcionarle soporte, o causar su dispersión.

### Retroalimentación y generación de turbulencia

Uno de los efectos más estudiados de la retroalimentación estelar es la producción de turbulencia en el MI. Uno de los primeros trabajos en este sentido fue el de Norman & Silk (1980), quienes sugirieron que los anchos de línea de las nubes resultan de la superposición de líneas más delgadas ( $0.1 - 3 \text{ km s}^{-1}$ ), provenientes de grumos individuales moviéndose a velocidades supersónicas de  $\sim 3 \text{ km s}^{-1}$ , producidos por los vientos estelares de las estrellas de masa baja e intermedia. Posteriormente el efecto de la retroalimentación estelar comenzó a investigarse numéricamente (Bania & Lyon 1980; Chiang & Prendergast 1985; Chiang & Bregman 1988; Rosen, Bregman & Norman 1993; Vázquez-Semadeni *et al.* 1995, 1996; Korpi *et al.* 1999; de Avillez 2000; de Avillez & Breitschweidt 2004, 2005). Estos trabajos investigaron principalmente el efecto del calentamiento por radiación UV ionizante y supernovas.

Más recientemente, varios trabajos han investigado el efecto de los flujos colimados. Nakamura & Li (2006) y Li & Nakamura (2007), en sus simulaciones MHD isotérmicas, encontraron que la turbulencia interestelar inicial es rápidamente reemplazada por la turbulencia producida por los flujos estelares y que la distribución de densidad, con un perfil cercano a una ley de potencias de  $r^{-3/2}$ , evoluciona hacia un estado más centralmente condensado de equilibrio cuasi-estático. Matzner (2007) realizó un modelo analítico tomando en cuenta la colimación y la distribución de intensidad de los flujos estelares, así como una cascada turbulenta de momento. Encontró que

la relación tamaño-ancho de línea y el balance de energía son funciones del flujo de masa y la densidad de columna. Carroll *et al.* (2009, 2010) hicieron comparaciones de turbulencia agitada por flujos colimados *vs.* turbulencia aleatoria (agitada por campos de fuerza azarosos), notando un espectro de pendiente mucho más pronunciada para el caso de turbulencia generada por los flujos a escala de los mismos. Wang *et al.* (2010) realizaron simulaciones con turbulencia inicial, campo magnético y flujos estelares añadidos en secuencia, encontrando que las estrellas masivas se alimentan de regiones a escala de los grumos y lo hacen en menor proporción en presencia de flujos estelares y campos magnéticos.

En general, los flujos estelares parecen capaces de generar turbulencia a escala de los grumos ( $\sim 1$  pc) pero no a escala de las NMGs completas. Así mismo, cuando se incluye la autogravedad, ocurre una concentración central continua a pesar de la generación de turbulencia.

### La retroalimentación, el estado final de la nube, la TFE y la EFE

En 1954, Oort sugirió la existencia de un ciclo dinámico de formación estelar y de las nubes: las nubes en colapso forman nuevas estrellas, alrededor de las cuales se forman regiones HII, que a su vez dispersan la nube, formando subnubes. La fusión de estas sub-nubes forma nuevas nubes que posteriormente se tornan inestables y colapsan, repitiendo el ciclo. Field & Saslaw (1965), formularon su derivación analítica suponiendo que las sub-nubes se encuentran en balance de presión con el medio difuso interestelar y que su masa crece únicamente por fusión:

- $SFR \propto \rho^2$   
Ley de Schmidt (1959). Este resultado proviene de que, en su modelo, las nubes crecen por colisiones, un proceso de 2 cuerpos, dependiente de  $\rho^2$ .
- $\frac{dN}{dM} \propto M^{-3/2}$   
Espectro de masa de las nubes derivado de la dinámica de las colisiones: crecimiento *vs.* ruptura.
- Una baja TFE.

Withworth (1979) calculó la fracción de la masa de una nube que es erosionada por la radiación ionizante proveniente de estrellas masivas, concluyendo que únicamente convertir el 4% de la nube en estrellas (con una FMI estándar) es suficiente para destruir la nube completamente. Norman & Silk (1980) fueron los primeros en considerar el papel de los vientos isotrópicos provenientes de estrellas de baja masa (ahora conocidos como flujos estelares, concluyendo que las nubes tienen tiempos de vida del orden de  $10^7 - 10^8$  años y que el espectro de masa de las nubes es de la forma  $\sim m^{-n}$ ,  $n > 2$ . Por su parte, Franco & Cox (1983) calcularon la TFE por unidad de volumen necesaria para soportar las nubes mediante vientos estelares de baja masa, encontrando  $\text{TFE} \sim n_{H_2}^{13/8}$ .

McKee (1989) desarrolló un modelo del balance de energía de las nubes y la TFE, basándose en la fotoionización externa: las NMs inician contrayéndose como nubes frías, magnéticamente supercríticas, con grumos magnéticamente subcríticos. Conforme lo hacen, se auto-escudan más eficientemente, permitiendo que la DA opere en los grumos, los cuales entonces colapsan y forman estrellas de baja masa. Estas estrellas producen flujos que estabilizan a las nubes, llevándolas a un estado cercano al equilibrio, hasta que las estrellas masivas finalmente dispersan a las nubes. Como resultado, encontró que, para las nubes en equilibrio, el *parámetro virial*  $\alpha_{vir} \equiv 2E_k/|E_G| \sim 4 - 8$ ,  $B \sim 20 - 40\mu G$  y el tiempo de agotamiento del gas por formación estelar  $\sim 2 - 4 \times 10^8$  años. Franco *et al.* (1994), calcularon el número máximo de estrellas OB que pueden existir en una nube sin desgarrarla, resultando en una EFE  $\sim 2 - 16\%$  dentro de complejos de nubes, dependiendo de la localización de las estrellas en la nube y de los detalles de la FMI, y con  $\langle \text{EFE} \rangle \sim 5\%$  para el gas molecular promedio. Matzner & McKee (2000) hicieron un estimado analítico para núcleos densos que forman cúmulos estelares regulados por flujos estelares colimados. Encontraron una TFE mucho mayor ( $\sim 10\times$ ) para flujos colimados que para vientos esféricos (TFE  $\sim 30 - 50\%$ ).

Matzner (2002), argumentando que las regiones HII (en lugar de los flujos estelares) son la principal fuente de energía de NMGs, estableció un balance de energía entre el decaimiento turbulento y su reposición por las regiones HII, concluyendo que el flujo ionizante requerido es del orden un factor de dos veces el flujo galáctico total observado y que la escala de tiempo para la

destrucción de las NMGs decrece con la masa como:

$$t \approx 2 \times 10^7 \left( \frac{M}{10^6 M_\odot} \right)^{-1/3} \text{ años.} \quad (3.19)$$

Krumholz, Matzner & McKee (2006), realizaron un estudio semi-analítico con geometría esféricamente simétrica, considerando que la nube está en balance virial y su turbulencia está alimentada por la radiación UV ionizante de las estrellas masivas, con la TFE dada por la expresión de Krumholz & McKee (2005), que depende de la masa, tamaño y dispersión de velocidades en la nube. Consideraron además que la nube puede “morir” por desgarrar, disociación o colapso. Así obtuvieron como resultado que las nubes más grandes sobreviven del orden de  $2 - 3 t_{cl}$ ,  $TFE \sim 5 - 10\%$  y  $\alpha_{vir} \sim 1 - 3$ , y para que las nubes pequeñas sobreviven  $\sim 1 t_{cl}$ .

Vázquez-Semadeni *et al.* (2010a), realizaron simulaciones numéricas de formación y evolución de NMGs incluyendo la retroalimentación por formación estelar, autogravedad y la función enfriamiento de Koyama & Inutsuka (2002), que modela la inestabilidad térmica del MI atómico. Incluyeron también una prescripción sencilla para modelar la retroalimentación estelar por calentamiento por ionización de estrellas masivas con el objeto de estudiar su efecto sobre la evolución de la NMG. Encontraron que las nubes más masivas no se mantienen en equilibrio virial ni son dispersadas por retroalimentación estelar. En su lugar, la acreción aproximadamente balancea el consumo de gas por formación estelar y la dispersión por retroalimentación estelar, lo cual sugiere que las nubes están continuamente acretando gas y procesándolo, “canalizando” el material a las estrellas a partir del MNT, debido al pozo de potencial a gran escala de la nube. Por lo tanto, la masa de la nube está principalmente determinada por la acreción proveniente del MNT,

$$\dot{M}_{cl} \approx 3 \times 10^3 M_\odot \text{ Mega año}^{-1}, \quad (3.20)$$

mientras que la TFE es inhibida por la retroalimentación estelar, que reduce la tasa de conversión de gas a estrellas. La retroalimentación opera en escalas más pequeñas que el pozo de potencial gravitacional de la nube y, por lo tanto, no puede evitar la acreción proveniente de grandes escalas. Una limitación de estas simulaciones es que sólo se considera un tipo de estrella ( $\sim$  tipo B), que resulta excesiva para nubes pequeñas y muy débil para las masivas, pudiendo tener un papel no despreciable en la dispersión de nubes pequeñas y la permanencia de las cavidades en nubes más grandes.

Como se puede apreciar, la retroalimentación por flujos estelares, el calentamiento radiativo y la presión de radiación juegan un papel fundamental en la regulación de la TFE, EFE.

Resulta importante entonces implementar la retroalimentación en códigos numéricos que permitan la suficiente resolución para seguir simultáneamente la evolución de una NMG a gran escala y de las inmediaciones de una estrella masiva, en donde actúa el calentamiento producido por ésta. Vázquez-Semadeni *et al.* (2010), utilizaron un código de malla adaptiva, pero es conveniente implementar la retroalimentación en códigos Lagrangianos, como *smoothed particle hydrodynamics* (SPH), que también permiten muy alta resolución en las regiones de alta densidad y un seguimiento preciso de la masa acretaada por las estrellas recién formadas (Jappsen *et al.* 2005). En la actualidad existen códigos numéricos SPH de dominio público, como GADGET2 (Springel, 2005), pero la versión pública no incluye retroalimentación. En lo que resta de esta tesis describiremos nuestro trabajo para incluirla en éste código, siguiendo el algoritmo propuesto por Dale *et al.*(2007).

# Capítulo 4

## Descripción del código GADGET2.

El software GADGET2, creado por Springel (2005) a partir de la versión inicial de Springel, Yoshida & White (2001), es un código adecuado para simulaciones cosmológicas de formación de estructura y para simulación de interacción de galaxias. Evoluciona fluidos autogravitantes sin colisiones con un enfoque de N-cuerpos, y un gas colisional mediante una dinámica de partículas suavizadas (SPH). Está escrito en el lenguaje de programación C y está adaptado para emplear programación en paralelo con el fin de distribuir la carga de cómputo entre varios procesadores. A continuación se describe su funcionamiento en el sentido de la mecánica de fluidos numérica y se extiende un poco la descripción matemática y computacional con propósitos meramente ilustrativos.

### 4.1. Método SPH

La técnica SPH permite resolver problemas hidrodinámicos y es adecuada para problemas tridimensionales en astrofísica que no requieren resolver en forma exacta frentes de choque. En ella, el comportamiento de un fluido es simulado siguiendo el movimiento de un número finito de partículas discretas con coordenadas  $\mathbf{r}_i$ , velocidades  $\mathbf{v}_i$  y masas  $m_i$  que pueden considerarse como elementos del fluido que muestrean el espacio fase del fluido. El índice  $i$  corre

sobre todas las partículas incluidas en la simulación.

En esencia, el método SPH es un método de interpolación que permite que cualquier función sea expresada en términos de sus valores en un conjunto desordenado de puntos (las partículas). La idea original fue propuesta por Lucy (1977), Gingold & Monaghan (1977) y Monaghan (1992). En este método, la estimación de la densidad es de importancia fundamental y en GADGET2 se hace en la forma:

$$\rho_i = \sum_{j=1}^N m_j W(|\mathbf{r}_{ij}|, h_i), \quad (4.1)$$

en donde  $\mathbf{r}_{ij} \equiv \mathbf{r}_i - \mathbf{r}_j$ , los índices  $i$  y  $j$  denotan a las partículas y  $W(r, h)$  es un *kernel* de interpolación de tipo spline llamado *kernel de suavizado* (Monaghan & Lattanzio 1985), definido por:

$$W(r, h) = \frac{8}{\pi h^3} \begin{cases} 1 - 6 \left(\frac{r}{h}\right)^2 + 6 \left(\frac{r}{h}\right)^3, & 0 \leq \frac{r}{h} \leq \frac{1}{2}, \\ 2 \left(1 - \frac{r}{h}\right)^3, & \frac{1}{2} \leq \frac{r}{h} \leq 1, \\ 0, & \frac{r}{h} > 1. \end{cases} \quad (4.2)$$

siendo  $h$  la *longitud de suavizado*. El kernel  $W(r, h)$  tiene las propiedades

$$\int W(r, h) dr = 1, \quad (4.3)$$

y

$$\lim_{h \rightarrow 0} W(r, h) = \delta(r), \quad (4.4)$$

en donde  $\delta$  es la función delta de Dirac.

Es importante tomar en cuenta que, contrariamente a otras implementaciones SPH, GADGET2 usa una convención en la cual el kernel se hace nulo en  $r/h = 1$  y no en  $r/h = 2$ .

La longitud de suavizado de cada partícula es definida de tal forma que el volumen que ocupa su kernel contenga una masa constante, es decir, por:

$$h_i = \left( \frac{3N\bar{m}}{4\pi\rho_i} \right)^{1/3}, \quad (4.5)$$

en donde  $N$  es un parámetro ajustable que define es el número partículas dentro de una esfera de radio  $h_i$  alrededor de la partícula  $i$  y  $\bar{m}$  es la masa promedio de las partículas.

## 4.2. Cálculo de la fuerza gravitacional

La gravedad es uno de los agentes más importantes en la formación de estructura en el MI. Desafortunadamente su naturaleza de largo alcance y el enorme rango dinámico de las simulaciones en astrofísica hacen de su cálculo preciso y eficiente un reto.

El método principal empleado en GADGET2 para calcular la fuerza gravitacional y lograr la adaptividad espacial es una expansión jerárquica multipolo, conocida como *algoritmo de árbol* BH (Barnes & Hut 1986) y es obtenida a través de una subdivisión recursiva del espacio. Existen tres características importantes en estos algoritmos: el tipo de agrupamiento empleado para las partículas, el orden a escoger para la expansión multipolo y el criterio de apertura de los nodos.

En el problema de N-cuerpos, el cálculo de la fuerza gravitacional ejercida sobre una partícula  $i$  por todas las demás partículas  $j$  es calculada empleando el enfoque de suma directa ( $F_i = \sum_j \frac{Gm_j}{r_{ij}^2}$ ). En lugar de requerir  $N - 1$  fuerzas parciales por partícula, el enfoque BH calcula la fuerza gravitacional en  $\mathcal{O}(\log N)$  operaciones. Para ello, se emplea un nodo raíz (celda espacial cúbica) para englobar la distribución de masa completa, el cual es repetidamente subdividido en ocho nodos hijo (celdas cada vez más pequeñas) cada uno de la mitad de su longitud lateral, hasta que se termina con un nodo “hoja” conteniendo una sola partícula. Las fuerzas ejercidas sobre las partículas son obtenidas “recorriendo” el árbol, es decir, iniciando en el nodo raíz y tomando la decisión de cuando o no la expansión multipolo del nodo provee una fuerza parcial suficientemente precisa, lo cual en general es el caso de los nodos que son pequeños y suficientemente distantes. Si la respuesta es afirmativa, se usa la fuerza multipolo y el recorrido a través de la rama actual del árbol puede ser terminado; en caso contrario el nodo es “abierto”, es decir, se consideran sus nodos hijo individualmente. A pesar de que este tipo de algoritmos generalmente representa solamente una aproximación a la fuerza verdadera, el error puede ser controlado convenientemente modificando el criterio de apertura de los nodos, debido a que una alta exactitud puede ser obtenida si se recorren los niveles inferiores del árbol.

GADGET2 emplea también un árbol octal con momentos de monopolo con

el fin de permitir actualizaciones dinámicas múltiples del árbol consistentes con el esquema temporal descrito posteriormente.

Específicamente, el criterio de apertura de las celdas considera que un nodo de masa  $M$  y extensión  $l$  a una distancia  $r$  puede usarse si:

$$\frac{GM}{r^2} \left(\frac{l}{r}\right)^2 \leq \alpha |\mathbf{a}|, \quad (4.6)$$

donde  $|\mathbf{a}|$  es el tamaño de la aceleración total obtenida en el último paso temporal y  $\alpha$  es un parámetro de tolerancia. Este criterio típicamente mantiene constante el error de la fuerza absoluta (introducido en cada interacción nodo-partícula), comparando una estimación del error de truncamiento con el tamaño de la fuerza total esperada y ajustando el estado dinámico de la simulación en caso de ser necesario para lograrlo.

GADGET2 adicionalmente emplea un algoritmo híbrido consistente en una síntesis del método de *partícula-malla* (PM) y el algoritmo de árbol. Esta implementación es llamada TreePM (Xu 1995; Bode, Ostriker & Xu 2000; Bagla 2002). En el método TreePM, la ecuación para el potencial

$$\phi(\mathbf{x}) = \sum_i m_i \varphi(\mathbf{x} - \mathbf{x}_i), \quad (4.7)$$

es explícitamente separada en el espacio de Fourier como una parte de largo alcance y otra de corto alcance  $\phi_k = \phi_k^{largo} + \phi_k^{corto}$ , donde

$$\phi_k^{largo} = \phi_k \exp(-k^2 r_s^2). \quad (4.8)$$

Aquí,  $r_s$  describe la escala espacial usada para la separación de la fuerza. Entonces, el potencial de largo alcance puede ser calculado muy eficientemente por métodos de Fourier y el potencial de corto alcance puede ser resuelto en el espacio real notando que para  $r_s \ll L$  está dado por

$$\phi^{corto}(\mathbf{x}) = -G \sum_i \frac{m_i}{r_i} \operatorname{erfc}\left(\frac{r_i}{2r_s}\right), \quad (4.9)$$

donde  $L$  es el tamaño de la caja computacional,  $r_i = \min(|\mathbf{x} - \mathbf{r}_i - \mathbf{n}L|)$  es la distancia más pequeña de cualquiera de las imágenes<sup>1</sup> de la partícula

<sup>1</sup>Si se tienen fronteras periódicas, ver sección 5.2.4

$i$  al punto  $\mathbf{x}$ , y  $\text{erfc}$  representa la función de error complementaria, la cual suprime rápidamente la fuerza para distancias grandes comparadas con  $r_s$  y, por lo tanto, solamente la imagen más cercana contribuye a la fuerza de rango corto. Entonces, la fuerza de corto alcance puede ser calculada por el algoritmo de árbol requiriendo que éste sea recorrido únicamente en una pequeña región espacial alrededor de cada partícula objetivo y no se requiere de correcciones para condiciones periódicas, lo cual resulta en una mejora substancial del desempeño. Normalmente, uno gana exactitud en el cálculo exacto de la fuerza de largo alcance y no en la aproximación del método de árbol. Sin embargo, al englobar todo el volumen de la simulación en el enfoque TreePM, la resolución lograda en el cálculo de la fuerza es igual en todas partes de la caja computacional. Separando el potencial de esta forma, el método TreePM mantiene las ventajas de los algoritmos de árbol, principalmente la insensibilidad al agrupamiento masivo de partículas (clustering), un rango dinámico esencialmente ilimitado y su control preciso cerca de la escala de suavizado de la fuerza gravitacional.

La ventaja de emplear un árbol octal es que, a comparación de un árbol binario, requiere un menor número de nodos para una cierta cantidad  $N$  de partículas ( $\sim 0.3N$  para una distribución de masa casi homogénea y  $\sim 0.65$  para una fuertemente agrupada). Esto lo hace ideal para aplicarlo en la búsqueda de partículas vecinas SPH, la cual es una tarea básica del método SPH. GADGET2 lo hace siguiendo el método de búsqueda de rango descrita por Hernquist & Katz (1989) en el que, para una región esférica de búsqueda dada de radio  $h_i$  alrededor de una posición  $r_i$ , el recorrido del árbol se hace siguiendo un criterio de apertura que examina cuando existe cualquier traslape geométrico entre el nodo actual (celda espacial) y la región de búsqueda. Si esto ocurre, el turno se cede a los nodos hijo para continuar la búsqueda. En caso contrario, el recorrido a lo largo de la rama actual del árbol se descarta. De esta forma, la búsqueda es restringida a la región local a la posición  $r_i$ .

Un enfoque comúnmente considerado para resolver problemas de N-cuerpos y SPH es la descomposición de volumen computacional en una serie de dominios, cada uno asignado a un procesador. GADGET2 emplea una estructura computacional fractal de llenado de espacio (Warren & Salmon 1993, 1995) llamada *curva de Peano-Hilbert* (PH), la cual mapea el espacio 3D en una curva unidimensional. Esta última es entonces dividida en piezas que definen

los dominios individuales (ver fig. 4.1). Entre las propiedades de la curva

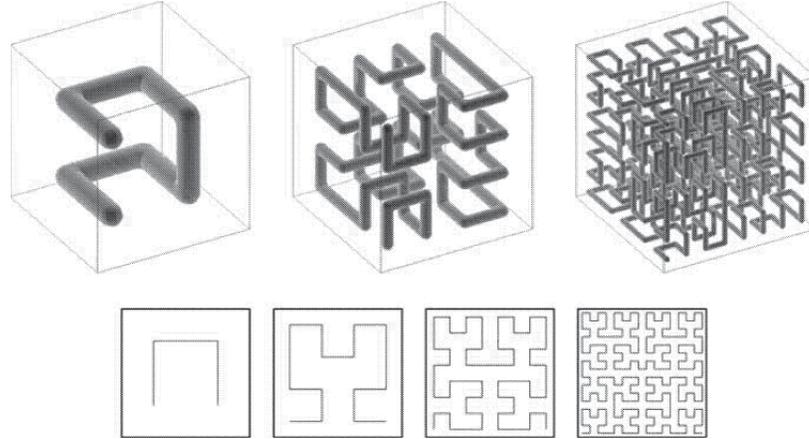


Fig. 4.1: La curva de Peano-Hilbert llenando el espacio en 3D y 2D.

de PH, se encuentran: a) la autosimilaridad, permitiendo “contraer” la curva (dividiendo el espacio en dos para cada dirección) y obtener una curva similar, pero de menor orden, es decir, más pequeña; b) el mapeo se preserva localmente, es decir, los puntos que son cercanos en la curva PH 1D, son por lo general, también cercanos en el espacio 3D; c) existe una correspondencia cercana entre la descomposición espacial obtenida mediante el árbol octal jerárquico BH y la obtenida segmentando la curva de PH (ver fig. 4.2). Por ejemplo, si consideramos una curva PH que llene una caja (el nodo raíz del árbol) abarcando todo el conjunto de partículas, al dividir esta curva en ocho partes iguales, y recursivamente subdividir cada parte en otras ocho partes iguales, se regenera la estructura de árbol octal BH. Si se asigna un segmento arbitrario de la curva PH a un procesador, el volumen correspondiente será entonces compatible con la estructura de nodo de un árbol BH global cubriendo el volumen completo, y pudiendo de esta forma asignar efectivamente una serie de ramas del árbol a cada procesador. Debido a ello, se obtiene un árbol cuya estructura no es afectada por el método de paralelización y el resultado para la fuerza gravitacional obtenida mediante el árbol se hace independiente del número de procesadores.

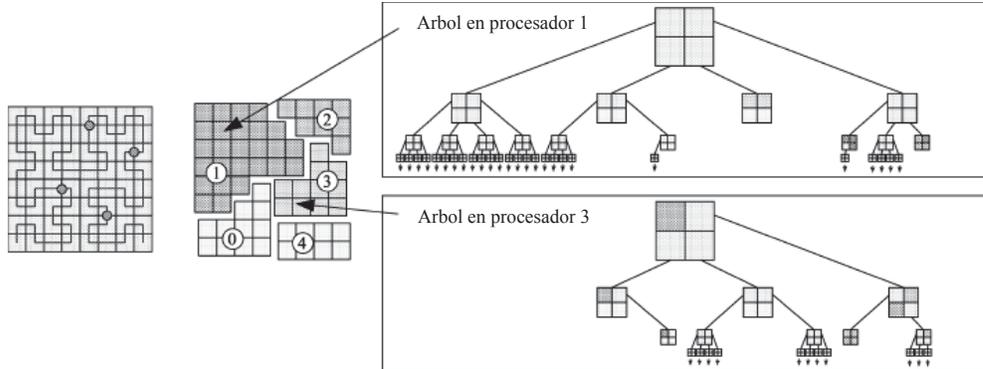


Fig. 4.2: Ilustración la relación entre el árbol octal BH y la descomposición de un dominio basada en la curva de Peano-Hilbert.

#### 4.2.1. Esquema de integración temporal.

La integración numérica de las ecuaciones diferenciales para las variables físicas de cada partícula pueden ser efectuadas mediante métodos estándar con un paso de tiempo controlado que involucra la condición de Courant (descrita más adelante; ver “Restricciones en el avance temporal”), términos de forzamiento y difusión por viscosidad (Monaghan 1989, 1992).

Las aproximaciones numéricas de la evolución temporal de un sistema Hamiltoniano obtenidas de un método ordinario de integración numérica (p. ej. Runge-Kutta) en general introducen perturbaciones no Hamiltonianas, las cuales pueden cambiar el comportamiento del sistema a largo plazo; es decir, los sistemas Hamiltonianos no son robustos en el sentido de que no son estructuralmente estables contra perturbaciones no Hamiltonianas. Es posible preservar la estructura Hamiltoniana del sistema durante la integración temporal si cada paso se formula como una transformación canónica (que preservan la forma de las ecuaciones de Hamilton, equivalente a preservar el espacio fase). Así, la integración temporal puede ser vista como una transformación canónica continua generada por el Hamiltoniano.

En el problema de N-cuerpos, el Hamiltoniano es separado en las partes cinética y potencial,

$$H = H_{cin} + H_{pot}, \quad (4.10)$$

en cuyo caso los operadores de evolución temporal, puede ser calculados exactamente para cada una de las partes. Esto da surgimiento a los operadores de transporte y aceleración de las partículas, también conocidos como “drift” y “kick” respectivamente:

$$D_t(\Delta t) = \begin{cases} \mathbf{p}_i \mapsto & \mathbf{p}_i \\ \mathbf{x}_i \mapsto & \mathbf{x}_i + \frac{\mathbf{p}_i}{m_i} \int_t^{t+\Delta t} \frac{dt}{a^2}, \end{cases} \quad (4.11)$$

$$K_t(\Delta t) = \begin{cases} \mathbf{x}_i \mapsto & \mathbf{x}_i \\ \mathbf{p}_i \mapsto & \mathbf{p}_i + \mathbf{f}_i \int_t^{t+\Delta t} \frac{dt}{a}, \end{cases} \quad (4.12)$$

donde  $\mathbf{x}_i$  y  $\mathbf{p}_i$  son los vectores de posición y de momento de la partícula  $i$ ,  $a$  es el factor de escala usado en cosmología, y

$$\mathbf{f}_i = - \sum_j m_i m_j \frac{\partial \phi(\mathbf{x}_{ij})}{\partial \mathbf{x}_i}. \quad (4.13)$$

Los esquemas de integración temporal conocidos como “leapfrog” (salto de rana) son equivalentes a calcular las posiciones y las velocidades en pasos temporales intercalados. Por ejemplo, la posición es conocida en pasos de tiempo enteros y la velocidad es conocida en tiempos enteros más medios pasos temporales. Este método es de segundo orden, y por lo tanto funciona mejor que la integración de tipo Euler, la cual es de primer orden (ver p. ej. Press *et al.* 1986, Sec. (19.1), pag. 833-835).

Por lo tanto, al separar la parte potencial del Hamiltoniano en una de corto alcance y otra de largo alcance, se puede obtener una generalización de los esquemas “leapfrog” sub-ciclando la evolución en la parte cinética y la parte de corto alcance obteniendo un operador de evolución temporal

$$\tilde{U}(\Delta t) = K_{largo} \left( \frac{\Delta t}{2} \right) \times \left[ K_{corto} \left( \frac{\Delta t}{2m} \right) D \left( \frac{\Delta t}{m} \right) K_{corto} \left( \frac{\Delta t}{2m} \right) \right]^m K_{largo} \left( \frac{\Delta t}{2} \right), \quad (4.14)$$

donde  $m$  es un entero positivo.

### Integrador para un sistema de N-cuerpos.

En el contexto de integraciones dinámicas usualmente se emplean pasos de tiempo individuales para las partículas. GADGET2 implementa un esquema con pasos de tiempo flexibles similar al usado por Groom (1997) y Hiotelis & Voglis (1991). A continuación se describe este esquema para N-cuerpos y posteriormente se describe el esquema incluyendo la parte SPH.

Cada partícula tiene un paso temporal  $\Delta t_i$ , un tiempo actual  $t_i$ , y su estado dinámico de posición, velocidad y aceleración  $(\mathbf{r}_i, v_i, \mathbf{a}_i)$  se almacena en memoria y puede ser predicho en tiempos  $t \in [t_i \pm 0.5\Delta t_i]$  con una precisión a primer orden. La siguiente partícula  $k$  a ser avanzada es entonces aquella para la que se predijo un tiempo menor, definido como  $\tau_p = \min(t_i + 0.5\Delta t_i)$ . El tiempo  $\tau_p$  se convierte entonces en el nuevo tiempo del sistema. Para avanzar la partícula correspondiente, primero se predicen las posiciones para todas las partículas al tiempo  $\tau_p$  de acuerdo con

$$\mathbf{r}_i^{(n+1)} = \mathbf{r}_i^{(n)} + \mathbf{v}_i^{(n)}(\tau_p - t_i), \quad (4.15)$$

en donde  $n$  indica el paso temporal actual.

Basándose en estas posiciones, la aceleración de la partícula  $k$  a la mitad de su paso temporal es calculada como

$$\mathbf{a}^{(n+1/2)} = -\nabla\Phi(\mathbf{r}_i^{(n+1)})|_{\mathbf{r}_k^{(n+1)}}, \quad (4.16)$$

es decir, se calcula la aceleración debida al potencial ejercido por las partículas  $i$  sobre la partícula  $k$  en las posiciones predichas.

La posición y la velocidad de las partícula  $k$  son entonces avanzadas usando

$$\mathbf{v}_k^{(n+1)} = \mathbf{v}_k^{(n)} + 2\mathbf{a}_k^{(n+1/2)}(\tau_p - t_k), \quad (4.17)$$

$$\mathbf{r}_k^{(n+1)} = \mathbf{r}_k^{(n)} + \left[ \mathbf{v}_k^{(n)} + \mathbf{v}_k^{(n+1)} \right] (\tau_p - t_k), \quad (4.18)$$

y su tiempo actual puede ser actualizado a

$$t_k^{(nuevo)} = t_k + 2(\tau_p - t_k). \quad (4.19)$$

Finalmente, se estima un nuevo paso de tiempo  $\Delta t_k^{(n+1)}$ .

### Avance temporal

Al inicio de la simulación, todas las partículas inician con el mismo paso de tiempo actual. Sin embargo, debido a que los pasos temporales de las partículas son diferentes, los tiempos actuales de las partículas se distribuyen a sí mismos, casi simétricamente alrededor de la predicción actual del tiempo, por lo tanto, la predicción del paso temporal involucra una predicción hacia atrás y hacia adelante en el mismo contexto.

Debido a que es impráctico avanzar una sola partícula, es preferible avanzarlas por grupos. Puede pensarse que las partículas están ordenadas de acuerdo a sus tiempos predcidos  $t_i^{pred} = t_i + \frac{1}{2}\Delta t_i$ . La simulación en GADGET2 opera sobre esta línea temporal y siempre avanza a la partícula con el tiempo predcido  $t_i^{pred}$  menor, así como todas las partículas subsecuentes en la línea temporal, hasta que se encuentra la primera para la cual  $\tau_{pred} = t_i + \frac{1}{4}\Delta t_i$ , esto constituye el esquema de paso temporal flexible del que se hablaba anteriormente.

GADGET2 usa un esquema de actualización dinámica del árbol prediciendo al vuelo las propiedades de los nodos del árbol en lugar de predecirlas para todas las partículas en cada paso de tiempo. Para lograr esto, cada nodo almacena información de la posición y velocidad del centro de masa y predice las nuevas posiciones de los nodos que son “visitados” en el recorrido del árbol. Cuando el centro de masa de un nodo en consideración se ha movido más allá de una pequeña fracción de la longitud lateral del nodo desde la última vez que se reconstruyó dicha parte del árbol, entonces el nodo es completamente actualizado, es decir, se recalcula su centro de masa, su velocidad del centro de masa y los momentos multipolares a partir de las variables de espacio fase individuales de las partículas.

### Integrador incluyendo el método SPH

Para incluir el método SPH dentro de este esquema de integración temporal, se requiere integrar la ecuación de energía interna, así como también se requiere que las aceleraciones de las partículas tengan una componente hidrodinámica. Para calcular esta última, se necesita predecir las velocidades

mediante

$$\mathbf{v}_i^{(n+1)} = \mathbf{v}_i^{(n)} + \mathbf{a}_i^{(n-1)}(\tau_p - t_i), \quad (4.20)$$

notando que se aproxima la aceleración  $\mathbf{a}_i$  con la aceleración del paso de tiempo anterior. De forma similar, se obtienen las predicciones para la energía interna

$$u_i^{(n+1)} = u_i^{(n)} + \dot{u}_i^{(n)}(\tau_p - t_i), \quad (4.21)$$

y la densidad de las partículas inactivas (aquellas que no requieren una actualización de la fuerza en el paso de tiempo actual) como

$$\rho_i^{(n+1)} = \rho_i^{(n)} + \dot{\rho}_i^{(n)}(\tau_p - t_i). \quad (4.22)$$

Para las partículas que serán avanzadas en el paso de tiempo actual, estas cantidades predichas son usadas para calcular la parte hidrodinámica de la aceleración y la tasa de cambio de la energía interna con el estimado usual SPH descrito anteriormente.

### Restricciones en el avance temporal

En las simulaciones astrofísicas, es común enfrentarse con escalas de tiempo con rangos dinámicos muy grandes. Por ejemplo, en las regiones de alta densidad como el núcleo de las NMs, la escala de tiempo es de algunos órdenes de magnitud más pequeños que para las densidades bajas del medio interestelar. Por ello, se requiere de un esquema de integración temporal que considere posibles pasos de tiempo distintos y elija aquél que resulte adecuado para el avance temporal.

Usando la noción de la velocidad  $v_{ij}^{sig}$  de una señal entre dos partículas, el paso temporal hidrodinámico para una partícula está determinado por la *condición de Courant* (ver p. ej. Press *et al.* 1986, Sec. (19.1), pag. 829-830):

$$\Delta t_i^{(hidro)} = \frac{C_{cour} h_i}{\max_j (v_{ij}^{sig})}, \quad (4.23)$$

donde  $C_{cour}$  es una constante numérica,  $v_{ij}^{sig} = c_i + c_j - 3w_{ij}$ ,  $c$  representa la velocidad del sonido en la posición de la partícula y  $w_{ij} = \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} / |r_{ij}|$  es la velocidad relativa proyectada sobre el vector de separación. El máximo es determinado con respecto a todos los vecinos  $j$  de la partícula  $i$ . Esta

condición garantiza que el paso temporal sea menor que el tiempo de cruce de una partícula a otra a la velocidad de propagación de la información entre ellas.

Por otro lado, en GADGET2, el criterio considerado para fijar el paso temporal para partículas sin colisiones es de la forma

$$\Delta t_{grav} = \min \left( \Delta t_{max}, \sqrt{\frac{2\eta\epsilon}{|\mathbf{a}|}} \right), \quad (4.24)$$

donde  $\eta$  es un parámetro de exactitud,  $\epsilon$  es una longitud de escala empleada para suavizar el potencial gravitacional y  $\mathbf{a}$  es la aceleración de la partícula. El paso de tiempo máximo permitido es  $\Delta t_{max}$  y su valor usualmente es una fracción pequeña del tiempo dinámico del sistema bajo estudio. Para partículas SPH, GADGET2 extiende este criterio, considerando la condición de Courant y eligiendo la menor de las dos.

### 4.3. Versión con enfriamiento, calentamiento y partículas sumidero (CRyA)

Adicionalmente a las características descritas hasta aquí, la versión GADGET2 en la que se ha implementado la retroalimentación es la descrita en Vázquez-Semadeni *et al.* (2007) y Vázquez-Semadeni *et al.* (2009) y está enfocada principalmente a la simulación del gas atómico en el MI. A continuación se describen brevemente sus características particulares.

#### 4.3.1. Enfriamiento y calentamiento

Para simular correctamente la dinámica del MI se requiere de una función de enfriamiento  $\Lambda$  y una función de calentamiento  $\Gamma$  que describan las ganancias y pérdidas de energía interna, respectivamente, por parte del gas. Koyama & Inutsuka (2002) parametrizaron una variedad de procesos de enfriamiento atómicos y moleculares calculadas previamente por ellos mismos

en Koyama & Inutsuka (2000). La parametrización de los varios de los procesos de enfriamiento y calentamiento, aplicables a las densidades típicas del gas atómico es (Vázquez-Semadeni *et al.* 2007; ver fig. 4.3):

$$\Gamma = 2.0 \times 10^{-26} \text{ erg s}^{-1},$$

$$\frac{\Lambda(T)}{\Gamma} = 10^7 \exp\left(\frac{-1.184 \times 10^5}{T + 1000}\right) + 1.4 \times 10^{-2} \sqrt{T} \exp\left(\frac{-92}{T}\right) \text{ cm}^3. \quad (4.25)$$

El procedimiento usual para aplicar el enfriamiento a la evolución hidrodinámica involucra la consideración de la tasa de enfriamiento en la condición de Courant para restringir el paso temporal de la simulación. Sin embargo, dado que el tiempo de enfriamiento puede ser prohibitivamente corto, es más conveniente usar una aproximación a la evolución térmica del gas, permitiendo simplemente corregir la energía interna después de efectuarse paso hidrodinámico, sin necesidad de ajustar el paso temporal, por lo cual dicha aproximación ha sido implementada en GADGET2. Esta corrección se realiza interpolando la energía interna del gas a la densidad actual entre su valor adiabático y su valor de equilibrio  $n\Lambda(T) = \Gamma$  como:

$$e' = e_{eq} + (e - e_{eq}) \exp(-dt/\tau_\Lambda), \quad (4.26)$$

en donde  $\tau_\Lambda$  es el tiempo de enfriamiento:

$$\tau_\Lambda = \frac{e - e_{eq}}{n\Lambda - \Gamma}. \quad (4.27)$$

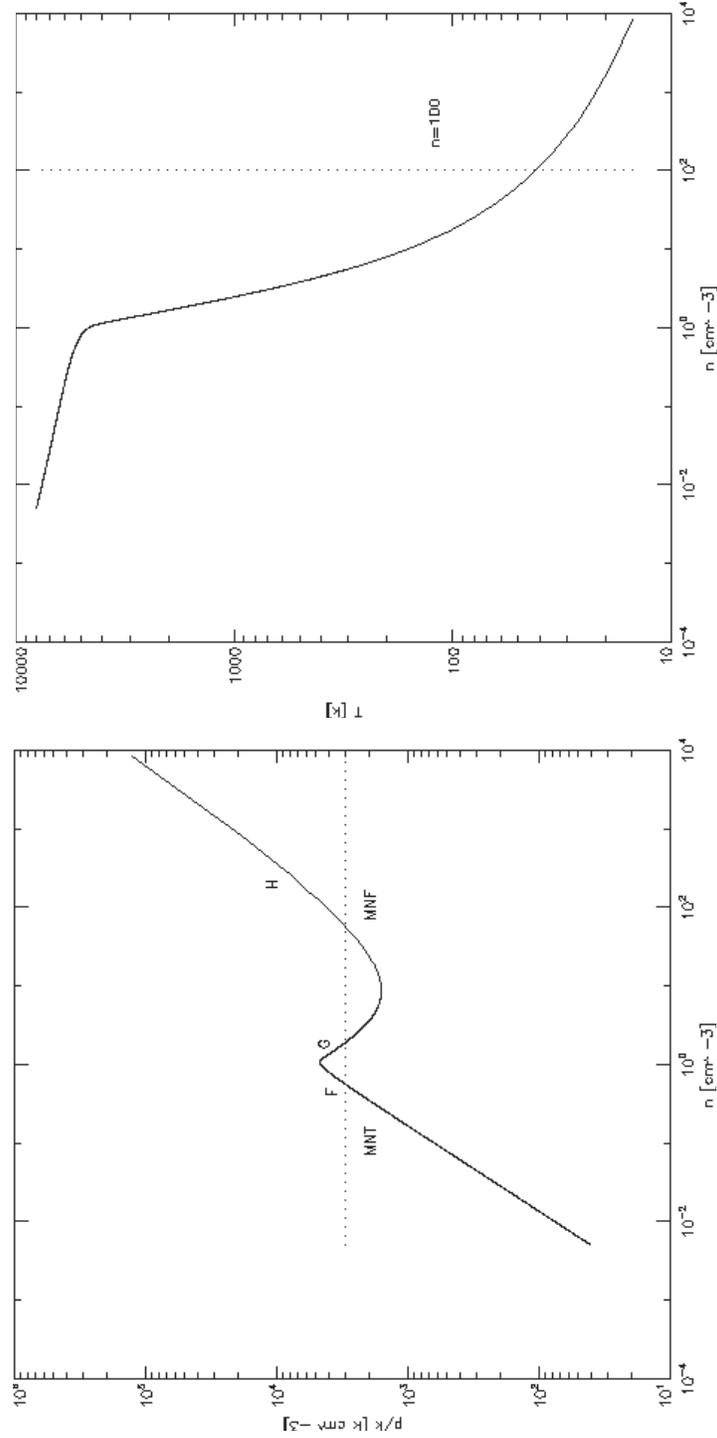


Fig. 4.3: La presión térmica (izq.) y la temperatura (der.) de equilibrio térmico contra densidad para las funciones de enfriamiento y calentamiento dadas por la ec. (4.3.1). Las regiones etiquetadas F, G y H representan las fases encontradas por Field, Goldsmith y Habing respectivamente.

### 4.3.2. Partículas sumidero (“sinks”)

Cuando en una simulación numérica se produce un colapso gravitacional, el paso temporal se hace arbitrariamente pequeño, ya que en este caso está dado por el tiempo de caída libre [el tiempo en el que se alcanza la singularidad; ec. (3.5)] que decrece conforme aumenta la densidad local. Para evitar que el paso temporal disminuya sin límite, se ha implementado la técnica de *partículas sumidero* (“sink”), consistente en reemplazar los centros de colapso gravitacional por una única partícula, lo cual permite continuar la simulación aún entre la presencia de múltiples colapsos locales.

Siguiendo el criterio de Bate *et al.* (1995), se define un volumen de control para la partícula sumidero mediante un radio fijo provisto como un parámetro de la simulación llamado *radio de acreción*. Entonces, si la densidad correspondiente a una partícula dada excede una cierta densidad crítica  $n_{SkF}$ , se ejecutan una serie de pruebas para determinar si se crea una partícula sumidero. Primero, se determina si la longitud de suavizado de la partícula en cuestión es menor que la mitad del radio de acreción, con el propósito de garantizar que el sink se formará con un número suficientemente grande de partículas SPH. Segundo, el cociente de la energía térmica a la magnitud de la energía gravitacional debe ser menor o igual a 0.5. Tercero, se requiere que la energía total sea menor que cero. Finalmente, la divergencia de las aceleraciones en las partículas debe ser menor que cero. Si todas estas pruebas se pasan, la partícula con la más alta densidad se convierte en una partícula sumidero, con la posición, velocidad y aceleración derivadas a partir de los valores del centro de masa de las partículas de gas originales (ver también Jappsen *et al.* 2005).

Para determinar si una partícula es posteriormente acretada a un sink ya existente o no, se consideran tres criterios. Primero, la partícula debe estar ligada a la partícula sumidero, es decir, su energía cinética debe ser menor que la magnitud de su energía gravitacional. Segundo, el momento angular específico de la partícula debe ser menor que el requerido para moverse en una órbita circular de radio  $r_{acc}$  alrededor de la partícula sumidero y, finalmente, la partícula debe estar más fuertemente ligada a la partícula sumidero candidata que a otras partículas sumidero.

En el apéndice (A.1) se encuentra un diagrama de flujo de GADGET2

donde se indican los módulos relevantes del código.

## Capítulo 5

# Implementación de la retroalimentación por calentamiento debido a radiación ionizante

En la presente tesis se ha implementado en GADGET2 una forma simplificada del algoritmo propuesto por Dale *et al.* (2007) para la simulación de calentamiento por fotoionización en el contexto SPH, que permite simular la formación y evolución de regiones HII dentro de una nube molecular. A continuación se describe dicho algoritmo y se mencionan los detalles de su implementación y las pruebas realizadas.

### 5.1. Algoritmo de Dale *et al.* (2007)

El algoritmo de fotoionización recorre todas las partículas SPH de una simulación, tomando secuencialmente a cada una como “partícula objetivo” y determinando si los fotones ionizantes provenientes de una fuente de radiación la alcanzan para ionizarla dentro del paso de tiempo actual o no; es decir, si se encuentra dentro del radio de Strömgren de la fuente ionizante. Esta

### 5.1 Algoritmo de Dale et al. (2007) Implementación de la retroalimentación

---

prueba se debe realizar durante todo el tiempo que la estrella ionizante pasa dentro de la secuencia principal. Las partículas pueden ser consideradas como completamente ionizadas o neutras. Una vez que el algoritmo ha decidido que la partícula es ionizada, se incrementa su temperatura a un valor de  $10^4$  K y se regresa el control a la parte dinámica del código. Las partículas ionizantes son las partículas sumidero masivas cuyo flujo ionizante está determinado por su masa, la cual puede crecer por acreción.

En una nube ópticamente gruesa, una buena aproximación al considerar la radiación difusa producida por la emisión del gas ionizado es que los fotones del campo de radiación difuso son absorbidos en alguna otra parte de la nube y ninguno de ellos puede escapar. La aproximación conocida como “*on the spot*” considera que esta misma situación ocurre localmente en cualquier región de la nube (ver p. ej. Osterbrock 1989, pags 18-20). En una nube de gas consistente de hidrógeno puro y caracterizada por las condiciones donde la aproximación “*on the spot*” es válida, los fotones secundarios no contribuyen al equilibrio de ionización total. Los fotones que controlan el balance de ionización son, por lo tanto, los fotones ionizantes primarios, los cuales viajan radialmente hacia afuera de la fuente de radiación hasta que son absorbidos. Debido a que los fotones siguen trayectorias puramente radiales desde la fuente a través del gas, el destino de cada fotón es independiente del de aquéllos que viajan a lo largo de otras trayectorias radiales. Esto tiene la consecuencia de que, en una nube no uniforme, cada fotón puede ser tratado como si se moviera a través de una nube esféricamente simétrica con un perfil de densidad igual al que existe a lo largo de la dirección de propagación del fotón.

Si la nube en cuestión es suficientemente densa en todas partes, de tal forma que la aproximación “*on the spot*” se mantiene, la estructura de ionización alrededor de la fuente puntual de radiación puede ser encontrada por el método del *volumen de Strömgren*, en el cual la distancia desde la fuente al frente de ionización en cualquier dirección dada puede ser encontrada usando el perfil de densidad radial en esa dirección. Vale la pena notar que en este punto, y en el caso de múltiples fuentes ionizantes con esferas de Strömgren traslapadas, el algoritmo descrito no puede reproducir el campo de radiación en la región traslapada. Por lo tanto, la masa ionizada podría resultar subestimada.

En una nube esférica con un perfil radial de densidad arbitrario  $n(r)$ , el flujo de ionización a un radio  $R_I$  está dado por  $Q_H - F$ , donde  $Q_H$  es el flujo de fotones ionizantes provenientes de la fuente (donde cualquier fotón cuya energía excede 13.6 eV es considerado ionizante) y  $F$  es la *integral de Strömngren* dada por

$$F = \int_0^{R_I} 4\pi r^2 n(r)^2 \alpha_B dr \quad (5.1)$$

y que representa el número de fotones absorbidos en un volumen de radio  $R_I$ . Por lo tanto, el flujo que llega a cualquier partícula dada puede ser encontrado si se conoce el perfil de la densidad numérica  $n(r)$  a lo largo de la línea de visión desde el sink hasta la partícula. El método se ilustra en la figura 5.1 y consiste en una versión modificada del método originalmente propuesto por Kessel-Deynet & Burkert (2000), que permite evaluar la función  $n(r)$ . Primero, se selecciona una partícula SPH como objetivo y se dibuja una línea de visión desde esta partícula hasta la fuente. A continuación, se buscan los vecinos de la partícula objetivo (aquéllos dentro del círculo punteado de la figura) a partir de la lista de vecinos disponible en GADGET2 y se selecciona el más cercano a la línea de visión (etiquetado con el número 4 en la figura). La posición de la partícula seleccionada es entonces proyectada a la línea de visión para producir un *punto de evaluación*, etiquetado  $r_4$  en la figura. Posteriormente, el proceso de selección de vecinos se repite alrededor de la partícula previamente seleccionada, hasta alcanzar la fuente de radiación. Denotamos por  $N_{LV}$  al número total de partículas seleccionadas a lo largo de la línea de visión. Una vez que este proceso ha sido completado, es posible determinar el perfil de densidad a partir de las densidades de las  $N_{LV}$  partículas SPH seleccionadas a lo largo de la línea de visión.

Se puede a continuación conocer el valor de  $n(r)$  en una serie de  $N_{LV}$  intervalos (“bins”) radiales cuyos radios interno y externo están definidos por las posiciones proyectadas de las partículas a lo largo de la línea de visión. Suponiendo que las densidades de estas posiciones proyectadas son simplemente las mismas que las de la partícula correspondiente  $i$ , se puede aproximar la integral (5.1) por una suma:

$$\frac{F}{4\pi} = \sum_{i=1}^{N_{LV}} r_i^2 \langle n(r) \rangle^2 \alpha_B \Delta r_i. \quad (5.2)$$

El radio interno del *bin*  $i$ ,  $r_{i-1}$ , es definido por la posición radial de la partícula

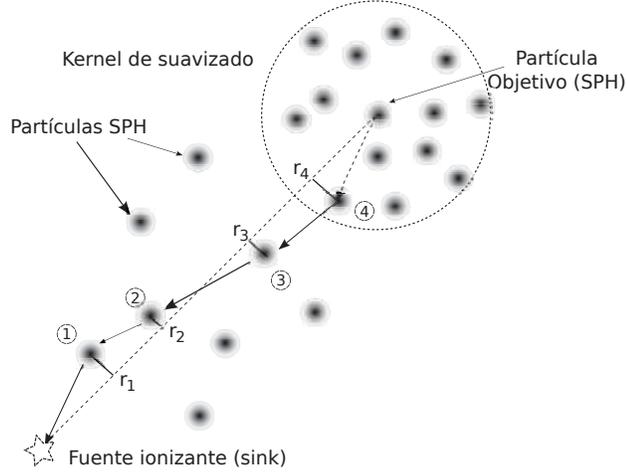


Fig. 5.1: Ilustración del método usado para seleccionar las partículas cuyas densidades son usadas para derivar el perfil de densidad a lo largo de la línea de visión entre la fuente de radiación y una partícula objetivo dada, denotada por la línea discontinua. Las distancias  $r_i$  están medidas desde la fuente ionizante.

$i - 1$  proyectada sobre la línea de visión; de la misma forma se procede para el radio externo  $r_i$  con la partícula  $i$ . El ancho de cada *bin* es entonces  $r_i - r_{i-1} = \Delta r_i$ . La densidad numérica promedio de las dos partículas SPH cuyas posiciones definen el *bin* es  $\langle n(r) \rangle = (n_i + n_{i-1})/2$ . La densidad de masa puntual, usada en el cálculo de la densidad numérica del primer *bin* se considera como 0. Una vez que la suma ha sido calculada, se puede sustraer  $F$  de la luminosidad de la fuente  $Q_H$  para obtener el flujo de fotones que llegan a la partícula objetivo, determinando así el estado de ionización de la partícula en el paso temporal actual. Es decir, si  $Q_H - F \leq 0$ , se considera que la partícula está ionizada, o bien, permanece neutra en caso contrario.

## 5.2. Incorporación a GADGET2

Una vez que se hubo analizado el funcionamiento interno y el diagrama de flujo (ver apéndice. A.1) de GADGET2, en su versión con partículas sumidero, así como la adaptación para los procesos de enfriamiento y calentamiento,

se determinó que el lugar del código donde se insertaría el algoritmo de retroalimentación por fotoionización es después de la rutina que implementa las modificaciones para el enfriamiento y calentamiento del gas y antes de hacer el vaciado de archivos de instantáneas (“snapshots”) con los datos de la simulación.

Con el propósito de implementar el algoritmo antes mencionado, llevamos a cabo la incorporación del código dentro de GADGET2, subdividiendo el problema físico en los siguientes pasos:

- I) Lectura de parámetros y condiciones iniciales.
- II) Cálculo de la FMI para las partículas sumidero.
- III) Encendido y apagado de estrellas en partículas sumidero.
- IV) Retroalimentación por fotoionización.

A continuación describimos estos pasos en detalle.

### 5.2.1. Lectura de parámetros y condiciones iniciales

Muchas de las características de GADGET2, así como de la física involucrada en una simulación, son controladas a través del archivo de parámetros `paramfile`, el cual debe ser especificado al iniciar la simulación. Dentro de este archivo, cada parámetro es definido por una etiqueta, seguida por un valor numérico o una cadena de caracteres.

En nuestra implementación, dentro del módulo `sink_feedback.c` existe una rutina llamada `SFB_parameters()` que especifica parámetros de control y depuración e inicializa constantes físicas y factores de conversión de unidades físicas. Adicionalmente, se imprime la salida de algunos valores útiles en los archivos de texto `log` y `trace`. Esto último, con el propósito de habilitar el rastreo en la selección de partículas y su ionización (ver listado de archivos con el código generado en los apéndices A.2.1 y A.2.2).

### 5.2.2. Cálculo de la FMI para las partículas sumidero

Debido a que requerimos simular la ionización del gas, primero debemos determinar cómo se distribuiría la masa en estrellas “dentro” de las partículas sumidero. Para esto, suponemos una función de masa inicial estelar de Kroupa (2001) dada por la ec. (3.17), con

$$\alpha = \begin{cases} -2.3 & (M \geq 0.5M_{\odot}), \\ -1.3 & (0.08 \leq M < 0.5M_{\odot}), \\ -0.3 & (0.01 \leq M < 0.08M_{\odot}). \end{cases} \quad (5.3)$$

A partir de esta función podemos calcular, normalizando la FMI a la masa del sink, cuántas estrellas de masa  $M$  “cabén” en la partícula sumidero y, de esta forma, podemos asignarle a la partícula sumidero el flujo de fotones ionizantes correspondiente a la estrella más masiva que “cabe” en ella (ver tabla 5.1).

Tipo estelar	Masa [ $M_{\odot}$ ]	Luminosidad [ $L_{\odot}$ ]	Flujo [ $\times 10^{48} \text{ s}^{-1}$ ]
O5	60	499000	51
O8	23	147000	3.9
B1	18	9950	0.0033

Tabla 5.1: Datos estelares.  $M_{\odot} = 1.9891 \times 10^{33} \text{ g}$ ,  $L_{\odot} = 3.839 \times 10^{33} \text{ erg s}^{-1}$

La fracción de la masa de la partícula sumidero que está en estrellas de una masa dada se calcula en la rutina `SFB_calculate_imf()`, normalizando el área bajo la curva de la FMI a la masa de la partícula sumidero.

### 5.2.3. Encendido y apagado de estrellas en partículas sumidero

Una vez que ha sido asignada la cantidad de estrellas en cada intervalo de masa que posee una partícula sumidero, se contabiliza el tiempo durante el

cual están radiando, permitiendo la posibilidad de “apagarlas” cuando salen de la secuencia principal. Esto se hace inmediatamente antes de ionizar las partículas de gas, pues en el caso de que la masa de la partícula sumidero cambie por acreción, se requiere una actualización del número de estrellas dentro de la partícula sumidero.

#### 5.2.4. Retroalimentación por fotoionización

Para simular la formación de la región HII se requiere determinar qué partículas SPH son “ionizadas”. Por ello, primero se lleva a cabo una ordenación de las partículas sumidero en orden creciente de su distancia a la partícula objetivo. Enseguida se hace un recorrido desde la partícula objetivo hasta la partícula sumidero mediante el algoritmo antes descrito (en la sección 5.1), seleccionando las partículas más cercanas a la línea de visión. Una vez seleccionadas, se hace un recorrido inverso desde la partícula sumidero hasta la partícula objetivo para determinar el flujo de fotones ionizantes disponible mediante el cálculo de la densidad columnar, empleando la proyección de su posición sobre la línea de visión y su densidad. Si el flujo es insuficiente en algún punto del recorrido se deja intacta a la partícula SPH. Si por el contrario, al llegar al final del recorrido el flujo es positivo, se eleva la temperatura de la partícula SPH a  $10^4$  K. Una vez que la partícula es “ionizada” ya no es necesario continuar con las demás partículas sumidero previamente ordenadas. Ver diagrama de flujo del algoritmo en la fig. (A.1).

La parte que más tiempo computacional consume dentro del algoritmo es la determinación de las partículas SPH más cercanas a la línea de visión. Primero se calcula el vector que va desde la partícula objetivo hasta la partícula sumidero y enseguida, para cada partícula intermedia previamente seleccionada, se encuentran las partículas SPH vecinas dentro de su longitud de suavizado. Para cada una de estas partículas se calcula su distancia a la línea de visión definiendo un vector a partir de la partícula objetivo y calculando el producto escalar entre los dos vectores para determinar su proyección sobre la línea de visión (punto de evaluación), es decir, el avance a partir de la partícula sumidero. Luego, se determina la distancia desde este punto a la partícula vecina, seleccionando la que tiene una distancia menor y repitiendo el proceso durante todo el recorrido.

En una simulación con condiciones periódicas, la caja computacional principal tiene en cada una de sus caras una caja computacional vecina, lo cual, requeriría hacer la búsqueda de las partículas vecinas en cada una de estas cajas incrementando considerablemente el costo numérico del algoritmo al menos por un factor de  $\sim 8$ . Los cálculos se incrementan por el mismo factor al incrementar la resolución, duplicando el número de partículas por dirección distribuidas uniformemente dentro de la caja. Por esta razón, se optó por no emplear fronteras periódicas e implementar el algoritmo en forma serializada, sin paralelizarlo en primera instancia. Además, se usaron ampliamente los arreglos como estructuras de datos eficientes (para el acceso) en la lista de partículas seleccionadas. También se emplearon variables adicionales para almacenar las distancias proyectadas sobre la línea de visión y la información de ésta misma, con el propósito de evitar cálculos innecesarios entre dos partículas seleccionadas en forma consecutiva.

En el tipo de simulaciones que se planea realizar, (evolución de nubes moleculares a lo largo de decenas de Mega años), Por otra parte, en GADGET2 no es posible resolver la fase de formación de la región HII (descrita en la sec. 2.1.4), ya que en términos del tiempo evolutivo de una NM es prácticamente instantánea. Por otra parte, la ionización de los átomos producida dentro de la región HII cambia la densidad numérica en un factor de 2 al separarlos en protones y electrones, lo cual implicaría disponer de una implementación en el código del esquema de separación de partículas, el cual no está disponible en GADGET2. Por ello, se implementó únicamente la fase de expansión (descrita en la sec. 2.1.4).

# Capítulo 6

## Pruebas y resultados

### 6.1. Parámetros de la simulación

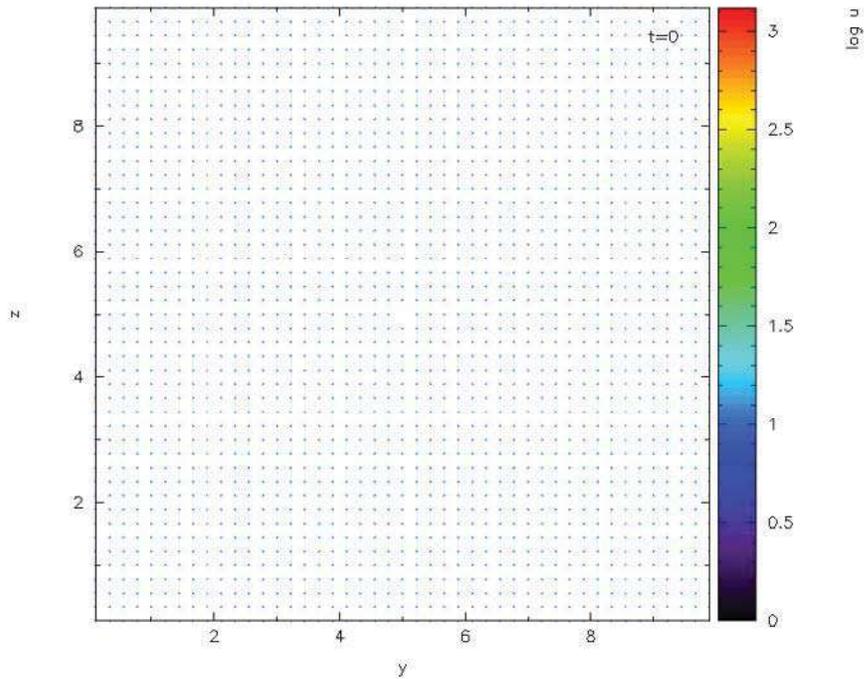
Las condiciones iniciales de la simulación consisten en un medio uniforme de gas frío ( $n = 100 \text{ cm}^{-3}$  y  $T = 100.0 \text{ K}$ ), con temperatura de equilibrio del gas de  $T = 41.769 \text{ K}$  y sometido a los procesos de enfriamiento y calentamiento descritos por las ecs. 4.3.1, con un peso atómico promedio de  $\mu = 2.36$ , correspondiente a las condiciones de una NMG, con cociente de calores específicos  $\gamma = 5/3$ .

Para simular la región HII, se creó manualmente una partícula sumidero de  $100M_{\odot}$  en las condiciones iniciales, ubicándola al centro de una caja numérica no periódica, de tamaño  $L = 10 \text{ pc}$  con  $45^3 = 91125$  partículas distribuidas uniformemente, cada una de  $0.063889M_{\odot}$ , y fijando en 40 el número promedio de partículas SPH dentro del volumen de suavizado.

Empleando la descripción de Kroupa para la FMI descrita en la sec. (3.2.3) se calculó que para la partícula sumidero, el 0.234 de su masa es convertido una única estrella de  $23M_{\odot}$  (tipo O8). Para dicha estrella, se calculó un radio de Strömngren teórico de  $R_{S,teo} = 2.19 \text{ pc}$ . Además, se ponen la densidad crítica para la formación de las partículas sumidero y el radio de acreción en  $n_{umb} = 1.0 \times 10^{20} \text{ cm}^{-3}$  y  $r_{acc} = 0.04 \text{ pc}$ , respectivamente. La simulación no considera la autogravedad del gas.

### 6.1.1. Interpretación cualitativa de la simulación

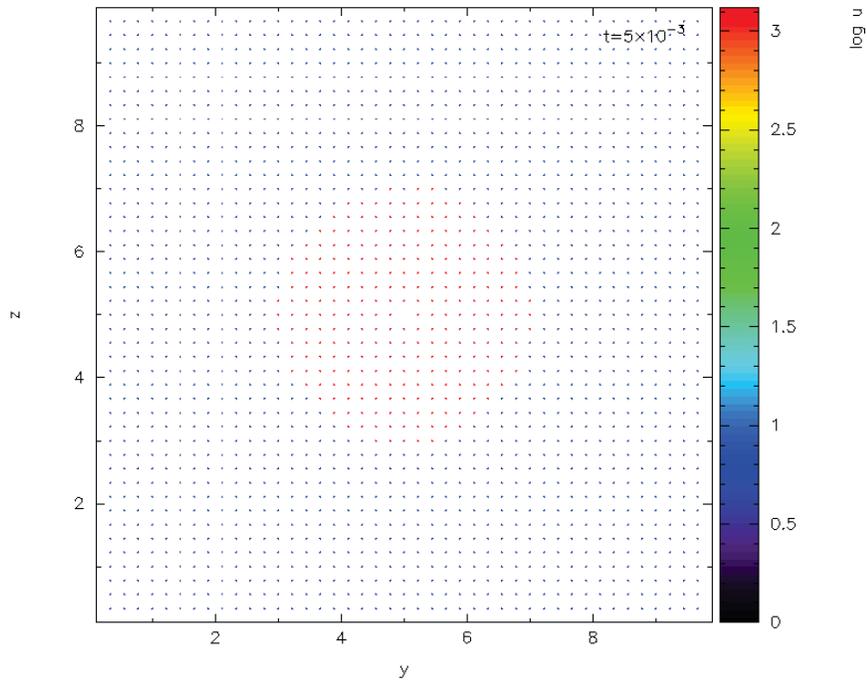
Enseguida se da la interpretación cualitativa de la simulación, en el sentido de las partículas SPH mostradas en la fig. (6.1). Cada uno de los puntos interpretados, se indican con un prefijo “c” (de cualitativa) y un número romano entre paréntesis para hacer referencia a ellos posteriormente en un análisis más completo de la simulación. Las imágenes en la fig. (6.1) muestran la gráfica del logaritmo de la energía interna por unidad de masa de las partículas ( $\log u$ , en unidades del código) en un corte transversal a  $x = 5$  pc, así como el tiempo de evolución (en unidades del código).



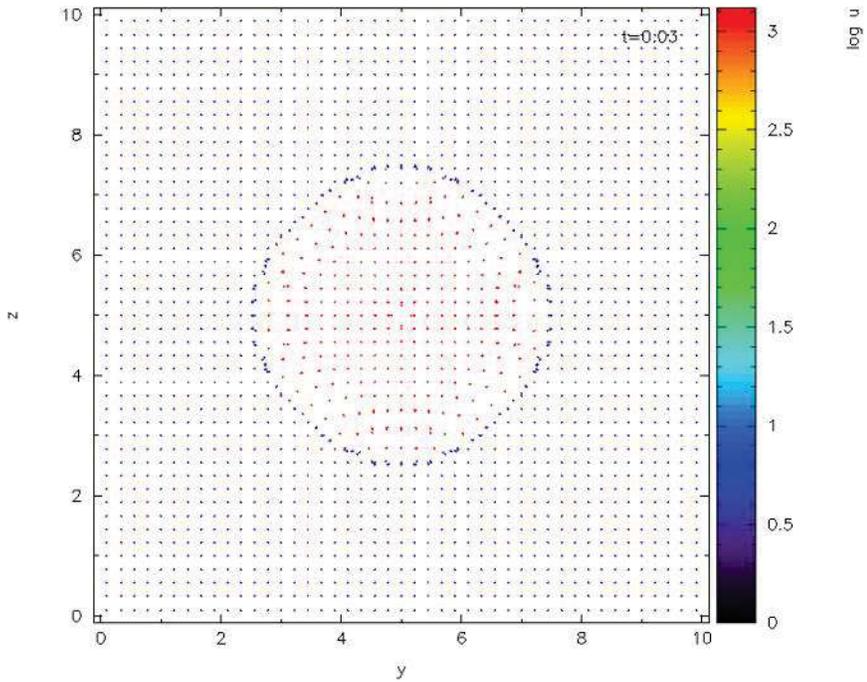
(a)

Fig. 6.1: Evolución de la región HII. a) Condiciones iniciales

En la fig. (6.1a) se muestran las partículas SPH de gas “neutro” (en azul) en las condiciones iniciales y con una distribución cartesiana dentro de la caja computacional. La temperatura de las partículas corresponde a la tempera-

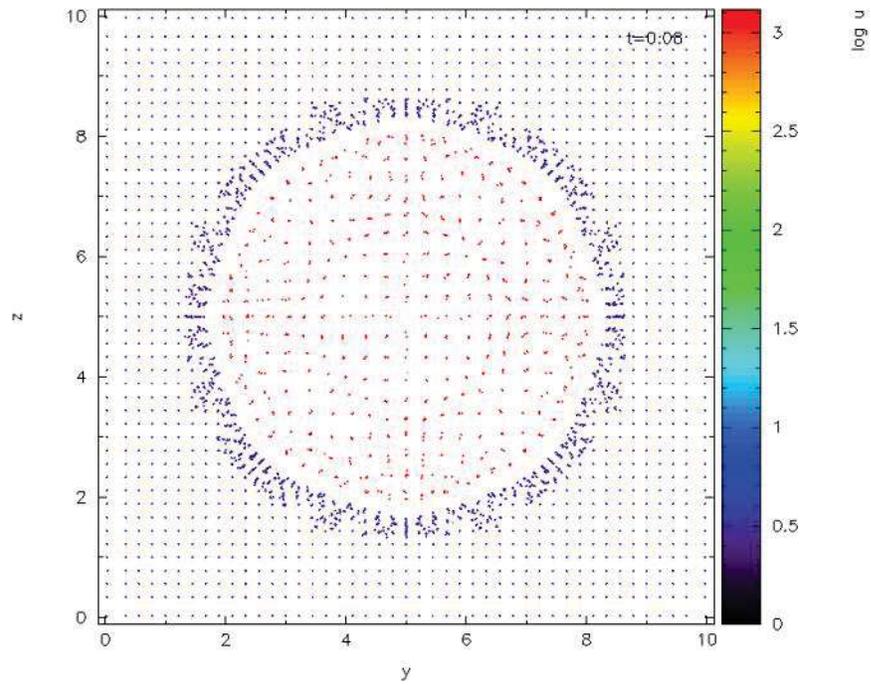


(b)

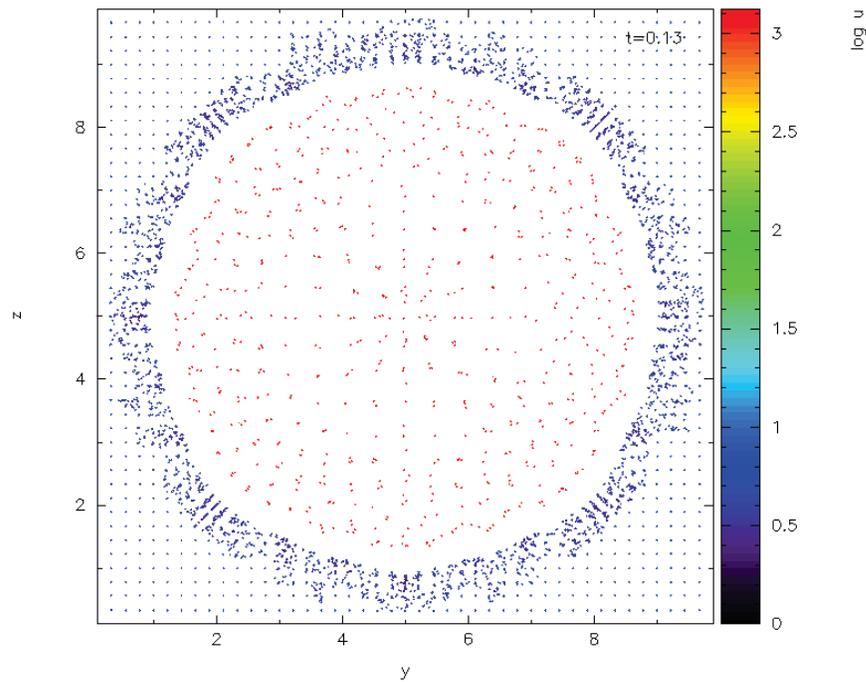


(c)

Fig. 6.1: Evolución de la región HII (cont.) b) Formación de la esfera de Strömngren, c) Identificación de la onda de choque.



(d)



(e)

Fig. 6.1: Evolución de la región HII (cont.) d) y e) evolución intermedia y final del avance en una caja de  $L = 10$  pc.

tura de equilibrio del gas según las funciones de enfriamiento y calentamiento provistas por Koyama & Inutsuka (2002) e implementadas en GADGET2.

(c.I) En la fig. (6.1b) puede observarse claramente que las partículas SPH (en rojo) han sido “ionizadas” por el algoritmo, elevando su temperatura a  $10^4$  K dentro de la esfera de Strömgren alrededor de la partícula sumidero ubicada al centro de la caja, mientras que fuera de la esfera, las partículas se mantienen neutras y sin movimiento. Obsérvese que la región ionizada no parece ser completamente esférica, lo cual es un artificio de la distribución cartesiana de las partículas y, principalmente, a la baja resolución ( $N = 45$  partículas por dirección) en la simulación. Sin embargo, se aprecia una simetría radial aceptable.

(c.II) En la fig. (6.1c), el rasgo más notable es que en la frontera de la esfera de Strömgren comienza a desarrollarse una onda de choque delgada, formada por partículas neutras. Dentro de la esfera, algunas partículas ionizadas se han desplazado de su posición inicial, lo cual indica que la esfera ya se ha expandido, dando origen a la capa chocada.

(c.III) En la fig. (6.1d), se muestra una etapa intermedia de la evolución de la región HII, con un avance aproximadamente a la mitad del tamaño de la caja. Aquí se observa que las partículas SPH cercanas a la frontera de la caja tienden a “escaparse”, debido a que en la simulación se consideran condiciones no periódicas.

(c.IV) En comparación con la fig. (6.1c), en la fig. (6.1d) se aprecia claramente que la esfera de Strömgren se ha expandido debido al gradiente de presión entre las partículas ionizadas y neutras.

(c.V) Como puede observarse en la fig. (6.1d), conforme el gas es aglomerado afuera de la esfera de partículas ionizadas, la onda de choque se engrosa y muestra claramente dos bordes. El borde exterior muestra una baja definición, ya que las partículas neutras empiezan a ser aglomeradas por la onda de choque, mientras que en el borde interior hay una mejor definición, pues las partículas neutras han sido aglomeradas por más tiempo, acomodándose alrededor de la esfera de partículas ionizadas.

(c.VI) Es notable también que, entre la parte exterior de la esfera y el borde interior de la onda de choque, existe un hueco de partículas, pues

las partículas ionizadas en la frontera de la esfera comienzan a desplazarse hacia afuera por la expansión antes mencionada, mientras que las partículas en zonas más interiores de la esfera se mantienen sin movimiento pues las partículas ionizadas más externas se lo impiden. Compárese esta situación con la de una larga fila de vehículos que avanzan progresiva y lentamente después de la luz verde de un semáforo. Es decir, teniendo como marco de referencia el borde interior de la onda de choque, la expansión las partículas ionizadas se propaga de afuera hacia adentro. Al comparar la fig. (6.1e) con fig. (6.1d), esto se aprecia más claramente dentro de la esfera ionizada, pues mientras las partículas más centrales poseen un movimiento casi nulo, las más externas poseen movimientos grandes, como resultado de la propagación antes mencionada.

(c.VII) En la fig. (6.1e) también es apreciable la disminución de la densidad en gran medida dentro de la esfera ionizada, como resultado de la expansión de la esfera. Por el contrario, conforme evoluciona la región HII se incrementa cada vez más la densidad de partículas en la onda de choque.

(c.VIII) Si se observa detalladamente la fig. (6.1e), se aprecia que en esta etapa de la evolución, en las regiones dentro de la onda de choque donde las partículas SPH están más aglomeradas hay una ligera la variación de color en las partículas, lo cual, representa un cambio de temperatura notable. Este debe ser el efecto de someter las partículas SPH a las funciones de enfriamiento y calentamiento antes mencionadas, que dependen esencialmente de la temperatura y la densidad.

## 6.2. Prueba del algoritmo

Considerando que el volumen que ocupa la región HII es aquél que corresponde al radio donde se ubica el frente de ionización, se puede analizar la expansión y llevar a cabo la prueba del código comparando los resultados con la solución analítica provista por las ecs. (2.25) y (2.47).

El proceso mediante el cual se determina la ubicación del frente de ionización consiste en calcular el promedio entre la posición de las 20 partículas de gas ionizadas más lejanas y neutras más cercanas a la partícula sumidero.

Esto fue necesario debido a que la posición del FI no se encuentra espacialmente bien definida.

La evolución del FI se describe en la fig. (6.2), la cual muestra la posición del FI como función del tiempo para tres simulaciones con diferentes resoluciones (6.1). En esta gráfica, las barras de error representan la dispersión en las posiciones de las partículas usadas para determinar la posición de FI, como se describe al inicio de esta sección. En esta figura se aprecia que, el FI se mantiene siempre por debajo de la solución analítica. Para determinar el origen de este comportamiento, se procedió a verificar el cumplimiento de las hipótesis (sec. 2.1.4) empleadas en el desarrollo de la solución analítica en la fase de formación de la región HII, encontrándose que las primeras tres implican la formación de una onda de choque. Sin embargo, al ver la fig. (6.1.b) se observa que al inicio de la simulación aún no se ha formado una onda de choque, por lo cual no se cumplen las hipótesis.

En la fig. (6.2) se observa que el FI no comienza a avanzar sino hasta que  $t > 0.99$  Myr. Esto es debido a que, al comienzo de la simulación, GADGET2 calcula el paso de tiempo antes de que el algoritmo de retroalimentación se ejecute. Debido a que este paso de tiempo se calcula para condiciones uniformes, éste es más largo que el tiempo entre salidas de datos. El código genera entonces, salidas de datos repetidas hasta alcanzar el tiempo físico de evolución. Por ello, GADGET2 no inicia realmente la evolución del sistema hasta  $t = 0.99$  Myr. Este problema es específico a las condiciones iniciales usadas para esta prueba y no se espera que tenga una influencia importante en una situación más realista. Entonces, podemos considerar como punto de partida de la evolución en la simulación la primera salida de datos que sucede a  $t = 1.32$  Myr, ver fig. (6.3).

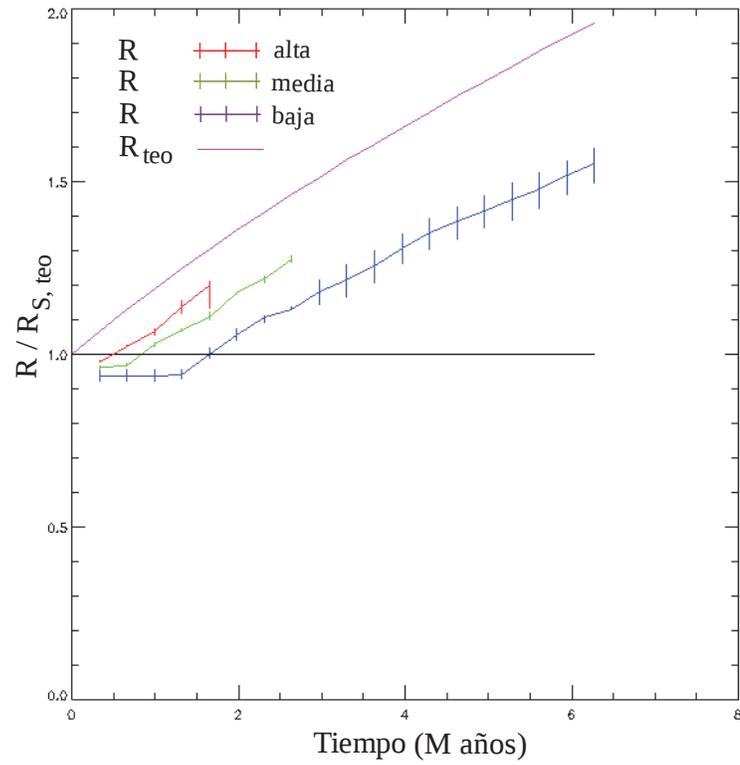


Fig. 6.2: Evolución del FI durante la fase de expansión de una región HII producida por una estrella tipo O8 (se muestran las evoluciones con distintas resoluciones)

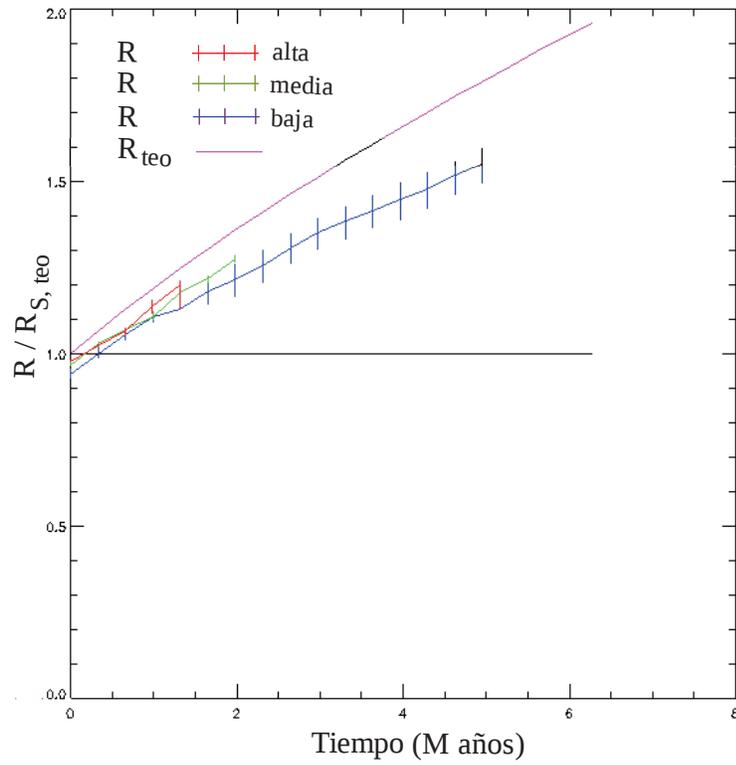


Fig. 6.3: Corrección de tiempo muerto en la evolución del FI.

Como puede observarse en las figs. (6.2) y (6.3), las barras de error para la ubicación del frente de ionización son relativamente grandes, lo que sugiere incrementar la resolución para ubicar con más precisión el frente de ionización. Por lo tanto, se corrieron otras dos simulaciones a mediana y alta resolución respectivamente, variando el número de partículas dentro de la caja numérica (ver tabla 6.1).

Simulación	No. de parts.	Masa por partícula [ $M_{\odot}$ ]
LO8	91125	0.063889
MO8	729000	0.007986
HO8	2460375	0.002366

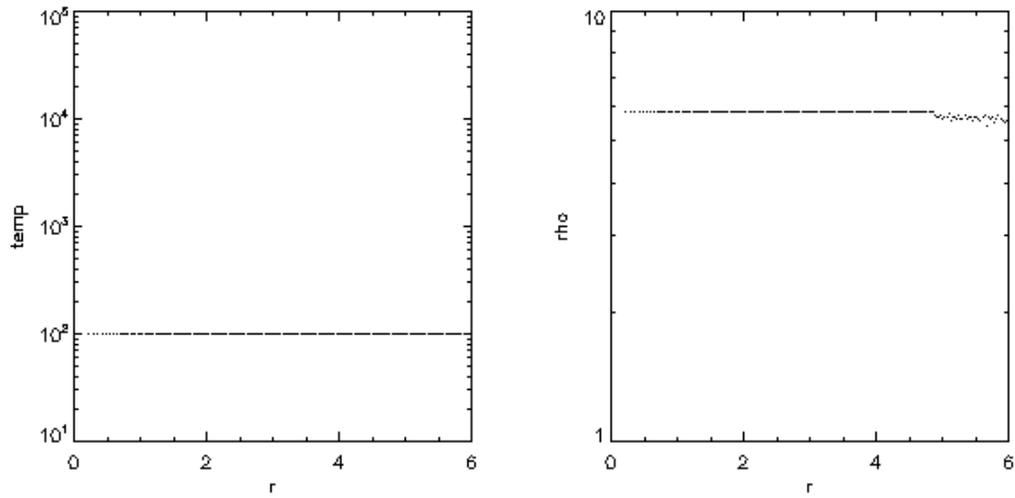
Tabla 6.1: Parámetros de las simulaciones

Con estas correcciones, en las simulaciones se ubicó la posición inicial del frente de ionización en  $\sim 0.941, 0.968$  y  $0.978 R_{S\ teo}$  respectivamente. Al analizar los datos, se observó que conforme se aumenta la resolución, se mejora la aproximación, aunque siempre queda por debajo de la solución analítica, aún considerando las barras de error. Esto significa que la solución numérica no ha convergido aún a la solución analítica, aún a la más alta resolución. De cualquier forma, observamos que a la máxima resolución utilizada, el error cometido al final de la simulación es de sólo 4%, por lo que consideramos satisfactoria la solución numérica del problema.

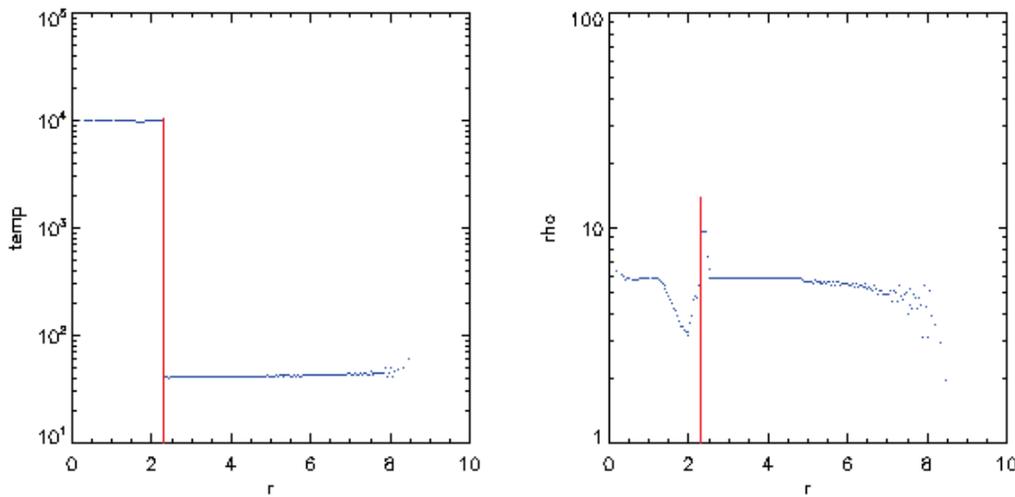
### 6.2.1. Interpretación física de la simulación

Enseguida se interpreta la simulación basada en los perfiles de densidad y temperatura mostrados en la fig. (6.4), identificando los procesos físicos involucrados en la evolución de la región HII y haciendo una referencia a los puntos de la interpretación cualitativa. En todas las imágenes se muestra los valores en las unidades físicas para las partículas inscritas en una esfera de radio 4.5 pc dentro de la cual se localizan completamente las regiones de interés. En las imágenes se muestra la posición donde se ubicó el frente de ionización (en amarillo) siguiendo el método mencionado, en cada una de las

etapas evolutivas.

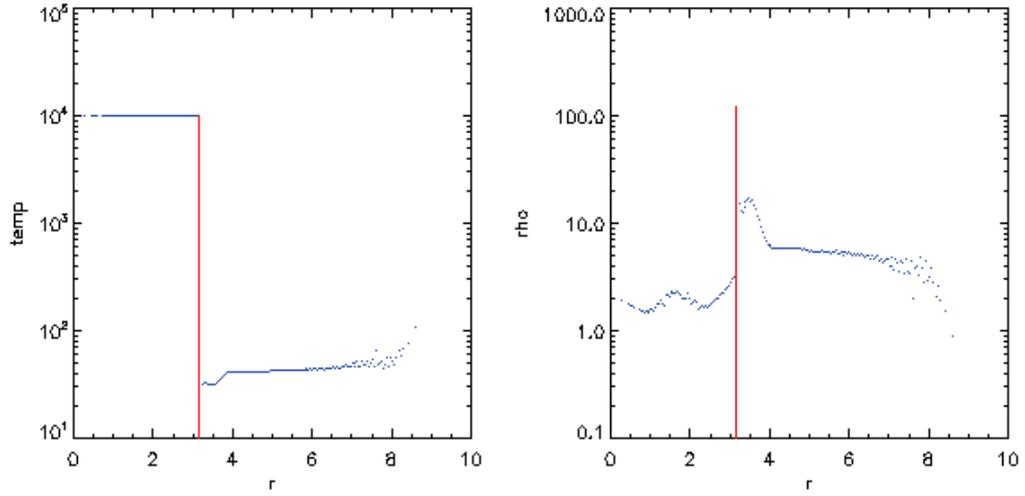


(a)

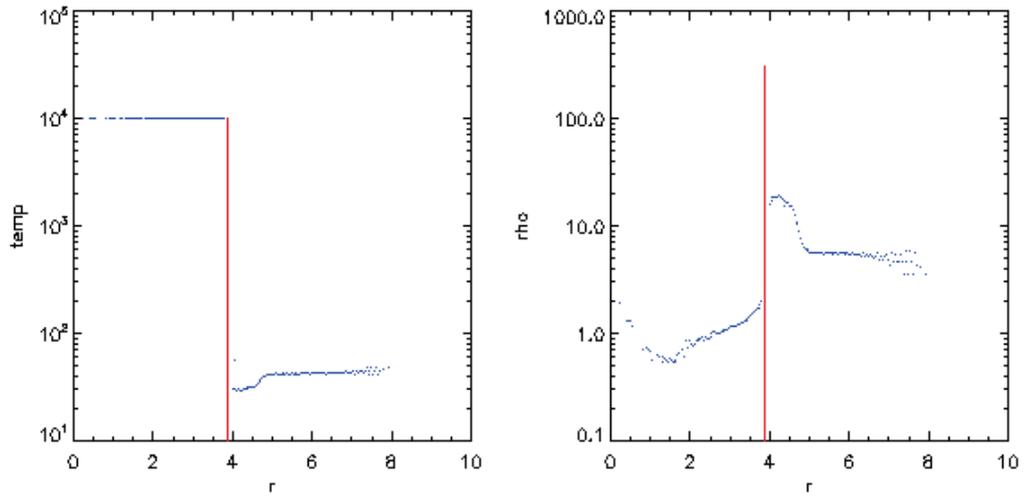


(b)

Fig. 6.4: Perfiles de temperatura y densidad de una región HII. a) Condiciones iniciales. b) Formación de la onda de choque. c) y d) Evolución intermedia y final del avance en una caja de  $L = 10$  pc.



(c)



(d)

Fig. 6.4: Perfiles de temperatura y densidad de una región III (cont.)

Como es claro, la primera imagen corresponde con las condiciones iniciales del gas a una temperatura de equilibrio y densidad correspondiente a las nubes moleculares dentro del MI.

Al observar el perfil de temperatura durante toda la evolución en la caja computacional, podemos ver una clara expansión de la región HII. Si consideramos los valores de distancia y tiempo entre la cuarta y la segunda imagen vemos que la velocidad promedio es  $v = (3.9 - 2.3)/(0.13 - 0.3)t \sim 3.2 \times 10^5 \text{ cm s}^{-1}$ , donde  $t = 1.54 \times 10^{14} \text{ s} \sim 5 \text{ Myr}$  es el valor de las unidades de tiempo en la simulación. Ahora, considerando que en el MI la velocidad del sonido es  $c_s \sim 2 \times 10^4 \text{ cm s}^{-1}$  y calculando el número de Mach obtenemos  $M \sim 16$ , lo cual nos indica que la expansión es altamente supersónica.

Por otra parte, al observar el perfil de temperatura se aprecia que el método para ubicar el frente de ionización resulta razonablemente aceptable para la simulación en baja resolución excepto en la última etapa de la evolución dentro de la caja, pues, al comparar con la fig. (6.3), observamos que en estas etapas, las barras de error crecen considerablemente respecto de las etapas iniciales. También pueden observarse que dentro de la ionizada la temperatura del gas es de  $10^4 \text{ K}$ , con algunas variaciones de  $\sim 50 \text{ K}$  por encima y hasta  $\sim 250 \text{ K}$  por debajo, lo cual coincide con el rango de temperaturas observado en las regiones HII. Nótese además que fuera de la región HII, el gas se mantiene neutro, a la temperatura de equilibrio, excepto en regiones muy cercanas al frente de ionización donde presenta variaciones de unas cuantas decenas de grados.

El perfil de densidad muestra rasgos más interesantes de la evolución de la región ionizada. Primero, se observa una capa comprimida de gas isotérmico por delante del frente de ionización, es decir, estas dos estructuras están inicialmente acopladas y conforme se desarrolla la simulación se desacoplan separándose cada vez más. También se observa que, conforme la capa chocada se desplaza hacia afuera, va empujando el gas neutro delante de ella, aglomerándolo y subiendo considerablemente su densidad, mientras se ensancha cada vez más. Nótese también, que la capa chocada posee no sólo un choque externo sino también uno interno en el cual la densidad baja más abruptamente que en el choque externo.

El segundo rasgo importante es que detrás del frente de ionización, existe una onda expansiva de gas ionizado que se propaga desde afuera de la esfera de Strömgren hacia adentro (obsérvese la parte con forma de cuchara), bajando la densidad en la región por la que va pasando. Obsérvese también que a distancias cercanas a la estrella, y por delante del frente de choque, la

densidad se mantiene en su valor inicial.

# Capítulo 7

## Conclusiones

### 7.1. Resumen

Durante el desarrollo del presente trabajo, se ha llevado a cabo el estudio del efecto de la retroalimentación por fotoionización a través de la implementación de un algoritmo basado en el método de radio de Strömgren de la estrella. Este método emplea la densidad columnar a lo largo de la línea de visión entre una estrella y la partícula de gas para determinar si el flujo de fotones es suficiente para ionizar el gas.

Para ello, fue necesario estudiar la física de los procesos que tienen lugar en el MI. En particular el calentamiento y enfriamiento del gas, la inestabilidad térmica, la turbulencia compresible, así como algunos de los efectos del campo magnético. También se dio una breve descripción de las componentes del MI y los valores característicos de densidad y temperatura. Así mismo, se revisaron las nociones de la estructura, evolución y destrucción de las NMs desde el punto de vista de la formación estelar, para lo cual fue necesario profundizar en algunos conceptos y fenómenos como la inestabilidad gravitacional, fragmentación de Hoyle, la tasa y la eficiencia de formación estelar, la función de masa inicial, así como los principales mecanismos de regulación de la formación estelar y el papel que desempeña la retroalimentación estelar.

Fue necesario también el estudio y la teoría analítica de las fases de for-

mación y expansión de las regiones HII, en particular, del frente de ionización y onda de choque que produce en el MI.

La implementación del algoritmo se llevó a cabo en el código SPH GADGET2, empleado en simulaciones de astrofísica, con algunas adecuaciones para el MI. Una parte muy importante para el desarrollo del presente trabajo fue entender el funcionamiento del método SPH, desde el punto de vista de una aproximación a la Lagrangiana a la dinámica de fluidos. Ello implicó el conocimiento de algunos conceptos básicos métodos numéricos como la condición de Courant y el método para calcular el avance temporal y la predicción de densidad, velocidad y aceleración de las partículas en la parte hidrodinámica del código.

Durante la etapa de implementación fue necesario extraer, depurar, analizar y visualizar los datos arrojados por la simulación, por lo cual se requirió implementar aplicaciones adicionales usando el software IDL. En particular, para los perfiles de densidad y temperatura, así como verificar la ionización de las partículas SPH.

Finalmente, se llevó a cabo la prueba del algoritmo verificando la evolución temporal del frente de ionización comparándolo con la solución analítica y, posteriormente se interpretaron cualitativa y físicamente los datos arrojados por la simulación.

## 7.2. **Discusión**

Sin duda, este trabajo representó la adquisición de un amplio conocimiento en las técnicas de programación del lenguaje C, así como de IDL. Sin embargo, la parte medular fue el conocimiento de la física involucrada en el modelado de las regiones HII.

### 7.2.1. **Limitaciones**

Durante el desarrollo del trabajo, se presentaron algunas limitaciones como la pobre documentación del código GADGET2 en su funcionamiento in-

terno de la implementación SPH, la falta de herramientas para visualizar a las partículas en 3D, aunada a la identificación de las mismas y la cantidad masiva de partículas aún a baja resolución. Otra de las limitaciones fue el esfuerzo necesario para adaptarse a la gran cantidad de conceptos empleados en astrofísica y la descripción de los procesos implicados, ya que la mayoría están mutuamente relacionados con otros procesos y fue necesaria la simplificación de la teoría en muchos aspectos. Sin embargo, estas simplificaciones son justamente las necesarias para simplificar la implementación del algoritmo y modelar dichos procesos de forma aceptable.

En el algoritmo, las estrellas no son resueltas individualmente dentro de las partículas sumidero (y consecuentemente, la evolución de regiones HII individuales). De igual forma, las regiones HII traslapadas requieren del cálculo dinámico de los coeficientes de ionización y de recombinación, así como de la implementación de un algoritmo bastante más complejo.

Las limitaciones en la capacidad de cómputo y las simplificaciones del modelo, como la consideración de gas puramente molecular sin considerar la componente atómica, especies químicas, el polvo interestelar, así como del campo magnético y la autogravedad del gas, representan un gran reto que debe llevarse a cabo paso a paso e implican una gran cantidad de conocimientos y estudios adicionales. Sin embargo, el modelo empleado es una muy buena aproximación en primera instancia y permite analizar en forma relativamente fácil los aspectos físicos y dinámicos de la evolución de regiones HII dentro de nubes moleculares.

## 7.3. Trabajo a futuro

### 7.3.1. Adaptaciones necesarias

El algoritmo descrito por Dale *et al.* es relativamente simple. Sin embargo, la complejidad de los cálculos y la cantidad masiva de partículas requeridas para simular correctamente el modelo físico hacen necesaria la consideración de algunas optimizaciones. Desde el punto de vista computacional se pueden llevar a cabo algunas adaptaciones para el algoritmo con el fin de agilizar los

cálculos.

La primera de ellas es paralelizar el código, es decir, llevar a cabo los cálculos simultáneamente en varios procesadores intercomunicados a través de una interfaz de transferencia de mensajes (MPI por sus siglas en inglés) por medio de la cual los datos relativos a las partículas del fluido dentro del dominio computacional pueden ser compartidos entre los procesadores. Posteriormente, se debe adecuar el código para el uso de fronteras periódicas para permitir la simulación de fluidos hidrodinámicos físicamente más extensos con una relativamente baja complejidad en su implementación.

En todas las simulaciones de los trabajos aquí citados se exhibe un colapso global, con una contracción gravitacional a escala de la nube, donde la retroalimentación parece regular únicamente la conversión de gas denso a estrellas localmente, sugiriendo que debería considerarse un modelo analítico de la TFE incluyendo la acreción, con la correspondiente implementación.

El algoritmo implementado no considera la mezcla de gas ionizado y neutro que se da en situaciones dinámicas altamente no homogéneas, como los flujos de acreción de partículas de gas por la estrella y vientos estelares. Sin embargo, Dale *et al.* describen una simplificación de este modelo que resulta relativamente simple de implementar a futuro.

## 7.4. Aplicaciones

El presente trabajo resulta de gran utilidad en la aplicación del algoritmo en simulaciones de regiones más extensas como las nubes moleculares, desde el punto de vista dinámico evolutivo y no como una región estática.

La disminución de la densidad del gas en las regiones HII cambia en gran medida la tasa de formación estelar en vecindad de las estrellas ionizantes. La retroalimentación estelar puede dar lugar de choques turbulentos que afectarían la tasa de formación estelar en regiones relativamente lejanas a la estrella ionizante, o bien, podría originar el desgarramiento de la nube en regiones cercanas a la parte externa de la misma. Por ello, el estudio posterior de su efecto en las nubes moleculares resulta por demás interesante para el

entendimiento de la dinámica del MI.

## **7.5. Conclusiones finales**

Basados en esta serie de análisis e interpretaciones y la verificación de la evolución temporal del FI con respecto a la solución analítica, podemos concluir que aún para la simulación en baja resolución el algoritmo tiene un comportamiento aceptable, por lo cual se puede llevar a cabo su uso en la simulación de la formación de regiones HII para determinar en una forma más detallada su efecto en la FE dentro de las NMs.

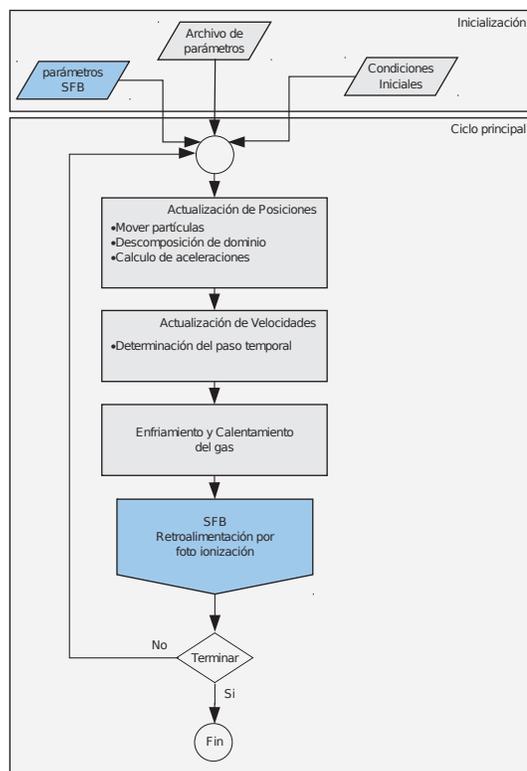




# Apéndice A

## Algoritmo

### A.1. Diagrama de Flujo GADGET2



Los bloques denotados “SFB” se refieren a la retroalimentación por los sinks; Sink FeedBack por sus siglas en inglés.

## A.2. Código generado

A continuación se muestra una lista de las adaptaciones implementadas en el código de GADGET2.

### A.2.1. sink\_feedback.h

```

1 #ifndef SINK_FEEDBACK
3 #define SFB_FALSE      0
  #define SFB_TRUE      1
5
  #ifndef PI
7 #define PI 3.141592654 // To avoid an error when Turbulence is
  not set
  #endif
9
  /* SINK_FEEDBACK needed constants */
11 #define He_PROTONMASS    6.6466e-24 // g
  #define H_PROTONMASS    1.6726e-24 // g
13 #define BOLTZMANN        1.3806e-16

15 FLOAT SFB_ECGS_CM3, SFB_MEFF, SFB_GM1_B; //Som conversion factors

17 FLOAT SFB_SUN_MASS, SFB_SUN_LIFETIME, SFB_STAR_MASS,
  SFB_STAR_LUMINOSITY;
  FLOAT SFB_STAR_PHOTON_FLUX, SFB_STAR_LIFETIME_ESTIMATE,
  SFB_SINK_n, SFB_EST_HSML;
19
  // Some definitions to indentify particle trace output
21 #define SFB_TR_SINK      1
  #define SFB_TR_SELECTED  2
23 #define SFB_TR_TARGET   3
  #define SFB_TR_RESULT_ON 4
25 #define SFB_TR_RESULT_OFF 40
  #define SFB_TR_MIN      -3
27 #define SFB_TR_TARGET_SEL -4

```

```

29 #define SFB_TR_SINK_INSIDE    -5
#define SFB_TR_NEIGHBOUR      -6
#define SFB_TR_ORIG_TARGET    -7
31 #define SFB_TR_NO_NEIGHBOURS -8

33 #define SFB_MAX_ELEMENTS      10000

35 FLOAT SFB_HALFBOXSIZE;
FLOAT SFB_ALPHA_B;           // cm3 sec-1 typical of an
                             HII galactic region. Constant in 2D/3D
37 int   SFB_MEASURES;         // Allow or avoid measures
                             calculation
    int   SFB_LOG_LEVEL;      //Log levels: 0 minimal, 1
                             informative, 2 resumed, 3 detailed, 4 extense
39 int   SFB_NUM_LOOP;
    int   SFB_MAX_LOOP;       //limit the number of loops
                             of the algorithm we execute to avoid hard disk problems
                             because excesive loggin
41
    FLOAT SFB_HSML_MAX_FACTOR; //max factor used to increment
                             hsml neighbour search
43 FLOAT SFB_LIGHTSPEED;
    FLOAT SFB_SUN_LUMINOSITY;
45 FLOAT SFB_SEC2YEAR;        // Seconds to years conversion
    FLOAT SFB_SEC2MYEAR;     // Seconds to mega years
                             conversion
47 FLOAT SFB_RS_pc;          // Initial stromgen radius (in
                             parsecs)
    FLOAT SFB_RS_pc2;
49
    //some factors to convert to/from physical units
51 FLOAT SFB_RHO_FACTOR;
    FLOAT SFB_4PI_ALPHA_FLXFACT;
53 FLOAT SFB_4PI_ALPHA_RHO_FACTOR;

55 int   SFB_CLASS_PROB_INITIALIZED;
    int   SFB_TERMINATED;
57 int   SFB_DEBUG_PARTICLE;
    int   SFB_CURRENT_PARTICLE;
59 FILE *SFB_LOG_ALLPOS;
    FILE *SFB_LOG_TRACE;
61
    //holds sinks index referent to P struct, used to sort sinks by
    distance2 or mass
63 struct sink_data_sort {

```

```

    int index;
65  FLOAT data;
    };
67
    struct tarray {
69     long int part[SFB_MAX_ELEMENTS];
        int count;
71 };

73 struct tarray selected_targets;
    struct tarray neighbours;
75 struct tarray tmp_list;

77 //time elapsed for execution variables
    struct timeval SFB_T1, SFB_T2;
79

81 //We need to make it public so we can execute from init.c
    void SFB_parameters();
83 void SFB_class_prob_init();
    FLOAT SFB_calculate_imf(FLOAT m0);
85 void SFB_do_sink_radiation();
    void SFB_initialize();
87 void SFB_terminate();
    #endif //SINK_FEEDBACK

```

### A.2.2. sink\_feedback.c

```

#if SINK_FEEDBACK
2
    #include <stdio.h>
4 #include <stdlib.h>
    #include <math.h>
6 #include <float.h>
    #include <time.h>
8 #include <sys/time.h>

10 #include "allvars.h"
    #include "proto.h"
12 #include "sink_feedback.h"

14
    void time_diff(struct timeval x, struct timeval y) {

```

```

16  int nsec;

18  if (x.tv_usec < y.tv_usec) {
20      nsec = (y.tv_usec - x.tv_usec) / 1000000 + 1;
21      y.tv_usec -= 1000000 * nsec;
22      y.tv_sec += nsec;
23  }
24  if (x.tv_usec - y.tv_usec > 1000000) {
25      nsec = (x.tv_usec - y.tv_usec) / 1000000;
26      y.tv_usec += 1000000 * nsec;
27      y.tv_sec -= nsec;
28  }

29  printf("Elapsed time = %d seconds, %d microseconds\n", abs(x.
30      tv_sec - y.tv_sec), abs(x.tv_usec - y.tv_usec));
31  }

32  void open_log_file(FILE **file , const char *filename , const char
33      *mode) {
34      *file=fopen(filename ,mode);
35      if ( *file == NULL )
36          printf("Error: can't open file %s\n",filename);
37  }

38  void close_log_file(FILE **file) {
39      if (*file != NULL) {
40          if (fclose(*file) == 0)
41              *file= NULL;
42          else
43              printf("Error closing file descriptor!!!");
44      }
45  }

46  int logthis(int min_level) {
47      if ( ((SFB_CURRENT_PARTICLE == SFB_DEBUG_PARTICLE) && (
48          SFB_DEBUG_PARTICLE>=0) && (SFB_CURRENT_PARTICLE>=0)) || //
49          Debug this particle
50          (SFB_LOG_LEVEL >= min_level) )
51          return 1;
52      return 0;
53  }

54  void log_part(int target , int i , int code) {
55      if ((P[i].Type == 0) && (P[i].ID >= 0)) //gas particle

```

```

    printf("% 7d,%d,% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6
        e,% 6.6e\n", target , code ,i , P[i].Pos[0] , P[i].Pos[1] , P
        [i].Pos[2] , SphP[i].Hsml , SphP[i].Entropy , P[i].SFB_dProj
        , P[i].SFB_dLOS);
58  else
    printf("% 7d,%d,% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6
        e,% 6.6e\n", target , code ,i , P[i].Pos[0] , P[i].Pos[1] , P
        [i].Pos[2] , 0.0 , 0.0 , P[i].SFB_dProj , P[i].SFB_dLOS );
60 }
62
63 void log_positions() {
64  int i;
65
66  if (logthis(2)) printf("Loggin all particle (%d) positions\n",
        NumPart);
    open_log_file(&SFB_LOG_ALLPOS,"log_allpositions" , "w");
68  printf("MinEgySpec= % 6.6e\n" , All.MinEgySpec);
    for (i=0; i<NumPart; i++) {
70  if ((P[i].Type == 0) && (P[i].ID >= 0))
        fprintf(SFB_LOG_ALLPOS, "%d,% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e
            ,% 6.6e,% 6.6e,% 6.6e,% 6.6e\n", SFB_TR_TARGET, P[i].ID
            , P[i].Pos[0] , P[i].Pos[1] , P[i].Pos[2] , SphP[i].Hsml ,
            SphP[i].Density , SphP[i].Entropy , P[i].SFB_dSink , P[i].
            SFB_dLOS);
72  else if ((P[i].Type == 5) && (P[i].ID < 0)) //Sinks only
        fprintf(SFB_LOG_ALLPOS, "%d,% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e
            ,% 6.6e,% 6.6e,% 6.6e,% 6.6e,\n", SFB_TR_SINK, P[i].ID,
            P[i].Pos[0] , P[i].Pos[1] , P[i].Pos[2] , 0.0 , 0.0 , 0.0 ,
            0.0 , 0.0);
74  }
    close_log_file(&SFB_LOG_ALLPOS);
76 }
78
79 //Nu : Star flux ( number of uv photons sec^-1)
80 FLOAT stromgren_radius(FLOAT N_uv, FLOAT ne) { // cm
    return pow( (3.0 * N_uv) / (4.0 * PI * SFB_ALPHA_B * ne * ne) ,
        1.0/3.0); // cm
82 }
83
84 void typicals_stromgren_radius() {
    //Matlab test code: Nu=51.0e48, ne=1, alfa=3.0e-13, pc=3.08568
    e18, Rs=( (3 * Nu) / (4 * pi * alfa * ne * ne) ) ^ ( 1/3),
    Rspc=Rs/pc

```

```

86 //so we can get Rspc in function of (n_e n_h)^1/3., with ne=1 .
    See Spitzer tables
    printf("O5 type Stromgren Radius= %6.6e pc\n",
        stromgren_radius(51.000e48, SFB_SINK_n) / All.
        UnitLength_in_cm);
88 printf("O8 type Stromgren Radius= %6.6e pc\n",
        stromgren_radius( 3.900e48, SFB_SINK_n) / All.
        UnitLength_in_cm);
    printf("B1 type Stromgren Radius= %6.6e pc\n",
        stromgren_radius( 0.003e48, SFB_SINK_n) / All.
        UnitLength_in_cm);
90 }

92 void SFB_class_prob_init() {
    int i;
94
    if (SFB_CLASS_PROB_INITIALIZED == 0) {
96     SFB_CLASS_PROB_INITIALIZED= 1;
        printf("\nInitializing classical problem conditions for sink
            feedback\n");
98     for (i=0; i< NumPart; i++) {
        if ((P[i].Type == 0) && (P[i].ID >= 0)) { //gas
100         P[i].SFB_heating_ID2= -All.MaxPart;
        }
102     else if ( (P[i].Type == 5) && (P[i].ID2 < 0) ) { //sinks
        //This sink is created initially radiating
104         P[i].SFB_alive= 1; //sink is capable of feedback the gas
            particles
            P[i].SFB_time_on= 0.0;
106         P[i].SFB_radiating= 0;
            P[i].SFB_radiating_elapsed= 0.0;
108         P[i].SFB_num_ionizing_stars= 0;
            P[i].SFB_photon_flux= 0.0;
110         //Reset gas data to make it clear
            P[i].SFB_heating_ID2= -All.MaxPart;
112         P[i].SFB_dSink= 0.0;
            P[i].SFB_dLOS= 0.0;
114     }
        } // for i
116 }
}

118

120 void SFB_parameters() {
    int stars_type;

```

```

122  FLOAT EnergyConverted, MassFractionLost;

124  SFB_LOG_ALLPOS= NULL; //don't modify manually
    SFB_LOG_TRACE= NULL; //don't modify manually
126
    SFB_TERMINATED= 0; //don't modify manually
128  SFB_NUM_LOOP= 0; //don't modify manually
    SFB_CLASS_PROB_INITIALIZED= 0; //Don't modify manually
130
132  SFB_HALFBOXSIZE= 0.5* All.BoxSize;

134  SFB_ALPHA_B= 3.0e-13; // cm^3 sec^-1 typical of an HII galactic
    region. (Constant)
    SFB_ECGS_CM3= All.UnitEnergy_in_cgs / pow(All.UnitLength_in_cm,
    3.0);
136  SFB_MEFF= All.MWeight * H_PROTONMASS;
    SFB_GMI_B= GAMMA_MINUS1 / BOLTZMANN;
138  SFB_RHO_FACTOR= ( All.UnitDensity_in_cgs / (All.MWeight *
    H_PROTONMASS) );
    if (logthis(2)) printf("SFB_RHO_FACTOR= % 6.6e\n",
    SFB_RHO_FACTOR);
140  SFB_4PI_ALPHA_FLXFACT= 4.0 * PI * SFB_ALPHA_B * pow(All.
    UnitLength_in_cm, 3.0);
    SFB_4PI_ALPHA_RHO_FACTOR= (SFB_RHO_FACTOR * SFB_RHO_FACTOR) *
    SFB_4PI_ALPHA_FLXFACT;
142  //SFB_EST_HSML= ( sqrt(3.0 * All.BoxSize * All.BoxSize) / pow(
    All.MaxPart, 1.0/3.0) ) * pow(All.DesNumNgb, 1.0/3.0); //
    estimate hsml using an aproximation of NPerDir particles.
    Note factor 2.0, necessary to catch at least one particle
    SFB_EST_HSML= ( sqrt(3.0 * All.BoxSize * All.BoxSize) / pow(All
    .MaxPart, 1.0/3.0) ); //estimate hsml using an aproximation
    of NPerDir particles. Note factor 2.0, necessary to catch
    at least one particle
144
    printf("GAMMA_MINUS1= % 6.6e, MWeight= % 6.6e, H_PROTONMASS= %
    6.6e, BOLTZMANN= % 6.6e\n", GAMMA_MINUS1, All.MWeight,
    H_PROTONMASS, BOLTZMANN);
146
148  SFB_DEBUG_PARTICLE= -1;
    SFB_CURRENT_PARTICLE= -1; // must take care of it when
    debugging a prarticle. Please set to -1 (when not used) in
    necessary places

```

```

150 SFB_LOG_LEVEL= 1; //Log levels: 0 minimal, 1 informative, 2
    resumed, 3 detailed, 4 extense
151 SFB_MEASURES= 0; //1 Allow measures calculation, 0 avoid this
152
153 SFB_HSML_MAX_FACTOR= 3.0; //Allow to increase hsml radius for
    neighbours search if none is found
154
155 SFB_MAX_LOOP= 200; // 0 -> don't run this algorithm , >=1000
    don't consider a loop limit
    printf("Max loop allowed is %d\n", SFB_MAX_LOOP);
156
157 if (SFB_MAX_LOOP < 1000)
    printf("Max loop allowed is %d\n", SFB_MAX_LOOP);
158
159 //SFB_MAX_HEATED_IN_LOOP=100000;
160
161 SFB_SINK_n= 100; //Numerical density at sink position to
    calculate Photon Flux. 10 part cm-3 ( HI Region typical
    density ). ne=np number of electrons
162
163
164 if (logthis(2)) printf("Calculating SINK_FEEDBACK parameters\n"
    );
165
166 //We must calculate Sun Lifetime here
167
168 SFB_SEC2YEAR = (float) (365*24*60*60);
169 SFB_SEC2MYEAR = SFB_SEC2YEAR * 10.0e6;
170
171
172 SFB_SUN_MASS = 1.9891e33; // g
173 SFB_LIGHTSPEED = 2.99792458e10; // cm/s
174 SFB_SUN_LUMINOSITY = 3.839e33; // erg/s
175
176 //When 4 Hydrogen are fusioned into Helium there is a missing
    mass , which was trasformed into Energy
177
178 MassFractionLost= (4.0*H_PROTONMASS - He_PROTONMASS) / (4.0*
    H_PROTONMASS);
    EnergyConverted= MassFractionLost * SFB_SUN_MASS *
    SFB_LIGHTSPEED*SFB_LIGHTSPEED;
179
180 printf("MassFractionLost in H->He fusion: % 6.6e, Sun's energy
    available: % 6.6e\n", MassFractionLost , EnergyConverted);
    SFB_SUN_LIFETIME= ( MassFractionLost * SFB_SUN_MASS *
    SFB_LIGHTSPEED*SFB_LIGHTSPEED) / SFB_SUN_LUMINOSITY; // t=
    E/L ( seconds )
181
182

```

```

184 /*
CARROL, APENDIX G. STELLAR DATA
186 SPECTRAL TYPE      TEMP EFF (K)      L/L0      R/R0      M/M0
-----
188 O5                  42000.0          499000.0  13.4      60
O8                  35800.0          147000.0  10.0      23
B1                  2540.0           9950.0    5.2       --
    -----> ( 18 is the mean B type star mass)
190 SUN (G2)          5777.0           1.0       1.0       1.0
*/
192
194 /*
SPITZER, CHAPTER IONIZATION AND DISSOCIATION, TABLE 5.3
196 SPECTRAL TYPE      TEMP. EFF      N_u x10^-48      R_s (n_e n_h)
    ^-1/3
    K              s^-1              pc cm^-2
198 -----
200 O5                  47000.0          51.0          110.0
O8                  36500.0          3.9           47.0
B1                  22600.0          0.0033        4.4
202 */
204 stars_type=2;
    if (stars_type==3) { //Considering O5 stars:
206     SFB_STAR_MASS= 60.0; // data in M/M0 column
     SFB_STAR_LUMINOSITY= 499000.0; //data in L/L0 column
208     SFB_STAR_PHOTON_FLUX= 51.0e48;
    } else if (stars_type==2) { //Considering O8 stars:
210     SFB_STAR_MASS= 23.0; // data in M/M0 column
     SFB_STAR_LUMINOSITY= 147000.0; //data in L/L0 column
212     SFB_STAR_PHOTON_FLUX= 3.9e48;
    } else if (stars_type==1) { //Considering B1 stars:
214     SFB_STAR_MASS= 18.0; // data in M/M0 column
     SFB_STAR_LUMINOSITY= 9950.0; //data in L/L0 column
216     SFB_STAR_PHOTON_FLUX= 0.0033e48;
    }
218
    //We use mass/luminosity ratio to estimate the star lifetime
220 SFB_STAR_LIFETIME_ESTIMATE= SFB_SUN_LIFETIME / pow(
    SFB_STAR_MASS, 2.5) ; //t = tSun * (M/M0)^-2.5 = (E0 / L0)
    *(M/M0)^-2.5

```

```

222 SFB_RS_pc= stromgren_radius(SFB_STAR_PHOTON_FLUX, SFB_SINK_n) /
      All.UnitLength_in_cm;
SFB_RS_pc2= SFB_RS_pc * SFB_RS_pc;
224
      if (logthis(1)) printf("SUN_LIFETIME= % 6.6e (s) = % 6.6e M(yrs)
        )\n", SFB_SUN_LIFETIME, SFB_SUN_LIFETIME / SFB_SEC2MYEAR );
226      if (logthis(1)) printf("SFB_STAR_MASS= % 6.6e (MSun),
        SFB_STAR_LUMINOSITY= % 6.6e (LSun), SFB_STAR_PHOTON_FLUX= %
          6.6e (photons / m^2.s),\nSFB_STAR_LIFETIME_ESTIMATE= % 6.6
          e (s) = %6.6e (Myears), Stromgren Radius= % 6.6e (pc)\n",
        SFB_STAR_MASS, SFB_STAR_LUMINOSITY, SFB_STAR_PHOTON_FLUX,
        SFB_STAR_LIFETIME_ESTIMATE, SFB_STAR_LIFETIME_ESTIMATE /
        SFB_SEC2MYEAR , SFB_RS_pc);
      if (logthis(1)) printf("UnitTime_in_s= % 6.6e (s)= % 6.6e (Myrs)
        )\n", All.UnitTime_in_s, All.UnitTime_in_Megayears);
228 typicals_stromgren_radius();
      log_positions();
230 open_log_file(&SFB_LOG_TRACE,"trace", "a+");
    }
232
234 int between(FLOAT value, FLOAT a, FLOAT b) {
      if ((value >= a) && (value<b))
236         return 1;
      else
238         return 0;
    }
240
242 FLOAT evaluate_imf(FLOAT f, FLOAT alfa, FLOAT a, FLOAT b) {
      return f*( pow(b, alfa + 2.0) - pow(a, alfa + 2.0) ) / ( alfa +
        2.0);
    }
244
    //calculate the mass fraction in stars with m>m0 Msun
246 FLOAT SFB_calculate_imf(FLOAT m0) {
      FLOAT A1,A2,A3,area1 ,area2 ,area3 ,total_area;
248     FLOAT i0 ,i1 ,i2 ,i3 ,fracc_mass;

250     const FLOAT m1=0.08;
      const FLOAT m2=0.5;
252     const FLOAT a1=-0.3;
      const FLOAT a2=-1.3;
254     const FLOAT a3=-2.3;
      const FLOAT mmin=0.01;
256     const FLOAT mmax=1e20; //just a high mass value

```

```

258  if (logthis(2)) printf("Starting IFM Calculus for M0=% 6.6 e
      MSun\n-----\n",m0);
      //segments adjustment. A1,A2 and A3 are scaling factors, a1,a2,
      a3 are the segments slope in logarithmic plot
260  A3= 1.0;
      A2= pow(m2, a3-a2);
262  A1= A2 * pow(m1, a2-a1);
      if (logthis(4)) printf("Segments adjustment A1= % 6.6 e, A2=
      %6.6e, A3= %6.6e \n",A1,A2,A3);
264
      //normalization
266  areal= -A1 * (pow(mmin, a1 + 1.0) - pow(m1, a1 + 1.0)) / (a1 +
      1.0);
      area2= -A2 * (pow(m1, a2 + 1.0) - pow(m2, a2 + 1.0)) / (a2 +
      1.0);
268  area3= -A3 * (pow(m2, a3 + 1.0) - pow(mmax, a3 + 1.0)) / (a3 +
      1.0);
      total_area= areal + area2 + area3;
270  if (logthis(4)) printf("First Normalization a1= %6.6e, a2= %6.6
      e, a3= %6.6e, tot_a= %6.6e \n", areal, area2, area3,
      total_area);

272  A1= A1 / total_area;
      A2= A2 / total_area;
274  A3= A3 / total_area;
      if (logthis(4)) printf("Rescaling Factors A1= %6.6e, A2= %6.6e,
      A3= %6.6e \n",A1,A2,A3);
276
      //Verification of Normalization
278  areal= -A1 * (pow(mmin, a1 + 1.0) - pow(m1, a1 + 1.0)) / (a1 +
      1.0);
      area2= -A2 * (pow(m1, a2 + 1.0) - pow(m2, a2 + 1.0)) / (a2 +
      1.0);
280  area3= -A3 * (pow(m2, a3 + 1.0) - pow(mmax, a3 + 1.0)) / (a3 +
      1.0);
      total_area = areal + area2 + area3;
282  if ( between(total_area, 1.0 - 0.001, 1.0 + 0.001) == 0 ) {
      printf("Second Normalization a1= %6.6e, a2= %6.6e, a3= %6.6e,
      tot_a= %6.6e \n", areal, area2, area3, total_area);
284  printf("IMF Normalization failed.!!!\n");
      SFB_terminate();
286  exit(1);
      } else
288  printf("IMF Normalization ok.\n");

```

```

    if (logthis(3)) printf("Second Normalization a1= %6.6e, a2=
        %6.6e, a3= %6.6e, tot_a= %6.6e \n", area1, area2, area3,
        total_area);
290
    //Calculus of the mass in stars which m>M0 Msun
292 i1= evaluate_imf(A1,a1,mmin,m1);
    i2= evaluate_imf(A2,a2,m1,m2);
294 i3= evaluate_imf(A3,a3,m2,mmax);

296 //identification of initia interval
    if ((m0>=mmin) && (m0<=m1))
298     i0= evaluate_imf(A1,a1,m0,m1) + i2 + i3;

300     if ((m0>m1) && (m0<=m2))
        i0= evaluate_imf(A2,a2,m0,m2) + i3;
302
    if (m0>m2)
304     i0= evaluate_imf(A3,a3,m0,mmax);

306 fracc_mass= i0 / (i1 + i2 + i3);
    if (logthis(4)) printf("i0= %6.6e, i1= %6.6e, i2= %6.6e, i4=
        %6.6e\n",i0,i1,i2,i3);
308     if (logthis(1)) printf("Fraction of Mass %6.6e, in stars of
        %6.6e Msun\n",fracc_mass,m0);
    if (logthis(2)) printf("
        _____\nIFM calculus
        ending\n");
310
    return fracc_mass;
312 } //calculate_imf

314 /* This function is a comparison kernel for sorting the sink-
        particle distances ( from nearest to farest ). */
signed int compare_sink_distances(const void *a, const void *b) {
316     if (((struct sink_data_sort *) a)->data < (((struct
        sink_data_sort *) b)->data))
        return -1;

318
    if (((struct sink_data_sort *) a)->data > (((struct
        sink_data_sort *) b)->data))
320     return +1;

322     return 0;
    }
324

```

```

/* This function is a comparison kernel for sorting the sink-
   particles mass ( from biggest to lowest ). */
326 signed int compare_sinks_mass(const void *a, const void *b) {
   if (((struct sink_data_sort *) a)->data > (((struct
       sink_data_sort *) b)->data))
328     return -1;

330   if (((struct sink_data_sort *) a)->data < (((struct
       sink_data_sort *) b)->data))
       return +1;
332
   return 0;
334 }

336 FLOAT distance2(int a, int b) {
   FLOAT dx, dy, dz;
338
   dx= P[b].Pos[0] - P[a].Pos[0];
340   dy= P[b].Pos[1] - P[a].Pos[1];
   dz= P[b].Pos[2] - P[a].Pos[2];
342   return (dx * dx) + (dy * dy) + (dz * dz);
   }
344

void get_vector(int a, int b, FLOAT v[]) {
346   int j;

348   for (j=0; j<3; j++)
       v[j]= P[b].Pos[j] - P[a].Pos[j];
350 }

352 //Returns vector (not unitary) length, if d==0, u is not unitary
FLOAT get_u_vector(int a, int b, FLOAT u[]) {
354   int j;
   FLOAT d;
356

   d= 0.0;
358   for (j=0; j<3; j++) {
       u[j]= P[b].Pos[j] - P[a].Pos[j];
360       d+= u[j] * u[j];
   }
362   d= sqrt(d);
   if (d != 0.0) //avoid zero division
364       for (j=0; j<3; j++)
           u[j]= u[j] / d;
366   return d;

```

```

}
368
369 FLOAT dot_prod( FLOAT v_a[], FLOAT v_b[]) {
370     FLOAT dp;
371     int j;
372
373     dp= 0.0;
374     for (j=0; j<3; j++)
375         dp+= (v_a[j] * v_b[j]);
376     return dp;
377 }
378
379 void cross_prod( FLOAT a[], FLOAT b[], FLOAT x[]) {
380     x[0]= a[1]*b[2] - a[2]*b[1];
381     x[1]= a[2]*b[0] - a[0]*b[2];
382     x[2]= a[0]*b[1] - a[1]*b[0];
383 }
384
385 void log_trace(int target, int i, int code) {
386     if (logthis(3))
387         if ((P[i].Type == 0) && (P[i].ID >= 0)) //gas particle
388             fprintf(SFB_LOG_TRACE, "% 3d,% 7d,%d,% 7d,% 6.6e,% 6.6e,%
389                 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e\n", SFB_NUM_LOOP,
390                 target, code, i, P[i].Pos[0], P[i].Pos[1], P[i].Pos[2],
391                 SphP[i].Hsml, SphP[i].Entropy, P[i].SFB_dProj, P[i].
392                 SFB_dLOS);
393         else
394             fprintf(SFB_LOG_TRACE, "% 3d,% 7d,%d,% 7d,% 6.6e,% 6.6e,%
395                 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e\n", SFB_NUM_LOOP,
396                 target, code, i, P[i].Pos[0], P[i].Pos[1], P[i].Pos[2],
397                 0.0, 0.0, P[i].SFB_dProj, P[i].SFB_dLOS );
398 }
399
400 int update_temp(int this_part, int modify) {
401     FLOAT entropy_new, temp, temp_new, energy, energy_cgs, rho_cgs,
402         yntot, rho_gamma;
403
404     rho_gamma= pow(SphP[this_part].Density, GAMMA);
405     energy= SphP[this_part].Entropy * rho_gamma / GAMMA_MINUS1;
406     if (energy < SphP[this_part].Density * All.MinEgySpec)
407         energy= SphP[this_part].Density * All.MinEgySpec;
408
409     rho_cgs = SphP[this_part].Density * All.UnitDensity_in_cgs;
410     energy_cgs = energy * SFB_ECGS_CM3;

```

```

404 | yntot = rho_cgs / SFB_MEFF;
406 | temp = energy_cgs * SFB_GM1_B/ yntot;

408 | if ( temp < 10000.0 ) {
      |   if (modify == SFB_FALSE) //just testing?
410 |     return -1; //check if particle must be heated
      |     temp_new = 10000.0;
412 |     energy_cgs = temp_new * yntot / SFB_GM1_B;
      |     energy = energy_cgs / SFB_ECGS_CM3;
414 |     entropy_new = GAMMA_MINUS1 * energy / rho_gamma;
      |     if (logthis(4)) printf("Temp. Part. %12d : Entropy= %6.6e ,
      |       NewEntropy= %6.6e, Temp= %6.6e, TempNew= %6.6e,
      |       Density= %6.6e, DeltaS= %6.6e\n", P[this_part].ID, SphP[
      |       this_part].Entropy, entropy_new, temp, temp_new, SphP[
      |       this_part].Density, entropy_new - SphP[this_part].Entropy
      |     );
416 |     SphP[this_part].Entropy= entropy_new;
      |     return 1;
418 |   }

420 | return 0; //no update is default return value
      | }
422 |

424 | void show_selected_particles(int this_sink, int orig_target) {
      |   int i;
426 |
      |   if (logthis(3)) printf("Showing selected targets list (
      |     including target particle and sink)\n");
428 |   //if (logthis(3)) log_part(orig_target, i, SFB_TR_TARGET);
      |   if (selected_targets.count == 0) {
430 |     if (logthis(3)) printf(" No targets found\n");
      |     return;
432 |   }

434 |   log_trace(orig_target, this_sink, SFB_TR_SINK);
      |   if (logthis(3)) log_part(orig_target, this_sink, SFB_TR_SINK);
436 |   for (i=0; i<selected_targets.count; i++) {
      |     log_trace(orig_target, selected_targets.part[i],
      |       SFB_TR_SELECTED);
438 |     if (logthis(3)) log_part(orig_target, selected_targets.part[i]
      |       ], SFB_TR_SELECTED);
      |   }
440 | }

```

```

    if (logthis(3)) printf("End of selected targets\n\n");
442 }
444
void copyPos(int this_part, FLOAT pos[]) {
446     int i;
448     for (i=0; i<3; i++)
        pos[i]= P[this_part].Pos[i];
450 }
452 FLOAT distanceTo(int this_part, FLOAT pos[]) {
    int i;
454     FLOAT d;
456     d= 0.0;
    for (i=0; i<3; i++)
458         d+= (P[this_part].Pos[i] - pos[i]) * (P[this_part].Pos[i] -
            pos[i]);
    return sqrt(d);
460 }
462 FLOAT distanceToPos(FLOAT p[], FLOAT q[]) {
    int i;
464     FLOAT d;
466     d= 0.0;
    for (i=0; i<3; i++)
468         d+= (p[i] - q[i]) * (p[i] - q[i]);
    return sqrt(d);
470 }
472 void advance_position(int this_part, FLOAT advance, FLOAT u[],
    FLOAT pos[]) {
    int i;
474
    for (i=0; i<3; i++)
476         pos[i]= P[this_part].Pos[i] + (advance * u[i]);
    }
478
void array_copy(struct tarray *a, struct tarray *b) {
480 int i;
    b->count= a->count;
482     for (i= 0; i< a->count; i++)
        b->part[i]= a->part[i];

```

```

484 }
486 void array_push(struct tarray *a, int this_part) {
488     if (a->count < SFB_MAX_ELEMENTS) {
488         a->part[ a->count ]= this_part;
488         a->count++;
490     }
492 }
492 int in_array(struct tarray *a, int this_part) {
494 int i;
494     for (i= (a->count)-1; i>=0 ; i--)
496         if (a->part[i] == this_part)
498             return 1;
498     return 0;
500 }
502 int array_push_once(struct tarray *a, int this_part) {
504     if (in_array(a, this_part) == 1)
506         return 1;
506     array_push(a, this_part);
508     return 0;
508 }
508 int array_pop_min_dProj(struct tarray *a) {
510 int i, min_part;
510 float min_dProj;
512     if (a->count == 0)
514         return -1;
514     min_dProj= All.BoxSize; //just a high value
514     for (i= 0; i < a->count; i++)
516         if ((a->part[i] != -1) && (P[a->part[i]].SFB_dProj <
518             min_dProj)) {
518             min_dProj= P[a->part[i]].SFB_dProj;
518             min_part= a->part[i]; //necessary if two particles have the
520                 same distance projected
520         }
520     for (i= 0; i < a->count; i++)
522         if (a->part[i] == min_part)
524             a->part[i]= -1;
524     return min_part;

```

```

526 void sort_selected_particles(struct tarray *a, struct tarray *b)
    {
    528     int i, k, n;

        //sort selected particles by dProj
    530     if (logthis(3)) printf("Sorting particles\n");
        b->count= 0;
    532     n= a->count;
        for (i= 0; i<n; i++) { // selected particles are stored from
            far to near sink
    534         k= array_pop_min_dProj(a);
            array_push(b, k);
    536     }
    }

538 void filter_particles(struct tarray *a, struct tarray *b) {
    540     int i, j, k, n;
        //filter redundant particles taking in account the advance and
            using min dLOS
    542     if (logthis(3)) printf("Filtering particles\n");
        n= a->count;
    544     for (i= 0; i<(n-1); i++) { // particles are stored from far to
            near sink
        for (j= i+1; j<n; j++) {
    546         //if (logthis(3)) printf("i=% 7d, j=% 7d", i, j);
            if (P[a->part[j]].SFB_dProj > P[a->part[i]].SFB_dProj) {//
                already min dLOS
    548         //if (logthis(3)) printf("last\n");
            break;
    550     } else if (P[a->part[j]].SFB_dProj == P[a->part[i]].
            SFB_dProj) {
        if (P[a->part[i]].SFB_dLOS == 0.0) {
    552         //if (logthis(3)) printf("Zero\n");
            a->part[j]= -1;
    554     } else if (P[a->part[j]].SFB_dLOS >= P[a->part[i]].
            SFB_dLOS);
        //if (logthis(3)) printf("Not min\n ");
    556         a->part[j]= -1;
            } //same dLOS
    558     } // for j
        } // for i
    560     b->count= 0.0;
        for (i= 0; i<n; i++)
    562         if (a->part[i] != -1) { // selected particles are stored from
            far to near sink

```

```

564     k= a->part[i];
        array_push(b, k);
566 }

568 //ionization algorithm starting from orig_target particle
//Returns: 0 for not ionized particle
570 //      1 newly ionized
//      -1 already ionized
572 signed int particle_ionization(int this_sink, int orig_target) {
    FLOAT l, min_h, ppos[3], Sigma, N, hsml, hsml2, hsml_frac,
        hsml_factor, tmp, hwidth, min_l;
574 FLOAT I_r, lr_i, flux_arriving, lfa, Delta_ri, n_ri, r_i,
        TotI_r;
    FLOAT d_os, v_op[3], u_os[3];
576 int startnode; // (int) to avoid compilation warning in
        function ngb_treefind_variable
    int i, i_tmp, j, k, n, ngb, this_part, sel_part, lsel_part,
        next_part, neigh_part, updated, target_found, ki,
        sink_reached;

578 SFB_CURRENT_PARTICLE= orig_target;
580 if ((SFB_CURRENT_PARTICLE == 7411) || (SFB_CURRENT_PARTICLE ==
        13710) || (SFB_CURRENT_PARTICLE == 19057)) {
        printf("Particle %7d found\n", SFB_CURRENT_PARTICLE);
582 SFB_DEBUG_PARTICLE= SFB_CURRENT_PARTICLE;
    }
584 if (logthis(2)) printf("Determining if particle [%7d]->% 7d is
        heated by sink [%7d]->% 7d\n", orig_target, P[orig_target].
        ID, this_sink, P[this_sink].ID2);

586 d_os= get_u_vector( orig_target, this_sink, u_os );
    P[orig_target].SFB_dSink= d_os;
588 P[orig_target].SFB_dProj= d_os;
    P[orig_target].SFB_dLOS= 0.0;
590 if (logthis(3)) printf("Orig_target[%d] at [ %6.6e, %6.6e, %6.6
        e], DistToSink=% 6.6e\n", orig_target, P[orig_target].Pos
        [0], P[orig_target].Pos[1], P[orig_target].Pos[2], P[
        orig_target].SFB_dSink);

592 if (logthis(3)) log_part(orig_target, orig_target,
        SFB_TR_ORIG_TARGET); //original target
    selected_targets.count= 0;

594 l= d_os;

```

```

596 sink_reached= 0;
    lsel_part= -1;
598 neigh_part= orig_target; //particle where we start searching
    for neighbours
    hsml= SphP[orig_target].Hsml;
600 hsml2= hsml * hsml;
    array_push(&selected_targets , orig_target); //manually put
    orig_target into selected targets list
602 while ((l > 0.0) && (sink_reached == 0)) {

604     if (l <= hsml)
        if (distance2(this_sink , neigh_part) <= hsml)
606         sink_reached= 1;

608     //printf("Sel ");
    min_h= All.BoxSize;
610 next_part= neigh_part; //no selected particle
    startnode= All.MaxPart;
612 ngb= ngb_treefind_variable(P[neigh_part].Pos, hsml, &
        startnode); //Note: we only need to pass the memory
        address of the fist position of the array
    if (startnode != -1) {
614         printf(">>startnode != MaxPart: %7d\n", startnode);
        exit(1);
616     }
    // NOTE: ngb routine return particles in a "BOX" not only a
        SPHERE of radius hsml
618 // considere it when orig_target particle is found
    if (logthis(3)) printf("\n%d particles found in current hsml
        %6.6e around position [ %6.6e, %6.6e, %6.6e]\n", ngb,
        hsml, P[neigh_part].Pos[0], P[neigh_part].Pos[1], P[
        neigh_part].Pos[2]);

620 //iterate over all neighbours
622 for (n= 0; n < ngb; n++) { //we suppose that at least one
        neighbour particle is returned!!!
    this_part= Ngblist[n];

624     if ((this_part == neigh_part) || (this_part == orig_target)
        )
626         continue;

628     if (distance2(this_part , neigh_part) > hsml2) //only
        particles inside hsml allowed
        continue;

```

```

630 //calculate projected position over LOS
632 if (P[this_part].SFB_dProj_target != orig_target) {
634     get_vector( orig_target , this_part , v_op );
634     P[this_part].SFB_dProj= dot_prod( u_os, v_op ); //
        distance from orig_target
        P[this_part].SFB_dProj_target= orig_target;
636 }

638 if ((P[this_part].SFB_dProj <= 0.0) || (P[this_part].
        SFB_dProj >= d_os)) //particle getting away from sink
        continue;

640 //if (P[this_part].SFB_dProj < l)
642     //continue;

644 //calculate distance to LOS (h)
646 if (P[this_part].SFB_dLOS_target != orig_target) {
        advance_position(orig_target , P[this_part].SFB_dProj,
        u_os, ppos); //ppos contains the coordinates of
        particle projected over LOS
        P[this_part].SFB_dLOS= distanceTo(this_part , ppos);
648 P[this_part].SFB_dProj= d_os - P[this_part].SFB_dProj; //
        distance from sink
        P[this_part].SFB_dLOS_target= orig_target;
650 }

652 if (logthis(3)) printf("P[%d] at [ %6.6e, %6.6e, %6.6e],
        dProj=% 6.6e, dLOS=% 6.6e, adv=%3.6f*", this_part , P[
        this_part].Pos[0] , P[this_part].Pos[1] , P[this_part].
        Pos[2] , P[this_part].SFB_dProj, P[this_part].SFB_dLOS,
        100.0 * P[this_part].SFB_dProj / d_os);

654 if (P[this_part].SFB_dLOS <= min_h) {
        if (P[this_part].SFB_dProj < l) {
656             if (in_array(&selected_targets , this_part) != 1) {
                next_part= this_part;
658                 min_h= P[next_part].SFB_dLOS;
                if (logthis(3)) printf(" Min");
660             }
        }
662 }

664 if (logthis(3)) printf("\n");

```

```

666     } // for n=0

668     if ((next_part == neigh_part) && (sink_reached == 0)) { //no
        particles inside hsm1!
        printf("\nUpps! No selected particles found in current hsm1
            %6.6e arround at OT[%7d]->% 7d, position Ngh[ %6.6e,
            %6.6e, %6.6e], d_OS=% 6.6e\n", hsm1, orig_target, P[
            orig_target].ID, P[neigh_part].Pos[0], P[neigh_part].
            Pos[1], P[neigh_part].Pos[2], d_os);
670     printf("min_h=% 6.6e, ", min_h);
        break; //break while. What we do now? Probably increase
            hsm1
672     }
        if (next_part != neigh_part) { //we have a particle with
            absolute minimal h
674     array_push(&selected_targets, next_part); //save it on
            targets ( evaluation points ) stack
            if (logthis(3)) printf("Num. total selected %7d\n",
            selected_targets.count);
676     l= P[next_part].SFB_dProj; //update advanced position from
            sink
            hsm1= SphP[next_part].Hsm1;
678     hsm12= hsm1 * hsm1;
            neigh_part= next_part;
680     } //next_part != neigh_part

682 } //while ((l < d_os) && (sink_reached == 0))
684

686 show_selected_particles(this_sink, orig_target);

688 //Sort selected targets by dProj from min to max. We use an
        insertion sort algorithm
        for (i = 1; i < selected_targets.count; ++i) {
690     i_tmp= selected_targets.part[i];
            tmp = P[ selected_targets.part[i] ].SFB_dProj;
692     for (j = i; j > 0 && P[ selected_targets.part[j - 1] ].
            SFB_dProj > tmp; --j)
            selected_targets.part[j]= selected_targets.part[j - 1];
694     selected_targets.part[j]= i_tmp;
        }
696 sort_selected_particles(&selected_targets, &tmp_list);

```

```

698 array_copy(&tmp_list, &selected_targets);//only necessary to
    printout selected particles
    show_selected_particles(this_sink, orig_target);
700
    filter_particles(&selected_targets, &tmp_list);
702 array_copy(&tmp_list, &selected_targets);//only necessary to
    printout selected particles
704 show_selected_particles(this_sink, orig_target);
    if (logthis(3)) printf("%7d target particles found between
        sink and current target particle\n\n", selected_targets.
        count);
706
    //Calculate flux
708 if (logthis(3)) printf("Calculating Flux \n");
    sel_part= -1;
710 lsel_part= -1;
    N= 0.0;
712 TotI_r= 0.0;
    for (i= 0; i < selected_targets.count; i++) { // selected
        particles are stored from far to near sink
714 sel_part= selected_targets.part[i];
        r_i= P[sel_part].SFB_dProj;
716 if (lsel_part == -1) { //there is the only one particle
            selected (until now)
            Delta_ri= r_i;
718 n_ri= SphP[sel_part].Density;
        } else { //more than one particles have been selected
720 Delta_ri= r_i - lr_i; //r(i)-r(i-1)
            n_ri= 0.5 * (SphP[sel_part].Density + SphP[lsel_part].
                Density);
722 }
724
    //We aproximate the density at LOS using a (linear
        approximation of kernel)
    //if (P[sel_part].SFB_dLOS > 0)
726 //if ( P[sel_part].SFB_dLOS < hsml_frac )
        //n_ri= n_ri * ( 1.0 - (P[sel_part].SFB_dLOS / hsml));
728
    //calc flux from sink to this particle
730 N+= n_ri * Delta_ri ; //column density
    I_r= (r_i * r_i) * (n_ri * n_ri) * Delta_ri *
        SFB_4PI_ALPHA_RHO_FACTOR; // F_r = 4*pi * r_i^2 * <n_i
        >^2 * Delta_ri
732 TotI_r+= I_r;

```

```

flux_arriving= P[this_sink].SFB_photon_flux - TotI_r; //Q_h -
F_r
734 if (logthis(3)) printf("r=%6.6e, n=%6.6e*, hsml=%6.6e, Dlt
    =%6.6e, N=%6.6e*, adv=%3.6f*, dLOS=%6.6e, p=%d, flx=%6.6e
    , flxarr=%6.6e\n", r_i, n_ri, hsml, Delta_ri, N *
    SFB_RHO_FACTOR * All.UnitLength_in_cm, 100.0 * P[sel_part
    ].SFB_dProj / d_os, P[sel_part].SFB_dLOS, sel_part, I_r,
    flux_arriving);

736 if (flux_arriving < 0.0) {//flux insufficient to ionize orig
    target
    if (logthis(2)) printf("\nParticle [%7d]->% 7d not heated\n
        \n", orig_target, P[orig_target].ID);
738 //log_trace(orig_target, orig_target, SFB_TR_RESULT_OFF);
        //Results here
        //if (logthis(3)) log_part(orig_target, orig_target,
        SFB_TR_RESULT_OFF); //Results here
740 return 0; //do not heat particle
    }

742 lsel_part= sel_part; //save selected particle
744 lr_i= r_i;
    } //for

746 //Ionize the particle
748 P[orig_target].SFB_heating_ID2= P[this_sink].ID2;
    updated= update_temp(orig_target, SFB_TRUE);

750 if (logthis(2)) printf("\nParticle [%7d]->% 7d now heated\n\n",
    orig_target, P[orig_target].ID);
752 //show_selected_particles(this_sink, orig_target);
    //log_trace(orig_target, orig_target, SFB_TR_RESULT_ON); //
    Results here
754 //if (logthis(3)) log_part(orig_target, orig_target,
    SFB_TR_RESULT_ON); //Results here
    return updated; //heat particle
756 } //particle_ionization_orig_target

758 void coord_s2r(FLOAT r, FLOAT theta, FLOAT phi, FLOAT pos[]) {
760 pos[0]= r * cos(theta) * sin(phi);
    pos[1]= r * sin(theta) * sin(phi);
762 pos[2]= r * cos(phi);
    }

764

```

```

void cleanup_and_measures() {
766 float d, hsml_min, hsml_mean, min_dens, max_dens, min_dist, pos
      [3], min_dist_pos, d_pos, dx, dy, dz, r, theta, phi,
      deg2rad_factor;
768 int min_a, min_b, hsml_count, into_rs, min_dist_part;
int i, j;

770 if (logthis(1)) printf("Starting cleanup and measure routine\n"
      );
      gettimeofday(&SFB_T1, NULL);
772
      /* particles cleanup */
774 for (i=0; i<NumPart; i++) {
      P[i].SFB_heating_ID2= -All.MaxPart;
776 P[i].SFB_dLOS_target= -All.MaxPart;
      P[i].SFB_dProj_target= -All.MaxPart;
778 P[i].SFB_dSink= 0.0;
      P[i].SFB_dLOS= 0.0;
780 P[i].SFB_dProj= 0.0;
      }
782
      if (SFB_MEASURES != 1) {
784 gettimeofday(&SFB_T2, NULL);
      printf("Cleanup and measure time:\n");
786 time_diff(SFB_T1, SFB_T2);
      printf("\n");
788
      return;
790 }

792 hsml_min= All.BoxSize * All.BoxSize * All.BoxSize; //just some
      initial value
      hsml_mean= 0.0;
794 hsml_count= 0;
      min_dist= All.BoxSize;
796 min_a= -1;
      min_b= -1;
798 min_dens= 1.111111111e99;
      max_dens= 0.0;
800 into_rs= 0;

802 //We define the position near a test particle (for ionization)

804 //spherical coordinates (in degrees)
      r= (2.0/3.0) * SFB_RS_pc;

```

```

806  theta= 90.0 - 27.0;
      phi= 90.0;
808
      //conversion to radians
810  deg2rad_factor= 2.0 * PI / 360.0;
      theta= theta * deg2rad_factor;
812  phi= phi * deg2rad_factor;
      printf("Particle location at Spher. coord [%6.6e,%6.6e,%6.6e]\n
            ", r, theta, phi);
814
      coord_s2r(r, theta, phi, pos);
816  //translation to sink position
      pos[0]= 0.5 * All.BoxSize + pos[0];
818  pos[1]= 0.5 * All.BoxSize + pos[1];
      pos[2]= 0.5 * All.BoxSize + pos[2];
820  printf("Particle location at Rect. coord [%6.6e,%6.6e,%6.6e]\n
            ", pos[0], pos[1], pos[2]);

822  min_dist_part= 0;
      min_dist_pos= 3e30;
824
      for (i=0; i<NumPart; i++) {
826  if (!(P[i].Type == 0) && (P[i].ID >= 0)))
            continue;
828
            if ( SphP[i].Hsml < hsml_min )
830  hsml_min= SphP[i].Hsml;
            hsml_mean+= SphP[i].Hsml;
832  hsml_count++;
            if ( SphP[i].Density < min_dens )
834  min_dens= SphP[i].Density;
            if ( SphP[i].Density > max_dens )
836  max_dens= SphP[i].Density;

838
            dx= P[i].Pos[0] - pos[0];
840  dy= P[i].Pos[1] - pos[1];
            dz= P[i].Pos[2] - pos[2];
842  d_pos= sqrt((dx * dx) + (dy * dy) + (dz * dz));
            if ( d_pos < min_dist_pos ) {
844  min_dist_pos= d_pos;
            min_dist_part= i;
846  }

848  d= 0.0;

```

```

850     for (j=0; j<3; j++) {
        d+= ( P[i].Pos[j] - SFB_HALFBOXSIZE )*( P[i].Pos[j] -
            SFB_HALFBOXSIZE );
    }
852     if (sqrt(d) <= SFB_RS_pc)
        into_rs++;
854
    for (j= i+1; j<NumPart; j++) {
856         if (!(P[i].Type == 0) && (P[i].ID >= 0))
            continue;
858         d= sqrt(distance2(i, j));
            if ( d < min_dist ) {
860             min_dist= d;
                min_a= i;
862             min_b= j;
            }
864     }
}
866
hsml_mean/= (float) hsml_count;
868 i= min_a;
printf("Measuring HSML, density and distances\n");
870 printf("Minimal distance is %6.6e\n", min_dist);
printf("Min a: [%7d]->% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e,%
    6.6e\n", i, P[i].ID, P[i].Pos[0], P[i].Pos[1], P[i].Pos
    [2], SphP[i].Hsml, SphP[i].Density, SphP[i].Entropy);
872 i= min_b;
printf("Min b: [%7d]->% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e,% 6.6e,%
    6.6e\n", i, P[i].ID, P[i].Pos[0], P[i].Pos[1], P[i].Pos
    [2], SphP[i].Hsml, SphP[i].Density, SphP[i].Entropy);
874 printf("Minimal density is %6.6e\n", min_dens);
printf("Maximal density is %6.6e\n", max_dens);
876 printf("Minimal HSML is %6.6e\n", hsml_min);
printf("Mean HSML is %6.6e\n", hsml_mean);
878 printf("%7d particles are inside Stromgren radius (%6.6e)\n",
    into_rs, SFB_RS_pc);
i= min_dist_part;
880 SFB_DEBUG_PARTICLE= min_dist_part;
printf("Min dist part: [%7d]->% 7d,% 6.6e,% 6.6e,% 6.6e,% 6.6e
    ,% 6.6e,% 6.6e\n\n", i, P[i].ID, P[i].Pos[0], P[i].Pos[1],
    P[i].Pos[2], SphP[i].Hsml, SphP[i].Density, SphP[i].Entropy
    );
882 gettimeofday(&SFB_T2, NULL);
printf("Cleanup and measure time:\n");
884 time_diff(SFB_T1, SFB_T2);

```

```

886 }
888
890 /*
891  Iterates over all SPH particles, evaluating if they are heated
892  or not by the sink.
893  all heated particles are flagged by the "SFB_on" variable
894 */
895 void sink_feedback(struct sink_data_sort sinks_by_mass[N_sinks],
896                  int num_sinks_alive) {
897     struct sink_data_sort sinks_by_distance[N_sinks];
898     int i, j, k, this_sink, this_target, num_heated,
899         num_not_ionized;
900     signed int heated;
901     FLOAT d2;
902
903     cleanup_and_measures();
904     gettimeofday(&SFB_T1, NULL);
905     num_heated= 0;
906     if (logthis(1)) printf("Starting gas heating routine\n");
907     for (i=0; i<NumPart; i++) {
908         if (!(P[i].Type == 0) && (P[i].ID >= 0)) //only gas
909             particles
910             continue;
911         this_target= i;
912
913         //Check if there is need to (re-)ionize the particle
914         if (update_temp(this_target, SFB_FALSE) != -1) //t>10000 ?
915             continue;
916
917         //For each particle, we must sort the sinks according to his
918         distance to the particle
919         k= 0;
920         for (j=0; j<num_sinks_alive; j++) {
921             this_sink= sinks_by_mass[j].index;
922             if ((P[this_sink].SFB_radiating == 1) && ((P[this_sink].
923                 Type == 5) && (P[this_sink].ID < 0))) { //Only Sinks
924                 radiating
925                 //prepare sink data to sort by distance
926                 sinks_by_distance[k].index= this_sink;
927                 sinks_by_distance[k].data= distance2(this_sink,
928                     this_target);
929                 k++; //update num of sinks radiating
930             } //if sink_radiating==1

```

```

922     } //for j
      qsort(sinks_by_distance, k, sizeof(struct sink_data_sort),
            compare_sink_distances); //sort sinks by distance to
            target particle
924
      for (j=0; j<k; j++) { //iterate over all sinks from nearest
            to farthest
926         this_sink= sinks_by_distance[j].index;
            heated= particle_ionization(this_sink, this_target);
928         SFB_CURRENT_PARTICLE= -1;
            if ( heated == 1 ) { //Remember that only Radiating sinks
                are considered here
930                 num_heated++;
                    break; //only first ( nearest ) sink heating is allowed
932             } //if ( heated == -1) we don't update num_heated
            } //for j
934         if (SFB_LOG_TRACE != NULL) //dump inmediately the list to
            log_trace file
            fflush(SFB_LOG_TRACE);
936         fflush(stdout);

            //if (num_heated == SFB_MAX_HEATED_IN_LOOP)
            // break; // for i
940     } //for i

942     num_not_ionized= 0;
      if (logthis(1)) printf("Starting verification routine\n");
944     for (j=0; j<k; j++) { //iterate over all sinks from nearest to
            farthest
          for (i=0; i<NumPart; i++) {
946             if ((P[i].Type == 0) && (P[i].ID >= 0) && (P[i].
                SFB_heating_ID2 == -All.MaxPart)) { //only not ionized
                    gas particles
                d2= distance2(sinks_by_distance[j].index, i);
948                 if (d2 <= SFB_RS_pc2) {
                    num_not_ionized++;
950                     printf("Warning! particle not ionized: d_err=% 6.6e\n",
                        sqrt(d2) - SFB_RS_pc);
                }
952             } //if
            } //for
954     }
      printf("% 7d particles not ionized, please verify algorithm!\n"
            , num_not_ionized);
956     fflush(stdout);

```

```

958     if (num_not_ionized > 0) {
        //exit(1);
    }
960     if (logthis(1)) printf("Ending verification routine\n");
962     if (logthis(1)) printf("Ending gas heating routine\n");
964     if (logthis(1)) printf("\nIonization Results: %7d of %7d
        particles heated. \n",num_heated, NumPart);
        gettimeofday(&SFB_T2, NULL);
966     printf("Sink feedback time:\n");
        time_diff(SFB_T1, SFB_T2);
968 }

970 void SFB_do_sink_radiation() {
972     struct sink_data_sort sinks_by_mass[N_sinks];
        int i, this_sink, num_sinks_alive, num_sinks_radiating,
            num_stars_radiating, num_sinks_dead;
974     FLOAT gas_mass, sinks_mass, fracc_mass_radiating,
            sink_photon_flux, sink_num_ionizing_stars;
        FLOAT sink_radiating_elapsed;

976     if (logthis(1)) printf("Starting Sink Radiation routine\n");
978     //Should we execute this algorithm?
980     SFB_NUM_LOOP++;
        if (SFB_MAX_LOOP <= 1000) {
982         printf("Max loop allowed is %d\n", SFB_MAX_LOOP);
            if (SFB_NUM_LOOP > SFB_MAX_LOOP) { //this condition ignores
                the loops before a Gadget2 restart
984                 printf("Max num. of loops %d in Sink Feedback algorithm
                    reached. Current loop is %d.\n", SFB_MAX_LOOP,
                        SFB_NUM_LOOP);
                    SFB_terminate();
986                 return;
            }
988         else
            if (logthis(1)) printf("Sink Feedback Algorithm Loop Num. %
                7d\n",SFB_NUM_LOOP);
990     }

992     if (logthis(3)) printf("N_sinks= %7d\n",N_sinks);
994     if ( SFB_NUM_LOOP == 3 ) //Reduce output to the minimum

```

```

SFB_LOG_LEVEL= 1;
996
//Verify mass fraction based on IMF has been previously
    calculated
998 if (All.SFB_MassFraction == 0.0) {
    printf("Error: IFM based mass fraction not calculated!!!\n");
1000    SFB_terminate();
    exit(1);
1002 }

1004 if (logthis(1)) printf("Updating Sink's stars\n");
//Calculate total gas and sinks mass
1006 gas_mass= 0.0;
sinks_mass= 0.0;
1008 num_sinks_alive= 0;
//num_parts_to_heat= 0;
1010 for (i=0; i<NumPart; i++) {
    if ((P[i].Type == 0) && (P[i].ID >= 0))
1012        gas_mass += P[i].Mass;
    else if ((P[i].Type == 5) && (P[i].ID < 0)) { //Sinks only
1014        sinks_mass += P[i].Mass;
        //NOTE: we must update sink_alive here!!!
1016        if (P[i].SFB_alive == 1) {
            sinks_by_mass[num_sinks_alive].index= i;
1018            sinks_by_mass[num_sinks_alive].data= P[i].Mass;
            num_sinks_alive++;
1020        }
    } // else if
1022 } //for

1024 if (logthis(2)) printf("gass mass= %6.6e  sinks mass= %6.6e\n",
    gas_mass,sinks_mass);

1026 if ((sinks_mass == 0.0) || (num_sinks_alive == 0.0) ) {
    printf("No gas to heat nor sinks alive found\n");
1028    if (logthis(1)) printf("Ending Sink Radiation routine\n\n");
    return; //There is nothing to do
1030 }

1032 //Sort sinks by mass
fracc_mass_radiating= All.SFB_MassFraction * sinks_mass; //
    calculate the mass fraction of sinks radiating based on
    IMF (MSun)
1034 if (logthis(1)) printf("fracc. of mass radiating= %6.6e, IMF
    Mass Fraction= %6.6e,  sinks mass= %6.6e\n",

```

```

    fracc_mass_radiating, All.SFB_MassFraction, sinks_mass);
if (logthis(3)) printf("Sorting sinks by mass\n");
1036 qsort(sinks_by_mass, num_sinks_alive, sizeof(struct
    sink_data_sort), compare_sinks_mass);

1038 //identify which sinks are radiating
num_sinks_radiating= 0;
1040 num_stars_radiating= 0;
num_sinks_dead= 0;
1042 for (i=0; i<num_sinks_alive; i++) {
    this_sink= sinks_by_mass[i].index;
1044 if (P[this_sink].SFB_alive == 1) {
    printf("Sink alive: %d\n", this_sink);
1046 printf("IMF Mass Fraction=%6.3g, Sink Mass=%6.3g, Stars
        Mass=%6.3g\n", All.SFB_MassFraction, P[this_sink].Mass,
        SFB_STAR_MASS);
    sink_num_ionizing_stars= (int) ( All.SFB_MassFraction * P[
        this_sink].Mass / SFB_STAR_MASS );
1048 sink_photon_flux= sink_num_ionizing_stars *
        SFB_STAR_PHOTON_FLUX;
    if ( P[this_sink].SFB_radiating == 0) {
1050 if ( sink_num_ionizing_stars > 0.0 ) { //check if the
        sink have ionizing stars within
        P[this_sink].SFB_num_ionizing_stars=
            sink_num_ionizing_stars;
1052 P[this_sink].SFB_photon_flux= sink_photon_flux;
        P[this_sink].SFB_radiating= 1;
1054 P[this_sink].SFB_time_on= All.Time;
        P[this_sink].SFB_radiating_elapsed= 0.0;
1056 num_stars_radiating+= P[this_sink].
            SFB_num_ionizing_stars;
        num_sinks_radiating++;
1058 if (logthis(1)) printf("New Sink : ID2= %7d, Mass= %
            6.6e, Ionizing Stars= %6.6e, Photon Flux= %6.6e,
            Time elapsed= %6.6e\n", P[this_sink].ID2, P[
            this_sink].Mass, P[this_sink].
            SFB_num_ionizing_stars, P[this_sink].
            SFB_photon_flux, P[this_sink].SFB_radiating_elapsed
            );
        log_part(this_sink, this_sink, SFB_TR_SINK);
1060     }
    }
1062 else { //sink is already radiating ( sink_radiating == 1 )
    if ( P[this_sink].SFB_num_ionizing_stars !=
        sink_num_ionizing_stars ) {

```

```

1064     P[this_sink].SFB_num_ionizing_stars=
        sink_num_ionizing_stars;
1066     P[this_sink].SFB_photon_flux= sink_photon_flux;
        if (logthis(1)) printf("Updt Sink: ID2= %7d, Mass= %
        6.6e, Ionizing Stars= %6.6e, Photon Flux= %6.6e,
        Time elapsed= %6.6e\n", P[this_sink].ID2, P[
        this_sink].Mass, P[this_sink].
        SFB_num_ionizing_stars, P[this_sink].
        SFB_photon_flux, P[this_sink].SFB_radiating_elpased
        );
        log_part(this_sink ,this_sink ,SFB_TR_SINK);
1068     }
        //check if we need to turn off radiating sinks based on
        stars timelife
1070     P[this_sink].SFB_radiating_elpased=( All.Time - P[
        this_sink].SFB_time_on ) * All.UnitTime_in_s;
        if ( P[this_sink].SFB_radiating_elpased >=
        SFB_STAR_LIFETIME_ESTIMATE ) { //turn off radiation
1072     P[this_sink].SFB_radiating= 0;
        P[this_sink].SFB_alive= 0;
1074     num_sinks_dead++;
        if (logthis(1)) printf("Dead Sink: ID2= %7d, Mass= %
        6.6e, Ionizing Stars= %6.6e, Photon Flux= %6.6e,
        Time elapsed= %6.6e\n", P[this_sink].ID2, P[
        this_sink].Mass, P[this_sink].
        SFB_num_ionizing_stars, P[this_sink].
        SFB_photon_flux, P[this_sink].SFB_radiating_elpased
        );
1076     log_part(this_sink ,this_sink ,SFB_TR_SINK);
        }
1078     else { // Sink is not enough time radiating
        num_stars_radiating+= P[this_sink].
        SFB_num_ionizing_stars;
1080     num_sinks_radiating++;
        }
1082     } //sink radiating == 1
        } //sink_alive=1
1084     } //for
        if (logthis(1)) printf("Sinks alive= %7d, sinks radiating= %7
        d, stars radiating= %7d, sinks dead= %7d\n",
        num_sinks_alive , num_sinks_radiating , num_stars_radiating ,
        num_sinks_dead);
1086
1088     if (num_sinks_radiating == 0) {
        printf("No sinks radiating found\n");

```

```

1090     return; //There is nothing to do
1091 }
1092 sink_feedback(sinks_by_mass, num_sinks_alive);
1093
1094 if ( SFB_NUM_LOOP == 1 ) //dump here to get Entropy and other
1095     interesting data
1096     log_positions();
1097 if (logthis(1)) printf("Ending Sink Radiation routine\n\n");
1098 }
1099
1100 void SFB_initialize() {
1101     SFB_parameters();
1102     All.SFB_MassFraction= SFB_calculate_imf(SFB_STAR_MASS);//
1103         calculate fraction of gas merged into XXMsun sinks,
1104 }
1105
1106 void SFB_terminate() {
1107     if (SFB_TERMINATED!=0) {
1108         printf("\nTerminating Sink Radiation routine and cleaning all
1109             SFB variables and files\n\n");
1110         close_log_file(&SFB_LOG_ALLPOS);
1111         close_log_file(&SFB_LOG_TRACE);
1112         SFB_TERMINATED= 1;
1113     }
1114 }
1115
1116 #endif //SINK_FEEDBACK

```

### A.3. Herramientas computacionales adicionales

GADGET2 arroja los datos de las partículas (posiciones, velocidades, temperaturas) dentro de archivos binarios llamados “*snapshots*” con una periodicidad definida por el usuario, los cuales son numerados secuencialmente. Para visualizar, analizar y comparar los resultados proporcionados por la simulación contra los resultados analíticos, fue conveniente desarrollar una serie de *scripts* para el software IDL. Adicionalmente se emplearon algunos scripts desarrollado con anterioridad en el CRyA para importar los datos de los snapshots dentro de IDL, en particular:

`utils.pro` Especifica las unidades del código GADGET2 empleadas en la simulación, algunas factores de conversión y valores astrofísicos útiles para algunos cálculos. También incluye algunas rutinas auxiliares en la salida de datos numéricos y gráficos.

`coolheat.pro` Programa auxiliar que permite evaluar y graficar la presión térmica del gas y determinar su temperatura de equilibrio a la densidad del medio neutro.

`evolution.pro` Este programa contiene dos rutinas básicas. La primera es `evolution`, en ella se indica el número de snapshot y despliega las gráficas de las partículas SPH, así como la temperatura y la densidad, indicando la ubicación del frente de ionización. La segunda rutina es `evolve`, la cual, considera una secuencia continua de snapshots y muestra la gráfica comparativa de la evolución del frente de ionización.

`thesis.pro` Este programa permite graficar las curvas de fracción de ionización y la evolución del frente de ionización en unidades adimensionales. Adicionalmente, permite calcular el radio de Strömgren, conversiones entre temperatura y energía interna, el flujo ionizante para distintos tipos de estrellas, cocientes de densidad, radio de la esfera de Strömgren y masa iniciales y finales para una región HII, densidad de columna y el consumo de fotones a densidad constante, así como el tiempo de enfriamiento del gas. La mayoría de los cálculos están basados en las ecuaciones mencionadas en el presente trabajo.

`units.pro` Especifica las unidades del código GADGET2 empleadas en la simulación, así como algunos factores de conversión.

`read_snap.pro` Este archivo permite leer los datos a partir de un archivo de snapshot (binario) y emplea el archivo `units.pro` para reescalar las lecturas de las variables correspondientes a las partículas SPH.



# Bibliografía

- [1] Armstrong J. W., Rickett B. J. & Spangler S.R. 1995. Electron density power spectrum in the local interstellar medium. 1995ApJ...443..209A.
- [2] Bagla J. S. 2002. TreePM: A Code for Cosmological N-Body Simulations. 2002JApA...23..185B.
- [3] Ballesteros-Paredes J., Vázquez-Semadeni E. & Scalo J. 1999. Clouds as Turbulent Density Fluctuations: Implications for Pressure Confinement and Spectral Line Data Interpretation. 1999ApJ...515..286B.
- [4] Ballesteros-Paredes J., Klessen R. S., Mac Low M.-M., Vázquez-Semadeni E. 2007. Molecular cloud turbulence and star formation. 2007prpl.conf...63B.
- [5] Ballesteros-Paredes J., Hartmann L. W., Vázquez-Semadeni E., Heitsch F. & Zamora-Avilés M. A. 2010. Gravity or turbulence. Velocity dispersion-size relation. 2010MNRAS.tmp.1635B.
- [6] Bania T. M. & Lyon J. G. 1980. OB star and the structure of the interstellar medium: cloud formation and effects of different equations of state. 1980ApJ...239..173B.
- [7] Barnes, J. & Hut, P. 1986. A hierarchical  $O(N \log N)$  force-calculation algorithm. 1986Natur.324..446B.
- [8] Bate, M. R., Bonnell, I. A., Price, N. M. 1995. Modelling accretion in protobinary systems. 1995MNRAS.277..362B.

- 
- [9] Bertoldi, F. & McKee, C.F. 1992. Pressure-confined clumps in magnetized molecular clouds. 1992ApJ...395..140B.
- [10] Binney, J. & Merrifield M. 1998. Galactic astronomy. Princeton University Press. 1998gaas.book.....B.
- [11] Blitz, L. 1993. Giant molecular clouds. 1993prpl.conf..125B.
- [12] Blitz, L. & Williams, J. P. 1999. Molecular Clouds. 1999os-ps.conf....3B.
- [13] Bode, P., Ostriker, J. P. & Xu, G. 2000. The Tree Particle-Mesh N-Body Gravity Solver. 2000ApJS..128..561B.
- [14] Bonnell I. A., Larson R. B. & Zinnecker H. 2006. The origin of the initial mass function. arXiv:astro-ph/0603447.
- [15] Boulares, A. & Cox, D. P. 1990. Galactic hydrostatic equilibrium with magnetic tension and cosmic-ray diffusion. 1990ApJ...365..544B.
- [16] Bourke, T. L., Myers, P. C., Robinson, G. & Hyland, A. R. 2001. New OH Zeeman Measurements of Magnetic Field Strengths in Molecular Clouds. 2001ApJ...554..916B.
- [17] Brunt, C. M. 2003. Large-Scale Turbulence in Molecular Clouds. 2003ApJ...583..280B.
- [18] Burgers, J. M. 1974. Statistical problems connected with the one-dimensional nonlinear diffusion equation. 1974ASAJ...55...50B.
- [19] Burke, J. R. & Hollenbach, D. J. 1983. The gas-grain interaction in the interstellar medium - Thermal accommodation and trapping. 1983ApJ...265..223B.
- [20] Burkert, A. & Hartmann, L. 2004. Collapse and Fragmentation in Finite Sheets. 2004ApJ...616..288B.
- [21] Carroll B. W. & Ostlie D. A. 1996. An introduction to modern astrophysics, AddisonWesley. 1996ima..book.....C

- 
- [22] Carroll, J. J., Frank, A., Blackman, E. G., Cunningham, A. J., Quillen, A. C. 2009. Outflow-Driven Turbulence in Molecular Clouds. 2009ApJ...695.1376C.
- [23] Carroll, J. J., Frank, A., Blackman, E. G. 2010. Isotropically Driven Versus Outflow Driven Turbulence: Observational Consequences for Molecular Clouds. 2010ApJ...722..145C.
- [24] Chiang, W.-H. & Prendergast, K. H. 1985. Numerical study of a two-fluid hydrodynamic model of the interstellar medium and population I stars. 1985ApJ...297..507C.
- [25] Chiang, W.-H. & Bregman, J. N. 1988. A model for the interaction between stars and gas in the interstellar medium. 1988ApJ...328..427C.
- [26] Cohen, M. 1995. The displacement of the sun from the galactic plane using IRAS and FAUST source counts. 1995ApJ...444..874C.
- [27] Colín, P., Avila-Reese, V., Vázquez-Semadeni, E., Valenzuela, O. & Ceverino, D. 2010. Low-mass Galaxy Formation in Cosmological Adaptive Mesh Refinement Simulations: The Effects of Varying the Sub-grid Physics Parameters. 2010ApJ...713..535C.
- [28] Cox, A. N. 2004. Allen's astrophysical quantities, 4th. Edition, AIP Press.
- [29] Cox, D. P., Smith, B. W. 1974. Large-Scale Effects of Supernova Remnants on the Galaxy: Generation and Maintenance of a Hot Network of Tunnels. 1974ApJ...189L.105C.
- [30] Cox, D. P. 2005. The Three-Phase Interstellar Medium Revisited. 2005ARA&A..43..337C.
- [31] Crutcher, R., Heiles, C., Troland, T. 2003. Observations of Interstellar Magnetic Fields. 2003LNP...614..155C.
- [32] Crutcher, R. M., Hakobian, N., Troland, T. H. 2009. Testing Magnetic Star Formation Theory. 2009ApJ...692..844C.

- [33] Dale J. E., Ercolano B. & Clarke C. J., 2007. A new algorithm for modeling photoionizing radiation in smoothed particle hydrodynamics. 2007MNRAS.382.1759D.
- [34] Dalgarno, A. & McCray, R. A. 1972. Heating and Ionization of HI Regions. 1972ARA&A..10..375D.
- [35] Dame, T. M., Ungerechts, H., Cohen, R. S., de Geus, E. J., Grenier, I. A., May, J., Murphy, D. C., Nyman, L.-A. & Thaddeus, P. 1987. A composite CO survey of the entire Milky Way. 1987ApJ...322..706D.
- [36] de Avillez, M. A. 2000. Disc-halo interaction - I. Three-dimensional evolution of the Galactic disc. 2000MNRAS.315..479D.
- [37] de Avillez, M. A. & Breitschweidt, D. 2004. Volume filling factors of the ISM phases in star forming galaxies. I. The role of the disk-halo interaction. 2004A&A...425..899D.
- [38] de Avillez, M. A. & Breitschweidt, D. 2005. Global dynamical evolution of the ISM in star forming galaxies. I. High resolution 3D simulations: Effect of the magnetic field. 2005A&A...436..585D.
- [39] de Jong, T., Boland, W. & Dalgarno, A. 1980. Hydrostatic models of molecular clouds. 1980A&A....91...68D.
- [40] Dieter, N. H. 1964. Neutral hydrogen near the north galactic pole. 1964AJ.....69..288D.
- [41] Dyson J. E. & Williams, D. A. 1980. The physics of the interstellar medium. ISBN: 0-201-54730-9.
- [42] Elmegreen, B. G. 1983. Quiescent formation of bound galactic clusters. 1983MNRAS.203.1011E.
- [43] Elmegreen, B. G. & Elmegreen, D. M. 1983. Regular strings of H II regions and superclouds in spiral galaxies - Clues to the origin of cloudy structure. 1983MNRAS.203...31E.

- [44] Elmegreen, B. G. 1993. Star Formation at Compressed Interfaces in Turbulent Self-gravitating Clouds. 1993ApJ...419L..29E.
- [45] Elmegreen B. G. & Scalo J., 2004. Interstellar turbulence I: Observations and Processes. 2004ARA&A..42..211E.
- [46] Falgarone E. & Puget, J.L. 1986. Model of clumped molecular clouds. II - Physics and evolution of the hierarchical structure. 1986A&A...162..235F.
- [47] Ferrière, K. M. 2001. The Interstellar Environment of our Galaxy. 2001RvMP...73.1031F
- [48] Field, G. B. & Saslaw, W. C. 1965. A Statistical Model of the Formation of Stars and Interstellar Clouds. 1965ApJ...142..568F.
- [49] Field, G. B., Somerville, W. B. & Dressler, K. 1966. Hydrogen Molecules in Astronomy. 1966ARA&A...4..207F.
- [50] Field, G. B., Goldsmith, D. W., Habing, H. J. 1969, Cosmic-Ray Heating of the Interstellar Gas. 1969ApJ...155L.149F
- [51] Field, G. B., Blackman, E. G., Keto, E. R. 2008. A model of cloud fragmentation. 2008MNRAS.385..181F.
- [52] Franco, J. & Cox, D.P. 1983. Self-regulated star formation in the galaxy. 1983ApJ...273..243F.
- [53] Franco, J. & Cox, D.P. 1986. Molecular clouds in galaxies with different Z - Fragmentation of diffuse clouds driven by opacity. 1986PASP...98.1076F.
- [54] Franco, J., Shore, S. N., Tenorio-Tagle, G. 1994. On the massive star-forming capacity of molecular clouds. 1994ApJ...436..795F.
- [55] Franco, J., Kim, J., Alfaro, E. J. & Hong, S. S. 2002. The Parker Instability in Three Dimensions: Corrugations and Superclouds along the Carina-Sagittarius Arm. 2002ApJ...570..647F.

- [56] Frisch, U. 1995. Turbulence. The legacy of A. N. Kolmogorov. 1995tlan.book....F.
- [57] Galván-Madrid, R., Vázquez-Semadeni, E., Kim, J. & Ballesteros-Paredes, J. 2007. Statistics of Core Lifetimes in Numerical Simulations of Turbulent, Magnetically Supercritical Molecular Clouds. 2007ApJ...670..480G.
- [58] Gibson, S. J., Taylor, A. R., Higgs, L. A. & Dewdney, P. E. 2000. A New View of Cold H I Clouds in the Milky Way. 2000ApJ...540..851G
- [59] Gingold, R. A. & Monaghan, J. J. 1977. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. 1977MNRAS.181..375G.
- [60] Goldreich, P. & Kwan, J. 1974. Molecular Clouds. 1974ApJ...189..441G.
- [61] Goldsmith, P. F. 1987. Molecular clouds - an overview. 1987ASSL..134...51G.
- [62] Grant, H. L. & Moilliet, A. 1962. The spectrum of a cross-stream component of turbulence in a tidal stream. 1962JFM....13..237G.
- [63] Groenewegen, M. A. T., Udalski, A., Bono, G. 2008. The distance to the Galactic centre based on Population II Cepheids and RR Lyrae stars. 2008A&A...481..441G.
- [64] Groom, W. 1997. Ph.D. thesis, University of Cambridge.
- [65] Haffner, L. M., Reynolds, R. J. & Tufte, S. L. 1999. WHAM Observations of H $\alpha$ , [S II], and [N II] toward the Orion and Perseus Arms: Probing the Physical Conditions of the Warm Ionized Medium. 1999ApJ...523..223H.
- [66] Hartman, L., Ballesteros-Paredes, J. & Bergin, E. A. 2001. Rapid Formation of Molecular Clouds and Stars in the Solar Neighborhood. 2001ApJ...562..852H.

- [67] Hartmann, L. & Burkert, A. 2007. On the Structure of the Orion A Cloud and the Formation of the Orion Nebula Cluster. 2007ApJ...654..988H.
- [68] Heides, C. & Troland, T. H. 2003. The Millennium Arecibo 21 Centimeter Absorption-Line Survey. II. Properties of the Warm and Cold Neutral Media. 2003ApJ...586.1067H
- [69] Heitsch, F., Burkert, A., Hartmann, L. W., Slyz, A. D. & Devriendt, J. E. G. 2005. Formation of Structure in Molecular Clouds: A Case Study. 2005ApJ...633L.113H.
- [70] Heitsch, F. & Hartmann, L. 2008. Rapid Molecular Cloud and Star Formation: Mechanisms and Movies. 2008ApJ...689..290H.
- [71] Heitsch, F., Slyz, A. D., Devriendt, J. E. G., Hartmann, L. W. & Burkert, A. 2006. The Birth of Molecular Clouds: Formation of Atomic Precursors in Colliding Flows. 2006ApJ...648.1052H.
- [72] Hennebelle, P. & Pérault, M. 1999. H<sub>2</sub> formation during thermal condensation in trans-sonic flows. 1999h2sp.confE..60H.
- [73] Hennebelle, P. & Pérault, M. 2000. Dynamical condensation in a magnetized and thermally bistable flow. Application to interstellar cirrus. 2000A&A...359.1124H.
- [74] Hennebelle P., Mac Low M.-M. & Vázquez-Semadeni E., 2007. Diffuse interstellar medium and the formation of molecular clouds. 2007arXiv0711.2417H.
- [75] Hennebelle, P., Mac Low, M.-M. & Vázquez-Semadeni, E. 2009. Diffuse interstellar medium and the formation of molecular clouds. 2009sfa..book..205H.
- [76] Hernquist, L. & Katz, N. 1989. TREESPH - A unification of SPH with the hierarchical tree method. 1989ApJS...70..419H.

- [77] Hiotelis, N. & Voglis, N. 1991. Smooth particle hydrodynamics with locally readjustable resolution in the collapse of a gaseous protogalaxy. 1991A&A...243..333H.
- [78] Hollenbach, D. & Salpeter, E. E. 1971. Surface Recombination of Hydrogen Molecules. 1971ApJ...163..155H.
- [79] Hollenbach, D. J. & Tielens, A. G. G. M. 1999. Photodissociation regions in the interstellar medium of galaxies. 1999RvMP...71..173H.
- [80] Hoyle, F. 1953. On the Fragmentation of Gas Clouds Into Galaxies and Stars. 1953ApJ...118..513H.
- [81] IIT, Visual Information Solutions. IDL Software, Interactive Data Language.
- [82] Jappsen A. K., Klessen R. S., R. B. Larson, Yet Li, Mordecai M. Mac Low, 2005. The stellar mass spectrum from non-isothermal gravoturbulent fragmentation. 2005A&A...435..611J
- [83] Katz, J. I. & Green, M.L. 1986. A Burgers model of interstellar dynamics. 1986A&A...161..139K.
- [84] Kennicutt, R.C. Jr. 1998. The Global Schmidt Law in Star-forming Galaxies. 1998ApJ...498..541K.
- [85] Kerr, F.J. & Lynden-Bell, D. 1986. Review of galactic constants. 1986MNRAS.221.1023K.
- [86] Kessel-Deynet, O. & Burkert A. 2000. Ionizing radiation in smoothed particle hydrodynamics. 2000MNRAS.315..713K.
- [87] Kim, W.-T. & Ostriker, E. C. 2001. Amplification, Saturation, and Q Thresholds for Runaway: Growth of Self-Gravitating Structures in Models of Magnetized Galactic Gas Disks. 2001ApJ...559...70K.
- [88] Klessen, R. S., Heitsch, F. & Mac Low, M.-M. 2000. Gravitational Collapse in Turbulent Molecular Clouds. I. Gasdynamical Turbulence. 2000ApJ...535..887K.

- [89] Klessen, R. S. & Hennebelle, P. 2010. Accretion-driven turbulence as universal process: galaxies, molecular clouds, and protostellar disks. 2010A&A...520A..17K.
- [90] Kolmogorov, A. 1941. The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers. 1941DoSSR..30..301K.
- [91] Korpi, M. J., Brandenburg, A., Shukurov, A., Tuominen, I., Nordlund, Å. 1999. A Supernova-regulated Interstellar Medium: Simulations of the Turbulent Multiphase Medium. 1999ApJ...514L..99K.
- [92] Koyama, H. & Inutsuka, S.-I. 2000. Molecular Cloud Formation in Shock-compressed Layers. 2000ApJ...532..980K.
- [93] Koyama, H. & Inutsuka, S.-I. 2002. An Origin of Supersonic Motions in Interstellar Clouds. 2002ApJ...564L..97K.
- [94] Kritsuk, A. G., Norman, M. L., Padoan, P. & Wagner, R. 2007. The Statistics of Supersonic Isothermal Turbulence. 2007ApJ...665..416K
- [95] Kroupa, P. 2001. On the variation of the initial mass function. 2001MNRAS.322..231K.
- [96] Krumholz, M. R. & McKee, C. F. 2005. A General Theory of Turbulence-regulated Star Formation, from Spirals to Ultraluminous Infrared Galaxies. 2005ApJ...630..250K.
- [97] Krumholz, M. R., Matzner, C.D. & McKee, C.F. 2006. The Global Evolution of Giant Molecular Clouds. I. Model Formulation and Quasi-Equilibrium Behavior. 2006ApJ...653..361K.
- [98] Lada, C. J. & Lada, E. A. 2003. Embedded Clusters in Molecular Clouds. 2003ARA&A..41...57L.
- [99] Landau & Lifshitz 1982. Fluid Mechanics, Second Edition: Volume 6 (Course of Theoretical Physics). ISBN-13: 978-0750627672

- 
- [100] Larson, R. B. 1981. Turbulence and star formation in molecular clouds. 1981MNRAS.194..809L.
- [101] Li, Z.-Y. & Nakamura, F. 2006. Cluster Formation in Protostellar Outflow-driven Turbulence. 2006ApJ...640L.187L.
- [102] Lucy, L. B. 1977. A numerical approach to the testing of the fission hypothesis. 1977AJ.....82.1013L.
- [103] Mac Low, M.-M. 2002. Turbulence in the Interstellar Medium: Energetics and Driving Mechanisms. 2002astro.ph.11616M.
- [104] Mac Low, M.-M. & Klessen, R. S. 2004. Control of star formation by supersonic turbulence. 2004RvMP...76..125M.
- [105] Magnani, L., Hartmann, D. Speck, B. G. 1996. A Catalog of Molecular Gas at High Galactic Latitudes. 1996ApJS..106..447M.
- [106] Matzner, C. D. & McKee, C. F. 2000. Efficiencies of Low-Mass Star and Star Cluster Formation. 2000ApJ...545..364M.
- [107] Matzner, C. D. 2002. On the Role of Massive Stars in the Support and Destruction of Giant Molecular Clouds. 2002ApJ...566..302M.
- [108] Matzner, C. D. 2007. Protostellar Outflow-driven Turbulence. 2007ApJ...659.1394M.
- [109] McCray, R. & Snow, T. P. Jr. 1979. The violent interstellar medium. 1979ARA&A..17..213M.
- [110] McKee, C. F. & Ostriker, J. P. 1977. A theory of the interstellar medium - Three components regulated by supernova explosions in an inhomogeneous substrate. 1977ApJ...218..148M.
- [111] McKee, C. F. 1989. Photoionization-regulated star formation and the structure of molecular clouds. 1989ApJ...345..782M.

- [112] McKee, C. F. 1990. The three phase model of the interstellar medium - Where does it stand now?. 1990ASPC...12....3M.
- [113] McKee, C. F. & Ostriker, E. C. 2007. Theory of Star Formation. 2007ARA&A..45..565M.
- [114] Mihalas D. & Binney, J. 1981. Galactic astronomy: Structure and kinematics, 2nd edition. 1981gask.book.....M
- [115] Miller, G. E. & Scalo, J. M. 1979. The initial mass function and stellar birthrate in the solar neighborhood. 1979ApJS...41..513M.
- [116] Monaghan, J. J. & Lattanzio, J. C. 1985. A refined particle method for astrophysical problems. 1985A&A...149..135M.
- [117] Monaghan, J. J. 1989. On the problem of penetration in particle methods. 1989JCoPh..82....1M.
- [118] Monaghan J. J. 1992. Smoothed Particle Hydrodynamics. 1992ARA&A..30..543M.
- [119] Mouschovias, T. Ch. 1991. Magnetic braking, ambipolar diffusion, cloud cores, and star formation - Natural length scales and protostellar masses. 1991ApJ...373..169M.
- [120] Myers, P. C., Dame, T. M., Thaddeus, P., Cohen, R. S., Silverberg, R. F., Dwek, E. & Hauser, M. G. 1986. Molecular clouds and star formation in the inner galaxy - A comparison of CO, H II, and far-infrared surveys. 1986ApJ...301..398M.
- [121] Nakamura, F. & Li, Z.-Y. 2007. Protostellar Turbulence Driven by Collimated Outflows. 2007ApJ...662..395N.
- [122] Nakano, T. & Nakamura, T. 1978. Gravitational Instability of Magnetized Gaseous Disks 6. 1978PASJ...30..671N.
- [123] Norman, C. & Silk, J. 1980. Clumpy molecular clouds - A dynamic model self-consistently regulated by T Tauri star formation. 1980ApJ...238..158N.

- [124] Obukhov, A.M. 1941. On the spectral distribution of energy in turbulent. *Izvestiya Acad. Sci. USSR, Geogr. and Geophys. Series 4*
- [125] Oort, J. H. 1966. Possible Interpretations of the High-Velocity Clouds. 1966BAN....18..421O.
- [126] Ossenkopf, V. & Mac Low, M.-M. 2002. Turbulent velocity structure in molecular clouds. 2002A&A...390..307O.
- [127] Osterbrock, Donald E. 1989. *Astrophysics of Gaseous Nebulae and Active Galactic Nuclei*. ISBN-13: 978-0935702224.
- [128] Padoan, P., Juvela, M., Kritsuk, A., Norman, M. L. 2009. The Power Spectrum of Turbulence in NGC 1333: Outflows or Large-Scale Driving?. 2009ApJ...707L.153P.
- [129] Passot, T., Vázquez-Semadeni, E. & Pouquet, A. 1995. A Turbulent Model for the Interstellar Medium. II. Magnetic Fields and Rotation. 1995ApJ...455..536P.
- [130] Porter, D. H., Pouquet, A., Woodward, P. R. 1994. Kolmogorov-like spectra in decaying three-dimensional supersonic flows. 1994PhFl....6.2133P.
- [131] Porter, D. H. Woodward, P. R. & Pouquet A. 1995. Inertial range structures in decaying compressible turbulent flows. doi:10.1063/1.869563.
- [132] Press W. H., Teukolsky S. A., Vetterling W. T. & Flannery B. P. 1986. *Numerical Recipes in Fortran 77. The Art of Scientific Computing*, ISBN: 0-521-43064-X.
- [133] Reynolds, R. J. 1985. The forbidden line of S II lambda 6716 in the galactic emission-line background. 1985ApJ...294..256R.
- [134] Rosen, A., Bregman, J. N. & Norman, M. L. 1993. Hydrodynamical simulations of star-gas interactions in the interstellar medium with an external gravitational potential. 1993ApJ...413..137R.

- [135] Saffman, P.G. 1970. Dependence on Reynolds Number of High-Order Moments of Velocity Derivatives in Isotropic Turbulence. 1970PhFl...13.2193S.
- [136] Salpeter, E. E. 1955. The Luminosity Function and Stellar Evolution. 1955ApJ...121..161S.
- [137] Scalo, J. M. 1988. Theories and Implications of Hierarchical Fragmentation. 1988LNP...315..201S.
- [138] Schmidt, M. 1959. The Rate of Star Formation. 1959ApJ...129..243S.
- [139] Sedgewick R., 2005. Algoritmos en C++. ISBN:968-44-401-X.
- [140] Shu, F. H. 1992. The physical universe. An introduction to astronomy. ISBN-13: 978-0935702057.
- [141] Shu, Adams & Lizano 1987. Star formation in molecular clouds - Observation and theory. 1987ARA&A..25...23S.
- [142] Smidth, A. M. & Stecher, T. P. 1971. Interstellar CO in the Spectrum of  $\zeta$  Oph. 1971BAAS....3..250S.
- [143] Spitzer, L. Jr. & Savedoff, M. P. 1950. The Temperature of Interstellar Matter. III. 1950ApJ...111..593S.
- [144] Spitzer, L. Jr. 1988. Ultraviolet absorption studies of the interstellar gas. 1988PASP..100..518S.
- [145] Spitzer, L. Jr. 1990. Theories of the hot interstellar gas. 1990ARA&A..28...71S.
- [146] Spitzer, L. Jr. 1998. Physical Processes in the Interstellar Medium. ISBN: 0-471-29335-0.
- [147] Springel V., Yoshida N. & White S. D. M. 2001. GADGET: a code for collisionless and gasdynamical cosmological simulations. 2001NewA....6...79S.

- [148] Springel V., 2005. The cosmological simulation code GADGET2. 2005MNRAS.364.1105S.
- [149] Springel V., 2005. User guide for GADGET2. <http://www.mpa-garching.mpg.de/gadget/>
- [150] Strömgren, B. 1939. The Physical State of Interstellar Hydrogen. 1939ApJ....89..526S.
- [151] Troland, T. H. & Crutcher, R. M. 2008. Magnetic Fields in Dark Cloud Cores: Arecibo OH Zeeman Observations. 2008ApJ...680..457T.
- [152] Vázquez-Semadeni, E., Ballesteros-Paredes, J. & Rodríguez, L. F. 1995. Low-Column Density Clouds in Numerical Simulations of the Interstellar Medium. 1995AAS...187.2002V.
- [153] Vázquez-Semadeni, E., Passot, T., Pouquet, A. 1996. Influence of Cooling-induced Compressibility on the Structure of Turbulent Flows and Gravitational Collapse. 1996ApJ...473..881V.
- [154] Vázquez-Semadeni E., Ostriker E.C., Passot T., Gammie C. F. & Stone J.M. 1999. Compressible MHD Turbulence: Implications for Molecular Cloud and Star Formation. arXiv:astro-ph/9903066v1.
- [155] Vázquez-Semadeni, E., Ostriker, E. C., Passot, T., Gammie, C. F., Stone, J. M. 2000. Compressible MHD Turbulence: Implications for Molecular Cloud and Star Formation. 2000prpl.conf....3V.
- [156] Vázquez-Semadeni, E., Ballesteros-Paredes, J. & Klessen R. S. 2003. A holistic scenario of turbulent molecular cloud evolution and control of the star formation efficiency: First tests. 2003ApJ...585L.131V.
- [157] Vázquez-Semadeni E., Gasol A., Passot T. 2003. Thermal Instability and Magnetic Pressure in the Turbulent Interstellar Medium. 2003LNP...614..213V.

- [158] Vázquez-Semadeni, E., Kim, J., Shadmehri, M. & Ballesteros-Paredes, J. 2005. The Lifetimes and Evolution of Molecular Cloud Cores. 2005ApJ...618..344V.
- [159] Vázquez-Semadeni, E., Ryu, D., Passot, T., González, R. F., Gazol, A. 2006. Molecular Cloud Evolution. I. Molecular Cloud and Thin Cold Neutral Medium Sheet Formation. 2006ApJ...643..245V
- [160] Vázquez-Semadeni E., Gómez G. C., Jappsen A. K., Ballesteros-Paredes J., González R.F. & Klessen R.S., 2007. Molecular cloud evolution II. From Cloud Formation to the Early Stages of Star Formation in Decaying Conditions. 2007ApJ...657..870V
- [161] Vázquez-Semadeni, E., Ballesteros-Paredes, J., Klessen, R. S. & Jappsen, A. K. 2008. Massive Star-Forming Regions: Turbulent Support or Global Collapse?. 2008ASPC..387..240V.
- [162] Vázquez-Semadeni, E., Gómez, G. C., Jappsen, A. K., Ballesteros-Paredes, J., Klessen, R. S. 2009. High- and Low-Mass Star-Forming Regions from Hierarchical Gravitational Fragmentation. High Local Star Formation Rates with Low Global Efficiencies. 2009ApJ...707.1023V.
- [163] Vázquez-Semadeni E., 2010. Molecular Cloud Evolution. arXiv:1009.3962v1
- [164] Vázquez-Semadeni E., Colín P., Gómez G., Ballesteros-Paredes, J. & Watson A. W. 2010. Molecular cloud evolution III. Accretion vs. stellar feedback.
- [165] Wang, Jie, Navarro, J. F., Frenk, C. S., White, S. D. M., Springel, V., Jenkins, A., Helmi, A., Ludlow, A. & Vogelsberger, M. 2010. Assembly History and Structure of Galactic Cold Dark Matter Halos. 2010arXiv1008.5114W.
- [166] Warren, M. S. & Salmon, J. K. 1993. A parallel hashed Oct-Tree N-body algorithm. doi:10.1145/169627.169640.

- 
- [167] Warren, M. S. & Salmon, J. K. 1995. A portable parallel particle program. 1995CoPhC..87..266W.
- [168] Williams, J. P. & McKee, C. F. 1997. The Galactic Distribution of OB Associations in Molecular Clouds. 1997ApJ...476..166W.
- [169] Williams J. P., Blitz L. & McKee C.F. 2000. The structure and evolution of molecular clouds: from clumps to cores to the IMF. 2000prpl.conf...97W.
- [170] Withworth, A. 1979. The erosion and dispersal of massive molecular clouds by young stars. 1979MNRAS.186...59W.
- [171] Wolfire, M. G., Hollenbach, D., McKee, C. F., Tielens, A. G. G. M., Bakes, E. L. O. 1995. The neutral atomic phases of the interstellar medium. 1995ApJ...443..152W.
- [172] Xu, G. 1995. A New Parallel N-Body Gravity Solver: TPM. 1995ApJS...98..355X.
- [173] Yorke H. W. 1999. The dynamical evolution of HII regions - recent theoretical developments. 1986ARA&A..24...49Y
- [174] Zuckerman, B. & Evans, N. J. II. 1974. Models of massive molecular clouds. 1974ApJ...192L.149Z.
- [175] Zuckerman, B. & Palmer, P. 1974. Radio radiation from interstellar molecules. 1974ARA&A..12..279Z.