

RD-4 RTIGE-OPM-YB-04.04 Milieu interstellaire

Yannick Boissel

22 mai 2012

# Table des matières

<b>I</b>	<b>Introduction</b>	<b>2</b>
0.1	Ajouts depuis la version 3 . . . . .	3
0.2	Introduction . . . . .	3
<b>II</b>	<b>Contenu de l'archive</b>	<b>4</b>
0.3	Fichiers de contenu . . . . .	5
0.3.1	Dossier Modele_galaxie . . . . .	5
0.3.2	Dossier HR . . . . .	6
0.3.3	Racine . . . . .	7
<b>III</b>	<b>Méthode</b>	<b>8</b>
0.4	Introduction . . . . .	9
<b>1</b>	<b>Simulation des nuages</b>	<b>10</b>
1.1	Quelques trucs à savoir sur les fractales . . . . .	10
1.1.1	Dimension fractale . . . . .	10
1.1.2	Taille des clumps . . . . .	10
1.2	Méthode pour simuler des nuages . . . . .	11
1.2.1	Algorithme . . . . .	11
1.2.2	Explication (avec N=5) : . . . . .	13
1.2.3	Résultat . . . . .	13
1.2.4	Simuler une masse M de nuages . . . . .	14
1.2.5	Représentation des nuages . . . . .	15
<b>2</b>	<b>Formation stellaire</b>	<b>16</b>
2.1	Comment représenter ces particules hybrides ? . . . . .	16
2.1.1	Méthode . . . . .	16
2.1.2	Autres âges de modèle . . . . .	18
2.2	Lien avec les amas ouverts . . . . .	18
2.2.1	Comment ça marche . . . . .	18
2.2.2	Méthode . . . . .	18
2.3	Interactions nuages-étoiles . . . . .	19
2.3.1	Extinction . . . . .	19
2.3.2	Zones de formation stellaire . . . . .	19
<b>IV</b>	<b>Journal de bord</b>	<b>20</b>

Première partie

Introduction

## 0.1 Ajouts depuis la version 3

Nouveau modèle dynamique de la librairie GaLMeR, avec 30 millions de particules, et 1.21 Gyr d'évolution (contre 80000 particules et 500Myr d'évolution pour le précédent)

## 0.2 Introduction

Ce document donne la méthode permettant de simuler la matière du milieu interstellaire, son effet sur la luminosité des étoiles ; les amas ouverts jeunes (étoiles indiscernables) et la formation stellaire. Dans l'archive, outre ce document, il y a deux dossiers :

- 'Modele\_galaxie' : il contient les fichiers du modèle de galaxie (particules hybrides gaz/étoiles seulement) et le catalogue d'amas ouverts.
- 'HR' : il contient de quoi simuler les paramètres stellaires pour des particules ou des étoiles (particules éclatées).

Tous les fichiers sont en ASCII.

Deuxième partie

Contenu de l'archive

## 0.3 Fichiers de contenu

### 0.3.1 Dossier Modele\_galaxie

Ce dossier contient 50 fichiers de modèle de galaxie et un fichier contenant une carte d'âges d'étoiles. Ils donnent les positions et paramètres de 10240000 particules hybrides gaz/étoiles. Il y a 25 fichiers de chaque type (numérotés 01 à 25) correspondant à 25 âges de la Galaxie, de 11.79 à 13 milliards d'années, par pas de 50 millions d'années. Le fichier 01 a un âge de 11.79 Gyr si on considère l'instant présent comme étant le modèle 25, qui lui a alors un âge de 13 Gyr. En prenant pour origine le début de la simulation des modèles, le fichier 01 a un âge de 10 Myr, le fichier 25 a un âge de 1.21 Gyr.

– **'RTIGE\_2012\_GALMER\_gas\_stars\_out##\_dat' :**

Dans ces fichiers on trouve un ensemble de particules, toutes de masse totale identique (Elle est donnée dans les fichiers), avec comme données :

0. position x en kpc (par rapport au centre galactique)
1. position y en kpc (par rapport au centre galactique)
2. position z en kpc (par rapport au centre galactique)
3. composante vx de la vitesse en 100 km/s
4. composante vy de la vitesse en 100 km/s
5. composante vz de la vitesse en 100 km/s
6. masse de la particule en  $2.3 \times 10^9 M_{\odot}$
7. masse totale de gaz en  $2.3 \times 10^9 M_{\odot}$
8. métallicité moyenne du gaz de la particule
9. densité locale de la particule en  $2.3 \times 10^9 M_{sun}/kpc^3$
10. kernel radius en kpc

Le rang des particules dans les fichiers est toujours le même, même si les particules se déplacent dans la Galaxie.

– **'RTIGE\_2012\_GALMER\_gas\_stars\_params.dat' :**

Ce fichier contient un cube de données ( $10240000 \times 25 \times 2$ ), en ASCII, précédé d'un header décrivant le fichier. Les tranches de cube sont écrites sous forme matricielle, et sont séparées par un #. Il contient les paramètres des étoiles de chaque classe d'âge contenues dans les 10240000 particules. Il y a 25 classes d'âge, de 01 à 25. Elles correspondent aux étoiles nées aux âges de simulation correspondants (les âges des fichiers de modèle de galaxie). Pour chaque tranche, l'origine est en haut à gauche, et les abscisses sont écrites verticalement de haut en bas, les ordonnées horizontalement de gauche à droite (les tableaux sont tournés de  $90^\circ$ ).

En abscisse, on trouve la classe d'âge des étoiles de la particule, de 01 à 25. Exemple : si on travaille avec le modèle 07, il faudra lire les abscisses correspondant aux classes d'âge 01 à 07. Et ces classes d'âges seront associées à des âges partant de 8.48 pour la classe 01 et 6.6 pour la classe 07 (en log10). Voir la section **Formation stellaire**. En ordonnées on a le numéro du paramètre considéré. Il y en a trois, dans l'ordre :

0. Métallicité Z des étoiles de la classe d'âge
1. Masse totale d'étoiles de la classe d'âge

Les valeurs des cases correspondent aux valeurs de ces 2 paramètres en fonction de la classe d'âge. La troisième dimension du cube est l'indice de la particule, c'est-à-dire, le rang dans les fichiers 'RTIGE\_2012\_GALMER\_gas\_stars\_out##\_dat'.

**Métallicités :**

J'ai discrétisé les valeurs de métallicité qui étaient données par le modèle de galaxie afin de

l'alléger. J'en ai choisi 6 : 0.005, 0.01, 0.015, 0.02, 0.025, 0.03. Pour chaque valeur, j'ai regardé quelles sont les particules qui ont une métallicité à  $\pm 0.0025$ , et je leur ai donné cette valeur. Cependant, la première tranche va de 0 à 0.0075, et la dernière de 0.0275 à 0.04. Voir Journal de bord au 15 novembre 2011.

– **'RTIGE\_2012\_OC\_catalogue.dat'** :

Ce fichier contient un cube de données ( $10240000 \times 25 \times 5$ ), en ASCII, précédé d'un header décrivant le fichier. Les tranches de cubes sont écrites sous forme matricielle, et sont séparées par un  $\#$ . Il contient les nombres d'amas ouverts de chaque classe d'âge contenus dans les particules. Il y a 25 classes d'âge, de 01 à 25. Elles correspondent aux étoiles nées aux âges de simulation correspondants (les âges des fichiers de modèle de galaxie). Pour chaque tranche, l'origine est en haut à gauche, et les abscisses sont écrites verticalement de haut en bas, les ordonnées horizontalement de gauche à droite (les tableaux sont tournés de  $90^\circ$ ).

En abscisse, on trouve la classe d'âge des étoiles de la particule, de 01 à 25. Elles représentent les groupes d'étoiles nés dans les modèles antérieurs au modèle choisi. Exemple : si on travaille avec le modèle 07, il faudra lire les abscisses correspondant aux classes d'âge 01 à 07. Et ces classes d'âges seront associées à des âges partant de 8.48 pour la classe 01 et 6.6 pour la classe 07 (en  $\log_{10}$ ). Voir la sous-section **Lien avec les amas ouverts**. En ordonnées il y a :

- 0. Nombre d'amas de masse  $10^2 M_\odot$
- 1. Nombre d'amas de masse  $10^3 M_\odot$
- 2. Nombre d'amas de masse  $10^4 M_\odot$
- 3. Nombre d'amas de masse  $10^5 M_\odot$
- 4. Nombre d'amas de masse  $10^6 M_\odot$

### 0.3.2 Dossier HR

Dans le dossier 'HR', on trouve pour 6 métallicités ( $z$  dans le nom des fichiers), 4 fichiers différents :

- **'RTIGE\_2011\_HRcube\_z#.dat'** : Ce fichier est constitué de 354 tableaux  $100 \times 100$  au format entier 16 bits. Les tableaux sont écrits à la file, séparés par un  $\#$ . Chaque tableau est un diagramme H-R calculé pour un âge donné : **il ne contient qu'une seule isochrone. Comme les particules contiennent des étoiles d'âges différents, il va falloir combiner différentes isochrones pour représenter les particules ou pour éclater des particules.**

En abscisses on a le  $\log_{10}$  de la température, en  $\log_{10}(K)$ , croissant de haut en bas, de 3.3 à 4.7, et en ordonnées, on a le  $\log_{10}$  de la luminosité en bande V, en  $\log_{10}(L_\odot)$ , croissant de gauche à droite, de -4.0 à 6.5.

Chaque case donne le  $\log_{10}$  de la proportion d'étoiles sur 1 million, fois 1000, ayant ce couple ( $\log_{10}(L_V), \log_{10}(T_e)$ ). Ces valeurs sont donc comprises entre 0 et 6000. Si les diagrammes HR sont donnés avec des nombres relatifs d'étoiles dans une unité si étrange, c'est parce que cela m'a permis de les enregistrer au format entier 16 bits, en perdant peu d'information, produisant ainsi des fichiers les plus légers possible.

Les 354 tableaux correspondent à 354 âges, espacés de 0.01 dans l'unité  $\log_{10}(\text{annees})$ , la même unité que pour les données du modèle de galaxie.

- **'RTIGE\_2011\_star\_params\_z#.dat'** : Ce fichier est un cube de données constitué de 15 tableaux  $100 \times 100$  au format float 64 bits. Les tableaux sont écrits à la file, séparés par un  $\#$ . Il s'agit de cartes de paramètres stellaires, correspondant aux diagrammes HR. Ainsi, pour chaque case de diagramme HR, on dispose de 15 paramètres stellaires en plus de la température et de la luminosité. En ordonnées on a le  $\log_{10}$  de la température, en  $\log_{10}(K)$ , croissant de haut en bas, de 3.3 à 4.7, et en abscisse, on a le  $\log_{10}$  de la luminosité en bande V, en  $\log_{10}(L_\odot)$ , croissant de gauche à droite, de -4.0 à 6.5.

Ces paramètres stellaires sont dans l'ordre :

0.  $\log_{10}(\text{Luminosité bolométrique})$  en  $\log_{10}(L_{\odot})$
  1.  $\log(\text{gravité})$  en  $\log_{10}(cm.s^{-2})$
  2. masse stellaire en  $M_{\odot}$
  3. rayon stellaire en  $R_{\odot}$
  4. magnitude absolue en bande  $U$
  5. magnitude absolue en bande  $B$
  6. magnitude absolue en bande  $V$
  7. magnitude absolue en bande  $R$
  8. magnitude absolue en bande  $I$
  9. magnitude absolue en bande  $J$
  10. magnitude absolue en bande  $H$
  11. magnitude absolue en bande  $K$
  12. couleur normalisée : rouge
  13. couleur normalisée : vert
  14. couleur normalisée : bleu
- **'RTIGE\_2011\_lcmmap\_z#.dat'** : Ce fichier contient les classes de luminosité des étoiles situées dans les cases de diagramme HR. Il y a 100x100 chaînes de caractères, le tableau est orienté dans le même sens que les diagrammes HR et les cartes de paramètres stellaires. Attention, suite à une erreur de code, les lignes sont pour l'instant encadrées de crochets, au lieu d'avoir un tableau écrit sous forme d'une matrice.
- **'RTIGE\_2011\_spcmap\_z#.dat'** : Ce fichier contient les classes spectrales des étoiles situées dans les cases de diagramme HR. Il y a 100x100 chaînes de caractères, le tableau est orienté dans le même sens que les diagrammes HR et les cartes de paramètres stellaires. Attention, suite à une erreur de code, les lignes sont pour l'instant encadrées de crochets, au lieu d'avoir un tableau écrit sous forme d'une matrice.

### 0.3.3 Racine

**'frac.py'** : J'ai mis également un programme python contenant une fonction qui fabrique des nuages fractals. Pour lancer ce programme, il faut les librairies numpy et scipy de python que l'on trouve facilement sur Internet. Pour lancer la fonction dans python :

```
>>>import os
>>>os.chdir('/dossier_contenant_frac.py/')
>>>import frac
>>>frac.fractalcloud(D,N,Lmax,imsize,mapsize,filename)
```

Avec :

D : dimension fractale

N : nombre d'éléments par niveaux

Lmax : nombre de niveaux

imsize : taille en pixels de l'image finale

imsize : taille en unités de distance de l'image finale (zoom). La taille du nuage au niveau 0 est de 100 unités.

filename : chemin/nom du fichier png de l'image finale



## Troisième partie

### Méthode

## 0.4 Introduction

Je distingue 3 problèmes à résoudre :

- La simulation des nuages (masse, forme, taille, composition)
- La formation stellaire : lien entre nuages et amas ouverts
- L'interaction lumineuse nuages-étoiles : extinction, réflexions, ionisations

# Chapitre 1

## Simulation des nuages

Pfenniger & Combes 1994 ont montré que les nuages moléculaires sont sous formes de fractales, dont les plus petits niveaux sont constitués de "clumpuscles" de  $10^{-3} M_{\odot}$ , de taille de l'ordre de 30 UA. Ces clumpuscles sont le dernier stade où on les considère encore comme du gaz avant la naissance de l'étoile.

### 1.1 Quelques trucs à savoir sur les fractales

#### 1.1.1 Dimension fractale

Soit  $M$  la masse de gaz contenue dans une sphère de rayon  $r$ . Comme le nuage est fractal, la masse obéit à la loi d'auto-similarité :

$$\frac{M}{M_0} = \left( \frac{r}{r_0} \right)^D \quad (1.1)$$

$M_0$  et  $r_0$  sont la masse et l'échelle spatiale de référence, et  $D$  est la dimension fractale de la distribution de masse (Mandelbrot 1982)

Donc :

$$D = \frac{\log_{10}(M/M_0)}{\log_{10}(r/r_0)} \quad (1.2)$$

Cette définition signifie que si on connaît la masse du nuage pour deux échelles spatiales, on a la dimension fractale. Solomon et al 1987 donne une densité moyenne de surface de  $170 M_{\odot} \cdot \text{pc}^{-2}$  et les auteurs en déduisent que la masse typique d'un nuage de 30 pc de diamètre est de  $5 \times 10^5 M_{\odot}$ , et que si on regarde la plus petite échelle 30 UA, les clumpuscles, on a une masse de  $0.001 M_{\odot}$ . Alors la dimension fractale d'un nuage est en moyenne de 1.64.

#### 1.1.2 Taille des clumps

Soit  $l = 0$ , le niveau de départ de la fractale (le nuage dans son ensemble). Soit  $N$  le nombre d'éléments par niveaux (le nombre de nuages secondaires à chaque niveau). Soit  $M_{l=l_{max}}$  la masse des clumps au niveau le plus avancé,  $l = l_{max}$ . On a :

$$M_l = N M_{l+1} = N^l M_{l=l_{max}} \quad (1.3)$$

On a aussi :

$$M_{tot} = N^{l_{max}} M_{l=l_{max}} \quad (1.4)$$

Avec l'équation (2.10) on a donc :

$$r_l^D = N r_{l+1}^D \quad (1.5)$$

On définit le facteur d'échelle  $\alpha$  comme :

$$\alpha = \frac{r_{l+1}}{r_l} = N^{-1/D} \quad (1.6)$$

$\alpha$  donne le rapport entre les tailles caractéristiques du nuage, donc les tailles des clumps entre deux niveaux de fractale. Il dépend de  $N$  et  $D$ . La dimension fractale influe donc sur la surface couverte par le nuage pour une masse de gaz donnée. Plus  $D$  est grande, plus le nuage couvre de surface (donc il est moins dense). Inversement, plus  $D$  est petite, plus le nuage se concentre en petits globules denses.

## 1.2 Méthode pour simuler des nuages

### 1.2.1 Algorithme

Le nombre de niveaux et le nombre d'éléments par niveaux donnent le nombre de particules à afficher qui est  $N^{l_{max}}$ . La limitation est informatique. On ne pourra pas simuler un nuage jusqu'aux clumpuscles de 30 UA, vu de très loin. Il faudra donc décider de  $N$  et  $l_{max}$  selon la distance à laquelle l'observateur se trouve par rapport au nuage, selon ce qui est visible avec la résolution de l'écran et de la grille de voxels, et selon la puissance de calcul utilisable. Le niveau de base est  $l = 0$  (l'ensemble des nuages), le niveau contenant les plus petits nuages est  $l_{max}$ .

Voici l'algorithme de construction des nuages fractals :

```
# Préparatifs :
 $\alpha = N^{-1/D}$ 
# Taille des nuages à chaque niveau :
 $r = \text{array}(l_{max})$ 
 $r[0] = 100$ . # taille des  $N$  nuages constituant le nuage du niveau 0
 $l = 1$ 
while  $l < l_{max}$  :
     $r[l] = r[l - 1] \times \alpha$ 
     $l += 1$ 

# Algorithme :
# A la première itération, les nuages sont positionnés autour d'une position  $(x_{old}, y_{old}, z_{old})$  prise comme
# origine : Le nuage représentant l'ensemble des nuages a une position (0,0,0) (c'est le niveau 0)
 $x_{old} = \text{array}(N)$ 
 $y_{old} = \text{array}(N)$ 
 $z_{old} = \text{array}(N)$ 
 $l = 1$ 
while  $l \leq l_{max}$  : # boucle sur les niveaux de fractale
     $x = \text{array}(N^l)$ 
     $y = \text{array}(N^l)$ 
     $z = \text{array}(N^l)$ 
    # Taille des nuages au niveau  $l - 1$ 
     $r_{l-1} = r[l - 1]$ 
    # Pour chaque élément  $n_{l-1}$  du niveau  $l - 1$ , il y aura  $N$  nouveaux éléments au niveau  $l$ .
    # Donc au total, au niveau  $l$ , il y aura  $N * N^{l-1} = N^l$  nuages de taille  $r_l$ 
     $N_l = N^{l-1}$ 
     $n = 0$ 
    while  $n < N$  : # boucle sur le nombre d'éléments par élément  $n_{l-1}$ 
         $n_l = 0$ 
        while  $n_l < N_l$  : # boucle sur les éléments du niveau  $l$ 
            # Tir aléatoire de coordonnées de nuages dans une sphère de rayon  $r_{l-1}$ 
            # autour des éléments du niveau  $l - 1$  de coordonnées  $(x_{old}, y_{old}, z_{old})$  selon une loi uniforme.
            # Le nuage de niveau  $l - 1$  a une taille  $r_{l-1}$  et est constitué de nuages de niveau  $l$ 
             $\rho = \text{random}[0 : r_{l-1}]$ 
             $\phi = \text{random}[0 : 2\pi]$ 
             $S = \text{random}[-1 : 1]$ 
             $C = \rho \times \sqrt{1 - S^2}$ 
             $x[n \times N_l + n_l] = C \times \cos(\phi) + x_{old}[n_l]$ 
             $y[n \times N_l + n_l] = C \times \sin(\phi) + y_{old}[n_l]$ 
             $z[n \times N_l + n_l] = \rho \times S + z_{old}[n_l]$ 
```

$$\begin{aligned}
n_l+ &= 1 \\
n+ &= 1 \\
x_{old} &= x \\
y_{old} &= y \\
z_{old} &= z \\
l+ &= 1
\end{aligned}$$

### 1.2.2 Explication (avec N=5) :

- 1) On prend un nuage, de rayon  $r_0$ , centré sur une position ( $x_{l=0} = 0, y_{l=0} = 0, z_{l=0} = 0$ ).
- 2) On le décompose en 5 nuages de rayon  $r_1 < r_0$ , centrés sur des positions tirées aléatoirement ( $x_{l=1_1} = x_1 + x_{l=0}, x_{l=1_2} = x_2 + x_{l=0} \dots$ ).
- 3) Dans chacun de ces 5 nuages, on introduit un nuage de rayon  $r_2 < r_1$ , centré sur une position tirée aléatoirement : ( $x_{l=2_{1_1}} = x_1 + x_{l=1_1}, x_{l=2_{2_1}} = x_1 + x_{l=1_2}, \dots$ )
- 4) On effectue 5 fois l'étape 3) pour avoir 5 nuages de niveau  $l = 2$  par nuage de niveau  $l = 1$ .  
On a donc 25 nuages de niveau  $l = 2$ .
- 5) Dans chacun de ces 25 nuages, on introduit un nuage de rayon  $r_3 < r_2$ , centré sur une position tirée aléatoirement : ( $x_{l=3_{1_{1_1}}} = x_1 + x_{l=2_{1_1}}, x_{l=3_{1_{2_1}}} = x_1 + x_{l=2_{1_2}}, \dots$ )
- 6) On effectue 5 fois l'étape 5) pour avoir 5 nuages de niveau  $l = 3$  par nuage de niveau  $l = 2$ .  
On a donc 125 nuages de niveau  $l = 3$ .
- 7) etc ...

### 1.2.3 Résultat

Sur les *figures* 1.1 et 1.2 je montre le résultat obtenu pour  $D=3, N=5, L_{\max}=8$ , et  $D=1.64, N=5, L_{\max}=8$  respectivement. Le calcul prend une trentaine de secondes sur un macbook pro 2.4Ghz Dual Core, disque SSD, 5 Go de Ram. Je fournis dans l'archive la fonction python que j'ai écrite après traduction du code fortran de Françoise combes.

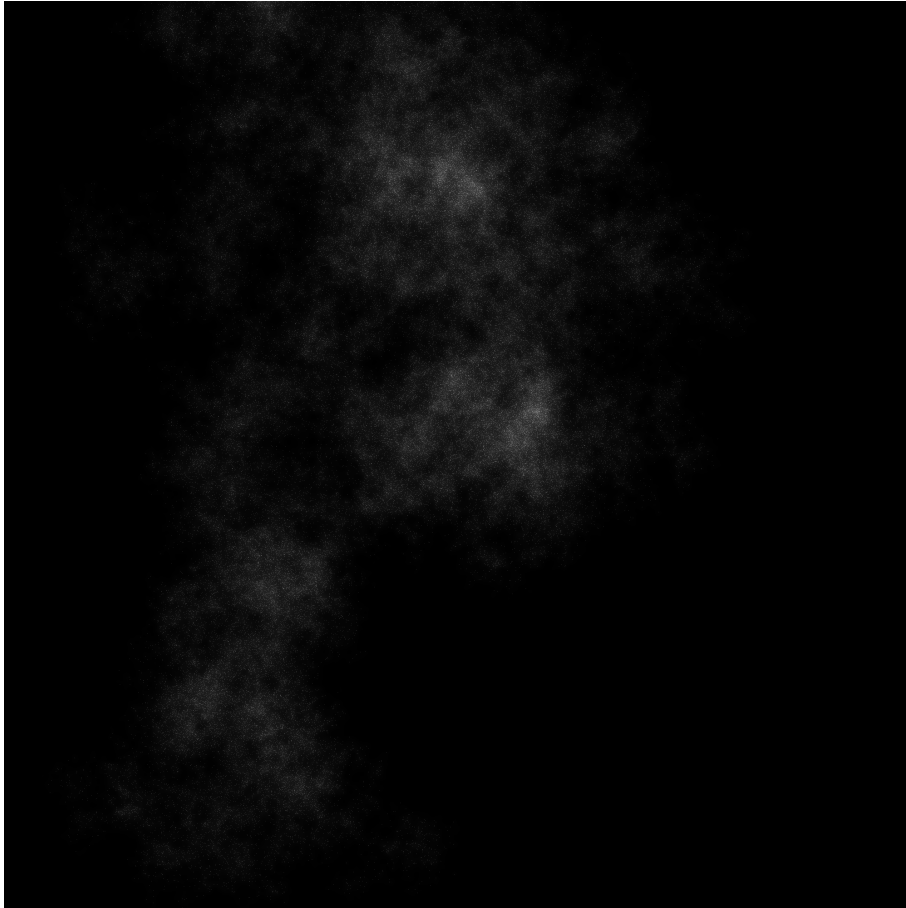


FIGURE 1.1 –



FIGURE 1.2 –

#### 1.2.4 Simuler une masse $M$ de nuages

Les nuages sont composés de gaz moléculaire et de poussière (régions HII), et baignent dans un gaz atomique (régions HI), de plus en plus ionisé quand la distance au nuage moléculaire augmente. Le gaz est ionisé par le rayonnement UV des étoiles et les rayons cosmiques. La limite entre le gaz ionisé et le gaz atomique correspond à la profondeur de nuage à partir de laquelle les UV ne peuvent plus casser les atomes. La limite entre le gaz atomique et le gaz moléculaire correspond à la profondeur de nuage à partir de laquelle les UV ne parviennent plus à casser les molécules. Les nuages moléculaires sont plus ou moins opaques en raison de l'existence de poussière : grosses molécules et grains solides. C'est la poussière qui est responsable de l'extinction.

Voilà des informations sur la composition du milieu interstellaire :

milieu	$v$	$m$	$\Delta$ (pc)
ionisé chaud	0.4-0.7	0.00047	32-324
ionisé moyen	0.15-0.4	0.00792	32-324
atomique moyen	0.2-0.6	0.03166	32-324
atomique froid	0.01-0.04	0.47498	32-324
moléculaire	0.01	0.47498	0.32-32
poussière	0.01	0.001	0.32-32

$v$  est la fraction du volume total du milieu interstellaire occupée par ce type de milieu. Cela peut se dire comme : soit  $V$  un volume quelconque d'espace interstellaire. Il contient  $v\%$  de tel ou tel ingrédient.  $m$  est la fraction de masse, par rapport à  $1 M_{\odot}$  correspondant à chaque ingrédient.  $\Delta$  est la taille caractéristique de chaque phase.

Quand on va vouloir simuler le contenu d'une particule de gaz du modèle de Fux, on peut donc considérer que la moitié de la masse est sous forme d'un nuage moléculaire

fractal, et que 0.1% de la masse de gaz moléculaire est sous forme de poussière. On ne représentera le gaz atomique qu'en présence d'étoiles, car il émet de la lumière lorsqu'il est ionisé par la lumière des étoiles.

### 1.2.5 Représentation des nuages

Une fois qu'on a la masse de gaz moléculaire  $M_{tot}$  (une particule pèse  $3.3 \times 10^4 M_{\odot}$ , mais on peut modéliser une masse différente selon qu'on considère une particule, plusieurs particules ou une fraction d'une particule, ce qui dépend de la distance de l'observateur) à modéliser et qu'on a défini des paramètres  $N$ ,  $l_{max}$  et  $D$ , il faut représenter le nuage non pas sous forme des points que l'on a calculés mais sous forme de voxels. Le niveau de fractale le plus détaillé est le seul niveau à afficher dans la simulation. La superposition d'un grand nombre de petits nuages donne les changements de densité du nuage dans son ensemble. On connaît la taille de ces petits nuages,  $r_{l=l_{max}}$ , elle est calculée avant la boucle sur les niveaux. On connaît également la masse  $M_{l=l_{max}}$  de ces petits nuages grâce à l'équation (1.4). Il suffit donc de diluer cette masse dans des sphères de rayon  $r_{l=l_{max}}$ , et de les représenter sous forme de voxels. De fait, les niveaux plus fins et invisibles sont moyennés dans les nuages du niveau  $l_{max}$ . Dans les images que je donne dans la partie *Résultats*, j'ai lissé le dernier niveau et superposé le niveau précédent car je n'ai pas accès à un code fabriquant des voxels, mais ce n'est pas réaliste.

On peut considérer que la masse de poussière est répartie de façon homogène dans ces nuages.



# Chapitre 2

## Formation stellaire

### 2.1 Comment représenter ces particules hybrides ?

Pour chaque particule du modèle de galaxie, on a la masse de gaz, la masse et le nombre d'étoiles  $N_*$  de 25 tranches d'âge différentes et leur métallicité. Ce nombre total d'étoiles est en moyenne ce que doit contenir une particule de cette masse totale d'étoiles. Il est calculé comme étant le rapport entre la masse totale d'étoiles  $M_{tot}$  de la tranche d'âge considérée et la masse moyenne de ces étoiles  $M_{moy}$ . Pour calculer le nombre total d'étoiles, la masse totale d'étoiles est donnée pour chaque particule et chaque tranche d'âge dans les fichiers RTIGE\_2012\_GALMER\_gas\_stars\_params.dat'.

$$N_* = \frac{M_{tot}}{M_{moy}} \quad (2.1)$$

Cette masse moyenne est calculée à partir de l'isochrone  $Iso_i$  correspondant à cette tranche d'âge  $i$  et de la carte de masse, pour la métallicité  $Z_{Iso_i}$  de ces étoiles.

$$M_{moy} = \left[ \sum_{j=0}^{nb_c^*} \frac{10^{n_j/1000}}{10^6} \times m_j \right] \quad (2.2)$$

$nb_c^*$  est le nombre de cases non vides du diagramme isochrone  $Iso_i$ .  $n_j$  est la valeur de la case  $j$  du diagramme isochrone  $Iso_i$  non normalisé.  $m_j$  est la valeur de la case  $j$  de la carte de masse de métallicité  $Z_{Iso_i}$ .

#### 2.1.1 Méthode

Je vais décrire la méthode dans le cas où l'on veut simuler la Galaxie à l'âge 25, soit 13 Gyr, l'âge actuel de la Galaxie. Soit donc le présent,  $t=0$ , correspondant au modèle 25. Les âges données par la suite le seront par rapport à cette référence.

#### Comment utiliser les fichiers du modèle de Galaxie

Pour le modèle d'âge 00 (rapporté au début de la simulation, non disponible dans l'archive), soit contenant les étoiles âgées de 1.21 Gyr, toutes les particules ne contiennent que du gaz. Pour le modèle 01, âgé de 1.20 Gyr, on trouve des particules ne contenant que du gaz, et des particules contenant du gaz et des étoiles âgées de 1.20 Gyr par rapport à l'âge actuel de la Galaxie (âgées de 10 Myr rapporté au début de la simulation). Dans le modèle 02, on a des particules ne contenant que du gaz, des particules contenant du gaz et des étoiles âgées de 1.20 Gyr, et des particules contenant du gaz et des étoiles âgées de 1.15 Gyr (âgées de 10+50 Myr rapporté au début de la simulation), et ainsi de suite pour les autres modèles.

Tout d'abord il faut séparer la masse de gaz et la masse d'étoiles et les simuler séparément. Comme on a des âges d'étoiles différents dans les particules il va falloir utiliser plusieurs diagrammes HR différents (contenant chacun une isochrone, soit un âge). Appelons ces diagrammes HR ne contenant qu'un âge des diagrammes isochrones. Pour ça on a besoin de quelques paramètres qui figurent dans les fichiers 'RTIGE\_2012\_GALMER\_gas\_stars\_out##.dat' et 'RTIGE\_2012\_GALMER\_gas\_stars\_params.dat'.

### Position des particules

Elles sont données dans les fichiers de type 'RTIGE\_2012\_GALMER\_gas\_stars\_out##.dat'. Les particules sont toujours rangées dans le même ordre quelque soit l'âge du modèle : les particules changent de position dans la Galaxie avec le temps.

### Paramètres des classes d'âge

Pour chaque particule, on a besoin du nombre d'étoiles par classe d'âge, et de la métallicité des étoiles de chaque classe :

1) Lire "l'image" correspondant à l'indice de la particule considérée dans le cube du fichier 'RTIGE\_2012\_GALMER\_gas\_stars\_params.dat'. Les particules y sont rangées dans le même ordre que dans les fichiers donnant les positions.

2) Les classes d'âges sont données en abscisses, de 01 à 25. Comme on travaille avec le modèle 25, les âges correspondant vont de 1.20 Gyr à 0 yr, soit en log10, dans l'ordre : 9.08,9.06,9.04,9.02,9.00,8.98,8.95,8.93,8.90,8.60. (6.60 est l'âge le plus jeune possible dans les cubes de diagrammes isochrones). Pour chaque classe d'âge on a en ordonnée, dans l'ordre : la métallicité  $Z$  des étoiles, la masse totale d'étoiles (on en aura besoin pour éclater les particules), le nombre total d'étoiles (calculé à partir de cette masse totale).

### Gestion des classes d'âge

Pour chaque particule on a  $N_i$  le nombre d'étoiles à simuler dans la tranche d'âge  $i$ , et la métallicité  $Z$  de ces étoiles. Il faut en fait considérer une particule comme un ensemble de 25 particules, dont on va calculer les luminosités dans les trois couleurs, séparément, puis que l'on va sommer ensuite pour avoir les luminosités totales de la particule.

Soit  $Iso_i$  les histogrammes contenant les isochrones d'âge  $i$ , pris dans les cubes de données 'RTIGE\_2011\_HRcube\_z#.dat' de métallicité  $Z_{Iso_i}$ , et soit  $n_{ages}$  le nombre maximum d'âges à considérer pour la particule choisie. Comme on travaille pour l'instant avec le modèle 25,  $n_{ages} = 25$ .

Avant toute chose il nous faut normaliser ces histogrammes à 1. Ils sont calculés à partir d'un ensemble de  $10^6$  étoiles, et les proportions d'étoiles sont données en log10, puis multipliées par 1000. On a donc :

$$Iso_{i_{norm}} = \frac{10^{Iso_i/1000}}{10^6} \quad (2.3)$$

Dans le cas d'étoiles indiscernables il ne reste plus qu'à calculer la luminosité totale de la particule pour les 3 couleurs, à partir de l'équation que je rappelle ici pour le canal rouge par exemple (la normalisation est déjà incluse dans l'équation) :

$$LV_{r_{tot_i}} = \left[ \sum_{j=0}^{nb_c^*} \frac{10^{n_j/1000}}{10^6} \times r_j \times 10^{x_j} \right] \times N_i \quad (2.4)$$

$nb_c^*$  est le nombre de cases non vides du diagramme isochrone  $Iso_i$ .  $n_j$  est la valeur de la case  $j$  du diagramme isochrone  $Iso_i$  non normalisé.  $r_j$  est la valeur de la case  $j$  de la carte de couleur rouge de métallicité  $Z_{Iso_i}$ .  $x_j$  est l'abscisse de la case  $j$ ,  $x_j = \log_{10}(LV_j)$ .  $N_i$  est le nombre total d'étoiles dans la classe d'âge  $i$ .

Enfin la luminosité totale de la particule, pour la couleur rouge est :

$$LV_{r_{tot}} = \sum_{i=0}^{n_{ages}} LV_{r_{tot_i}} \quad (2.5)$$

Dans le cas d'étoiles discernables, on éclate la particule en tirant des étoiles selon les  $n_{ages}$  diagrammes  $Iso_i$  et les cartes de masse associées (de même métallicité  $Z_{Iso_i}$  que les diagrammes  $Iso_i$ ) jusqu'à atteindre la masse en étoiles de chaque classe d'âge  $i$  de la particule.

### 2.1.2 Autres âges de modèle

Si maintenant on veut simuler une galaxie plus jeune, il faut prendre comme référence un fichier de modèle plus jeune. Prenons par exemple le modèle 07. Dans les fichiers 'RTIGE\_2012\_GALMER\_gas\_stars\_params.d' et 'RTIGE\_2012\_OC\_catalogue.dat' il ne faut lire que les cases d'abscisses  $\leq 07$ . Pour calculer les luminosités d'une particule, il faut prendre  $n_{ages} = 7$ . Les âges des classes d'âge seront alors (de 01 à 07) : 8.48, 8.40, 8.30, 8.18, 8.00, 7.70, 6.60 (en  $\log_{10}(\text{yr})$ ). Pour les autres parties de la galaxie, prendre les fichiers 'stars\_disc\_out07' et 'stars\_bulb\_out07', en suivant les instructions du document "Étoiles du champ".

## 2.2 Lien avec les amas ouverts

### 2.2.1 Comment ça marche

Comme on connaît la masse totale d'étoiles nouvelles à chaque modèle, autrement dit, la masse totale d'étoiles de chaque classe d'âge pour tous les modèles, on peut avoir le nombre d'amas ouverts que ça représente, connaissant la fonction de masse initiale des amas ouverts (ICMF), c'est-à-dire une fonction qui donne la proportion d'amas de chaque classe de masse qui naissent à un moment donné. La proportion d'amas par classe de masse est :

$$\frac{dN}{dM} \propto M^\alpha \quad (2.6)$$

avec  $\alpha = -2$ , en prenant  $M \in [10^2, 10^3, 10^4, 10^5, 10^6] M_\odot$ . Pour chaque particule, j'ai donc tiré des amas ouverts en prenant une masse aléatoire entre  $10^2$  et  $10^6 M_\odot$ , en les classant au fur et à mesure dans ces 5 boîtes, jusqu'à atteindre la masse totale de chaque classe d'âge. J'ai ainsi obtenu pour chaque particule le nombre d'amas de chaque classe d'âge et de chaque classe de masse. Ces données sont contenues dans le fichier 'RTIGE\_2012\_OC\_catalogue.dat'.

### 2.2.2 Méthode

Il faut représenter les amas ouverts les plus massifs par des sprites, les autres sont dilués dans la luminosité moyenne de la particule. Comment obtenir les infos ?

1) Lire "l'image" du cube du fichier 'RTIGE\_2012\_OC\_catalogue.dat' correspondant à l'indice de la particule considérée. Les particules sont rangées dans le même ordre que dans les fichiers du modèle de galaxie.

2) Pour avoir le nombre d'amas dans une classe d'âge et une classe de masse donnée, lire la case d'abscisse classe d'âge et d'ordonnée classe de masse choisie. Les masses en ordonnées sont rangées par ordre croissant. Les classes d'âge sont rangées comme dans le fichier de paramètres.

Je ne sais pas à partir de quelle masse représenter les amas par des sprites. Il faut faire des tests. Essayons  $10^4 M_\odot$ . Pour les amas moins massifs, on ne les simulera que lorsqu'on devra éclater les particules, en se référant au document "Amas ouverts".

Pour calculer la luminosité d'un sprite dans les trois couleurs, il faut :

1) Lire la métallicité de l'amas dans la colonne correspondant à la classe d'âge de l'amas, pour la particule hôte, dans le fichier 'RTIGE\_2012\_GALMER\_gas\_stars\_params.dat'.

2) Lire l'histogramme isochrone correspondant dans le bon cube 'RTIGE\_2011\_HRcube\_z#.dat', ainsi que la carte de masse correspondante dans le fichier 'RTIGE\_2011\_star\_params\_z#.dat'.

3) Calculer la masse moyenne des étoiles de l'amas avec la carte de masse et le diagramme isochrone. Calculer le nombre total d'étoile en divisant la masse de l'amas par cette masse moyenne.

4) Calculer les luminosités.

5) Soustraire au nombre d'étoiles de la classe d'âge considérée de la particule le nombre d'étoiles contenues dans l'amas, avant de représenter le reste de la particule.

## 2.3 Interactions nuages-étoiles

### 2.3.1 Extinction

#### Bande V

L'extinction  $A_V$  en bande V est calculée en fonction de la colonne densité de Gaz et la métallicité  $z$  dans chaque cellule du volume, donc en fonction de la colonne densité de poussière qui est le produit des 2. Pour chaque cellule traversée, on calcule donc :

$$\Delta mag = \frac{M_{gaz} \times z}{(9z_{\odot} \times a^2)} \quad (2.7)$$

avec :

$M_{gaz}$  la masse de gaz dans la cellule en masses solaires

$z$  la métallicité de la cellule en masse

$z_{\odot}$  la métallicité solaire en masse

$a$  la longueur d'un coté de la cellule en parsec

Cette extinction est à retrancher aux magnitudes apparentes des sources simulées si du gaz (et donc de la poussière) se trouve sur la ligne de visée.

#### Autres bandes

L'article Rieke 1985 donne les extinctions relatives dans les différentes bandes, en magnitudes (voir *tableau 2.1*)

$\lambda$	$k_{\lambda} = A_{\lambda}/A_V$
U	1.531
B	1.324
R	0.748
I	0.482
J	0.282
H	0.175
K	0.112

TABLE 2.1 –

Afin de représenter la galaxie dans une autre bande  $\lambda$ , avec de fausses couleurs, il suffit de calculer la luminosité dans cette bande  $L_{\lambda}$  :

$$L_{\lambda} = 10^{0.4(4.74 - M_{\lambda})} \quad (2.8)$$

$M_{\lambda}$  étant la magnitude absolue de l'étoile ou de la particule en bande  $\lambda$  et 4.74 étant la magnitude absolue bolométrique du Soleil.  $L_{\lambda}$  est en  $L_{\odot}$ . Il faut ensuite utiliser de fausses couleurs ; puis retrancher l'extinction correspondante à la magnitude apparente :

$$m_{\lambda} = m_{\lambda_0} + k_{\lambda} \times A_V \quad (2.9)$$

L'indice 0 indique une magnitude apparente non augmentée par l'extinction (on utilise le mot dérogie habituellement).

### 2.3.2 Zones de formation stellaire

Il faut représenter le gaz avec une couleur rose lumineuse autour des sprites contenant les gros amas ouverts les plus jeunes (ceux qui ont un âge de 6.6, soit ceux nés dans le modèle utilisé). Discuter avec RSacosmos.

Quatrième partie

**Journal de bord**

**19 octobre 2011 :**

Je vois 5 choses à simuler/intégrer dans la simulation de la Galaxie :

- Différence gaz/poussière et différence milieu ionisé, HI et HII pour le gaz.
  - Simuler des bulles et inhomogénéités
  - Simuler l'extinction (et le rougissement)
  - Intégrer les cartes 3D du milieu interstellaire local (cube de Rosine)
1. gaz et poussière

Table 3.1 Phases of the interstellar medium. These phases may not be in equilibrium but dynamic and short-lived at any particular location.

Phase	Temperature K	Density $\text{cm}^{-3}$	Fraction of Volume	Mass in $10^9 M_\odot$
Hot ionised medium	$3-20 \times 10^5$	$3 \times 10^{-3}$	0.4-0.7	0.003
Warm ionised medium	10,000	$3 \times 10^{-1}$	0.15-0.4	0.05
Warm neutral medium	8000	$4 \times 10^{-1}$	0.2-0.6	0.2
Cold neutral medium	40-100	$6 \times 10^1$	0.01-0.04	3
Molecular Clouds	3-20	$3 \times 10^2$	0.01	3

FIGURE 2.1 –

Dans le livre "The Origin of stars" de Michael David Smith, je trouve une description du gaz (*figure 2.1*) et une description de la poussière. Les grains de poussière interstellaire sont constitués de carbone, de silicium, d'hydrogène, d'oxygène, de magnésium et de fer sous la forme de glaces, silicates, graphite, métaux et matière organique. La poussière représente 1% de la masse de la matière interstellaire. La taille des grains peut atteindre 1000 Å et ils contiennent de l'ordre de  $10^9$  atomes. D'autres grains sont sous la forme de très grosses molécules, les PAHs : polycyclic aromatic hydrocarbons contenant une centaine d'atomes. En majorité, les grains sont petits, 50 Å.

La poussière se forme par condensation depuis la phase gazeuse à des températures  $< 2000$  K. A chaque élément correspond une température de condensation, à laquelle 50% des atomes condensent quand ils sont en équilibre thermodynamique : 1200 à 1600 K pour les éléments lourds MG, Si, Fe, Al, Ca ..., et  $< 200$  K pour les éléments légers O, N, H, C. La poussière provient également des enveloppes que les géantes rouges éjectent. Dans les nuages moléculaires, les éléments légers condensent sur des grains déjà formés.

D'après ce que j'ai compris, dans le modèle de Fux, on a une densité de gaz et poussière mélangés, donnant l'allure générale dans le disque. Il faut donc mettre des grumeaux dans le disque de particules de gaz du modèle de Fux : des grumeaux denses de gaz moléculaire et poussière enfouis dans des nuages moins denses de gaz atomique, eux mêmes enfouis dans des nuages de gaz ionisé.

Dans Fux 1999, on donne une masse totale de "gaz"  $M_{totg} = 5 \times 10^9 M_\odot$ . Il y a un total de 150000 particules. Donc la masse d'une particule est  $M_{Pg} \approx 3.3 \times 10^4 M_\odot$ . Si on fait la somme des masses du tableau de Smith, on trouve une masse plus grande que  $M_g$ . Prenons pour l'instant les valeurs de Smith. En additionnant, on sait que 99% de la masse de matière interstellaire représente une masse de  $M_{gs} = 6.253 \times 10^9 M_\odot$ , les 1% restant correspondant à la poussière (S pour Smith). La masse totale de matière interstellaire pour Smith est donc  $M_{MISs} = \frac{M_{gs}}{0.99} = 6.316 \times 10^9 M_\odot$ .

Donc, en proportion (par rapport à une masse 1  $M_\odot$ , les différentes composantes correspondent à :

- Milieu ionisé chaud : 0.00047 soit la masse  $M_{i_{hot}} = 2.35 \times 10^6 M_\odot$
- Milieu ionisé moyen : 0.00792 soit la masse  $M_{i_{med}} = 3.96 \times 10^7 M_\odot$
- Milieu neutre chaud : 0.03166 soit la masse  $M_{n_{hot}} = 1.583 \times 10^8 M_\odot$

- Milieu neutre moyen : 0.47498 soit la masse  $M_{n_{med}} = 2.3749 \times 10^9 M_{\odot}$
- Nuages moléculaires : 0.47498 soit la masse  $M_{mol} = 2.3749 \times 10^9 M_{\odot}$
- Poussière : 0.01 soit la masse  $M_d = 5.0 \times 10^7 M_{\odot}$

Les masses sont données pour une masse totale de  $M_{tot_g} = 5 \times 10^9$ . On a donc les proportions de la masse donnée par le modèle de Fux correspondant à chacun des constituants du milieu interstellaire. On connaît les densités des différentes composantes. Il faudrait donc tirer des nuages selon ces paramètres pour éclater des particules de "gaz" : on sait quelle proportion de la masse d'une particule doit être sous forme de poussière, de gaz moléculaire, etc ... et on sait quels volume ces milieux doivent occuper dans le volume de la particule. Il faut aussi une taille caractéristique pour les nuages moléculaires (partie dense) et les nuages atomiques (partie diffuse). Smith les donne au début du livre (*figure 2.2*).

Table 2.2 Major star formation scales. The temperature, T, is in Kelvin and the final column lists the dynamical time scale in seconds.

Phase	Size (cm)	Density g cm <sup>-3</sup>	T (K)	Time (s)
Atomic ISM	10 <sup>21</sup> –10 <sup>20</sup>	10 <sup>-26</sup> –10 <sup>-22</sup>	10 <sup>6</sup> –10 <sup>2</sup>	10 <sup>15</sup>
Molecular cloud	10 <sup>20</sup> –10 <sup>18</sup>	10 <sup>-22</sup> –10 <sup>-18</sup>	10 <sup>2</sup> –10 <sup>1</sup>	10 <sup>14</sup>
Protostar	10 <sup>18</sup> –10 <sup>12</sup>	10 <sup>-18</sup> –10 <sup>-3</sup>	10 <sup>1</sup> –10 <sup>6</sup>	10 <sup>13</sup>
collapse				
Pre-main-seq. contraction	10 <sup>12</sup> –10 <sup>11</sup>	10 <sup>-3</sup> –10 <sup>0</sup>	10 <sup>6</sup> –10 <sup>7</sup>	10 <sup>15</sup>

FIGURE 2.2 –

Pour les nuages atomiques il donne une taille entre 10<sup>20</sup> et 10<sup>21</sup> cm soit entre 32 et 324 pc (1pc=3.085677 × 10<sup>18</sup>cm). Pour les nuages moléculaires entre 10<sup>18</sup> et 10<sup>20</sup> cm soit entre 0.32 et 32 pc. Enfin, on voit que le globule contenant une proto-étoile a une taille entre 10<sup>12</sup> et 10<sup>18</sup> cm soit entre 3.2 × 10<sup>-7</sup> pc et 0.32 pc. Les détails à la surface d'un nuage moléculaire doivent alors avoir une taille caractéristique de 3.2 × 10<sup>-7</sup> pc.

Pour le volume des particules, Mikaël me dit que Françoise Combes lui a donné les données pour le faire :

" Dans les données du modèle de Françoise, pour chaque particule, il y a le rayon du kernel. C'est à dire le rayon d'influence de la particule.

Et la fonction de poids pour répartir les particules dans ce volume était donné dans un fichier fortran :

```

C -----
C WEIGHT : kernel function : spherically symmetric spline kernel function
C proposed by Monaghan and Lattanzio (1985).
C -----
REAL FUNCTION WEIGHT(x,h0)
REAL x,h0
REAL R
INCLUDE 'kernel.h'

R=2.*x/h0
denom=pi*h0**3/8.

IF (R .LE. 1) THEN
WEIGHT = (1.-1.5*R**2+0.75*R**3)/denom
ELSE IF(R .LE. 2) THEN
WEIGHT = (0.25*(2.-R)**3)/denom

```

ELSE  
WEIGHT = 0  
ENDIF

END"

J'ai envoyé un mail à Mikaël pour lui décrire cette idée :

"Salut,

*Je suis en train de réfléchir à la méthode pour simuler le gaz et les poussières.*

*Le modèle de Fux donne la densité de gaz+poussière, vu de loin. Pour avoir une vue réaliste, il faudrait séparer les différents constituants du milieu interstellaire, et avoir des détails quand on s'approche.*

*Voilà le raisonnement que j'ai fait :*

*1) Je dispose des proportions de masse (par rapport à  $1 M_{\odot}$ ) que représentent les différents composants du milieu interstellaire :*

*Milieu ionisé chaud : 0.00047 (transparent en visible)*

*Milieu ionisé moyen : 0.00792 (transparent en visible)*

*Milieu atomique chaud : 0.03166 (translucide en visible)*

*Milieu atomique moyen : 0.47498 (translucide en visible)*

*Nuages moléculaires : 0.47498 (opaque en visible)*

*Poussière : 0.01 (grosses molécules ou grains solides, opaque en visible)*

*2) J'ai la masse totale du "gaz" (gaz+poussière) du modèle de Fux :  $5 \times 10^9 M_{\odot}$ , et le nombre total de particules : 150000, donc une particule a une masse de  $3.3 \times 10^4 M_{\odot}$ .*

*3) Quand on va éclater une particule de  $3.3 \times 10^4 M_{\odot}$ , on saura donc quelle proportion de cette masse devra être donnée à chaque type de matière.*

*4) Voilà comment les nuages se présentent : de petits nuages opaques très denses de molécules et poussière enfouis dans des nuages d'atomes eux-mêmes enfouis dans des nuages ionisés. J'ai aussi la densité et la taille caractéristique max et min de ces phases.*

*Pour les nuages atomiques la taille se trouve entre 32 et 324 pc. Pour les nuages moléculaires entre 0.32 et 32 pc. Enfin, on voit que le globule contenant une proto-étoile a une taille entre  $3.2 \times 10^{-7}$  pc et 0.32 pc. Les plus petits détails à la surface d'un nuage moléculaire doivent alors avoir une taille caractéristique de  $3.2 \times 10^{-7}$  pc.*

*Avec tous ces éléments, et connaissant le volume dans lequel une particule du modèle de Fux doit être éclatée, je me dis qu'on doit pouvoir trouver un algorithme qui fabrique aléatoirement des nuages dans ce volume. Qu'en penses-tu ?*

*Comme d'habitude, il y a la deuxième approche : vue de loin. On n'éclate pas les particules. Il faut trouver un moyen de mettre des grappes de nuages denses le long des bras."*

**20 octobre 2011 :**

Je vois que le livre me donne aussi la fraction de volume occupée par les divers milieux. Ça plus la fraction de masse c'est plus pratique à utiliser qu'une densité (dont la normalisation n'est pas précisée dans le livre).

Voilà les données dont on va avoir besoin :



milieu	v	m	D (pc)
ionisé chaud	0.4-0.7	0.00047	32-324
ionisé moyen	0.15-0.4	0.00792	32-324
atomique moyen	0.2-0.6	0.03166	32-324
atomique froid	0.01-0.04	0.47498	32-324
moléculaire	0.01	0.47498	0.32-32
poussière	0.01 ?	0.01	0.32-32

$v$  est la fraction du volume total du milieu interstellaire occupée par ce type de milieu. Je suppose que cela peut se dire comme : soit  $V$  un volume quelconque d'espace interstellaire. Il contient  $v\%$  de tel ou tel ingrédient.  $m$  est la fraction de masse, par rapport à  $1 M_{\odot}$  correspondant à chaque ingrédient.  $D$  est la taille caractéristique de chaque phase.

Selon le niveau de détail il faudra ajouter des globules, contenant des étoiles en formation, dont la taille est de l'ordre de  $3.2 \times 10^{-7}$  à 0.32 pc, comme le font Nadeau et al. 2000.

J'ai eu une discussion avec Frédéric et Paola. Il faut aussi gérer la formation stellaire.

D'un côté, avec la densité de "gaz" du modèle de Fux, on peut avoir le taux de formation stellaire masse formée à un instant  $T$ , pour un volume donné de gaz, et avec une fonction de masse de Salpeter, savoir quelles étoiles à un instant  $T$  sont formées. On va donc fabriquer des amas ouverts le long des bras, et un peu ailleurs là où le gaz est dense. De l'autre côté, la densité de gaz nous dit combien de nuages et de quelle taille on a pour ce même volume (comme je l'ai dit hier). Là aussi, on aura des nuages le long des bras et un peu en dehors, là où le gaz est dense. Donc normalement, les étoiles jeunes et les nuages devraient se trouver à peu près aux mêmes endroits.

Une fois que dans un cube de galaxie donné on sait combien on d'étoiles apparaissent à un instant donné, et combien de nuages on doit avoir, il est possible de placer correctement les étoiles dans les nuages. Une fois les étoiles placées, les plus lourdes doivent créer des bulles et on doit former des disques de poussière autour des moins massives. Ainsi les étoiles formées sculptent le nuage.

On forme donc des amas ouverts dans les bras. Si on donne à chaque étoile une vitesse et une accélération, l'amas va se déplacer et se disloquer au bout de 1 Gyr et se mélanger au disque. Pendant ce temps, on fabrique de nouveaux amas avec le gaz restant.

### 27 octobre 2011 :

Françoise Combes me dit que les nuages moléculaires se présentent sous forme fractale. voir Pfenniger & Combes 1994. Elle m'a envoyé un code fortran qui contient un algorithme qui construit des nuages fractals. Je l'ai traduit en python, dans le programme "frac.py". Voilà ce que ça donne *figure 2.3*.

J'ai fait des essais de projection 3D. Mais comme ma fonction marche avec des points, ce n'est pas réaliste : j'ai fait deux filtres gaussiens pour lisser l'image. Voilà ce que ça donne *figure 2.4*.

### 28 octobre 2011 :

J'ai corrigé deux trois choses dans le programme. Je l'ai maintenant mis sous la forme d'une fonction "fractalclouds" dans la librairie "RTIGUtil.py". J'ai ajouté un lissage du dernier niveau de fractale. (voir *figure 2.5*)

### Masse des nuages (Pfenniger & Combes 1994)

Soit  $M$  la masse de gaz contenue dans une sphère de rayon  $r$ . Comme le nuage est fractal, la masse obéit à la loi d'auto-similarité :

$$\frac{M}{M_0} = \left( \frac{r}{r_0} \right)^D \quad (2.10)$$

$M_0$  et  $r_0$  sont la masse et l'échelle spatiale de référence, et  $D$  est la dimension fractale de la distribution de masse (Mandelbrot 1982)

Donc :

$$D = \frac{\log_{10}(M/M_0)}{\log_{10}(r/r_0)} \quad (2.11)$$

Cette définition signifie que si on connaît la masse du nuage pour deux échelles spatiales, on a la dimension fractale. Solomon et al 1987 donne une densité moyenne de surface de  $170 M_{\odot}.pc^{-2}$  et les

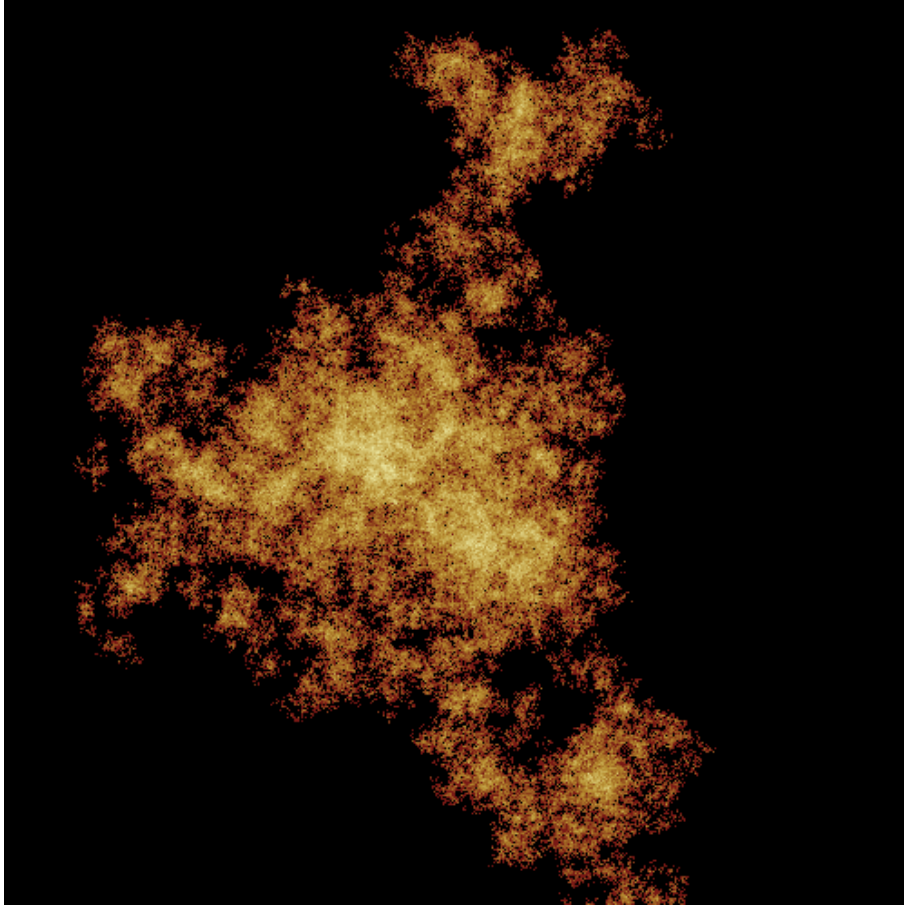


FIGURE 2.3 –

auteurs en déduisent que la masse typique d'un nuage de 30 pc de diamètre est de  $5 \times 10^5 M_{\odot}$ , et que si on regarde la plus petite échelle 30 UA, les clumpuscles, on a une masse de  $0.001 M_{\odot}$ . Alors la dimension fractale d'un nuage est en moyenne de 1.64.

#### Taille des clumps (Pfenniger & Combes 1994)

Soit  $L=10$ , le niveau de départ de la fractale (le nuage dans son ensemble). Soit  $N$  le nombre d'éléments par niveaux. Soit  $M_0$  la masse des clumps au plus petit niveau,  $L=1$ . On a :

$$M_L = N M_{L-1} = N^L M_0 \quad (2.12)$$

On a aussi :

$$M_{tot} = N^{10} M_0 \quad (2.13)$$

Avec l'équation (2.10) on a donc :

$$r_L^D = N r_{L-1}^D \quad (2.14)$$

On définit le facteur d'échelle  $\alpha$  comme :

$$\alpha = \frac{r_{L-1}}{r_L} = N^{-1/D} \quad (2.15)$$

$\alpha$  donne le rapport entre les tailles caractéristiques du nuage, donc les tailles des clumps entre deux niveaux de fractale. Il dépend de  $N$  et  $D$ . La dimension fractale influe donc sur la surface couverte par le nuage pour une masse de gaz donnée. Plus  $D$  est grande, plus le nuage couvre de surface (donc il est moins dense). Inversement, plus  $D$  est petite, plus le nuage se concentre en petits globules denses.

#### En pratique :

Le nombre de niveaux et le nombre d'éléments par niveaux donnent le nombre de particules à afficher.

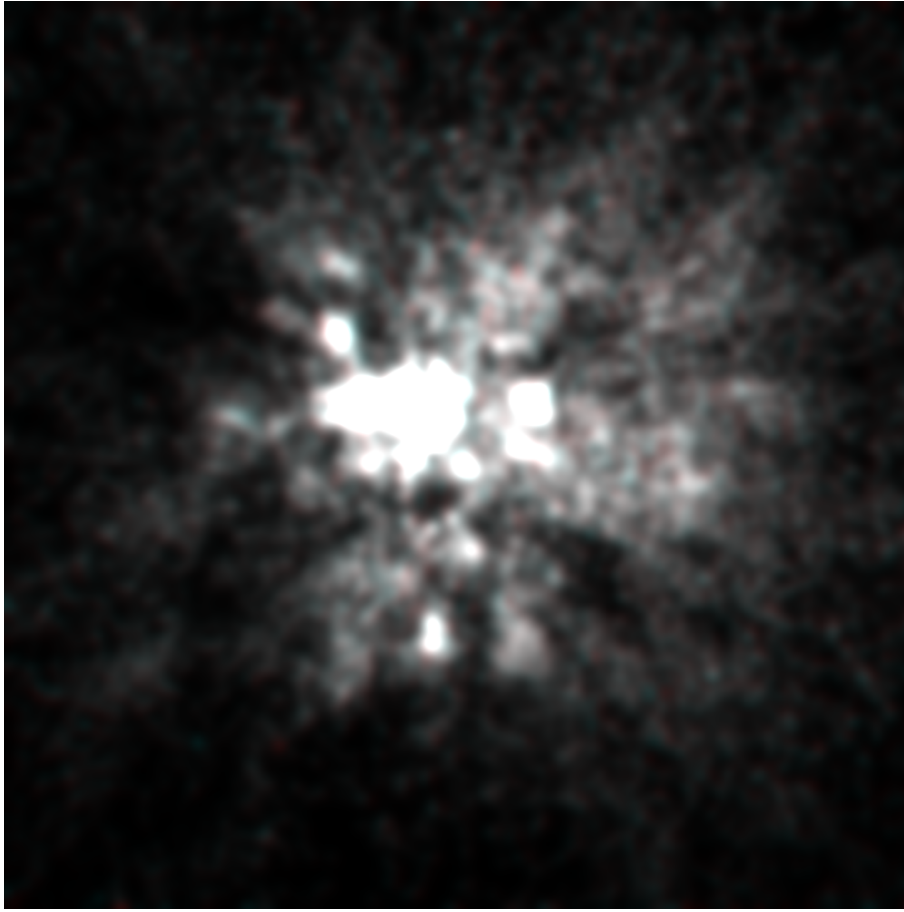


FIGURE 2.4 –

La limitation est informatique. On ne pourra pas simuler un nuage jusqu'aux clumpuscules de 30 UA, vu de très loin. Il faudra donc décider de  $N$  et  $L_{max}$  selon la distance à laquelle l'observateur se trouve par rapport au nuage et la puissance de calcul utilisable. Donc voilà la méthode :

- 1) Prendre une particule de gaz du modèle de Galaxie. On connaît sa masse  $M_{tot}$  et le volume qu'elle occupe, une sphère de rayon  $r_{max}$ .
- 2) Selon la distance à la particule et la capacité de calcul des machines, décider d'un  $N$  et d'un  $L_{max}$ . On prendra des valeurs de  $D$  différentes selon les endroits (c'est 1.64 en moyenne).
- 3) On calcule la fractale dans la sphère de rayon  $r_{max}$ , jusqu'à  $L=2$ .
- 4) On calcule la fractale pour  $L=1$ , et on lisse ce niveau : on ne voit pas de détails au dessous de  $L=2$ . On en verra plus si on s'approche. On connaît la masse des clumps à  $L=2$ . On connaît la masse des clumps à  $L=1$  aussi, donc on peut faire les calculs d'extinction.

Pour les calculs d'extinction on a aussi besoin de la poussière : Sa masse est  $10^{-3} M_{tot}$  dans un nuage.

#### 2 novembre 2011 :

Nuage en 3D avec le programme adapté aux étoiles (*figure 2.6*) :

Françoise Combes m'explique dans un mail que les plus petits nuages qui peuvent exister font  $10^{-3} M_{\odot}$ . Donc si on veut simuler les nuages contenus dans une particule de masse  $M$ , les plus petits niveaux de fractales qu'on devra simuler (si les ressources informatiques le peuvent) contiendront des clumps de  $10^{-3} M_{\odot}$ .

#### 4 novembre 2011 :

La densité de colonne est le nombre de particules contenues dans une section orthogonale d'une ligne de visée, cette section mesurant une unité d'aire. Elle permet d'évaluer le nombre de particules

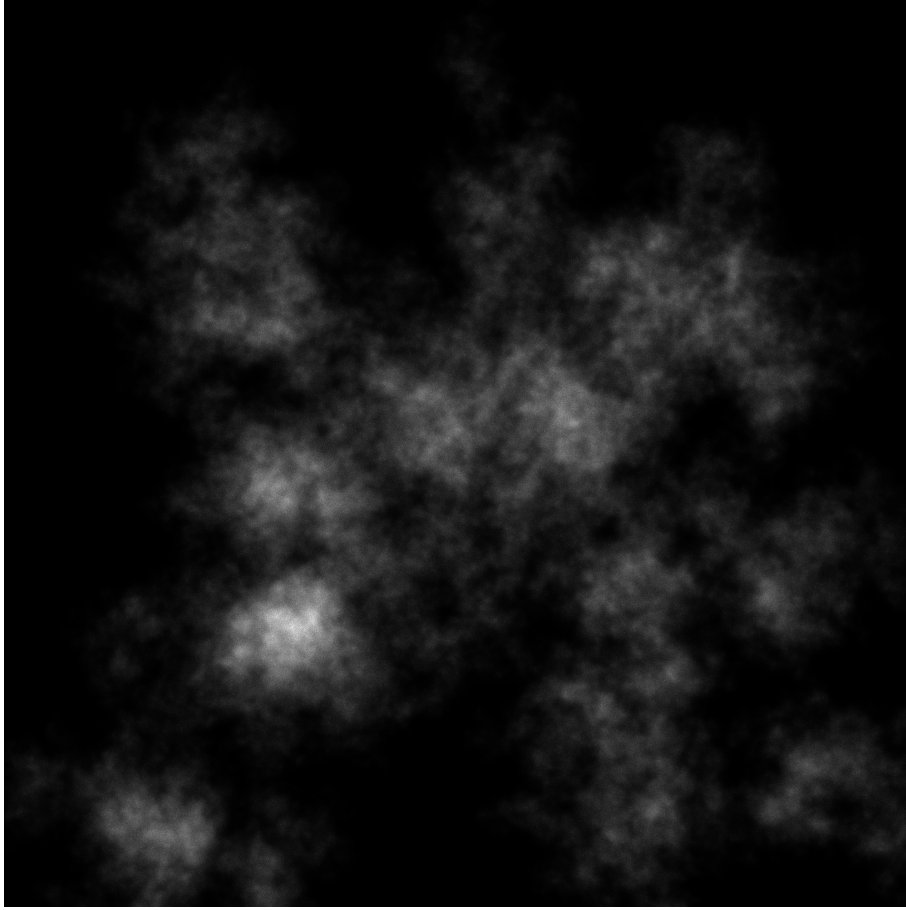


FIGURE 2.5 –  $d=3$ ,  $n=5$ ,  $nlevels=10$

rencontrées par un rayonnement le long de cette ligne de visée. Elle s’emploie en transfert radiatif et dans la description de l’extinction interstellaire.

Intégrale de la densité volumique, la densité de colonne a la dimension inverse d’une aire : son unité du système international est le  $m^{-2}$ , mais les astronomes emploient généralement le  $cm^{-2}$  du système CGS.

#### 15 novembre 2011 :

J’ai reçu les modèles basse résolution de Françoise Combes et Paola DiMatteo. J’ai une série de fichiers ”sft##.dat”, numérotés de 00 à 10, âgés de 0 à 500 millions d’années. Le premier modèle part d’une galaxie déjà formée. Ils contiennent des particules d’étoiles ( $N_{stars} = 40000$ ), des particules hybrides gaz/étoiles ( $N_{gas} = 40000$ ), et des particules de matière noire ( $N_{halo} = 40000$ ). Les 32000 premières particules d’étoiles correspondent au disque, les 8000 autres au bulbe et halo. Les fichiers contiennent 6 colonnes. Chaque vecteur est enregistré linéairement, chacun à la file de l’autre comme ceci :

- $x_{gas+stars+halo}$  : 120000 valeurs
- $y_{gas+stars+halo}$  : 120000 valeurs
- $z_{gas+stars+halo}$  : 120000 valeurs
- $vx_{gas+stars+halo}$  : 120000 valeurs
- $vy_{gas+stars+halo}$  : 120000 valeurs
- $vz_{gas+stars+halo}$  : 120000 valeurs
- $amass_{gas+stars+halo}$  : 120000 valeurs
- $tnow$  : 1 valeur sur la première colonne les 5 autres vides
- $amg_{gas}$  : 40000 valeurs
- $znew_{gas}$  : 40000 valeurs
- $zn_{gas+stars}$  : 80000 valeurs
- $tbirth_{gas+stars}$  : 80000 valeurs



FIGURE 2.6 –  $d=3$ ,  $n=5$ ,  $nlevels=10$

J’ai écrit un programme ‘RTIGE\_galaxy\_models\_2.py’ qui réécrit ces fichiers dans le format RTIGE : avec un header et des colonnes. J’ai écrit un second programme, ‘RTIGE\_S\_catalogue.py’ (l’ancien a été renommé avec la mention `_old`), qui ajoute les âges et modifie les métallicités des particules. Au final j’écris des fichiers ‘RTIGE\_2011\_CombesDiMatteo\_gas\_stars\_sft##.dat’ (particules hybrides gaz/étoiles), ‘RTIGE\_2011\_CombesDiMatteo\_stars\_disc\_sft##.dat’, ‘RTIGE\_2011\_CombesDiMatteo\_stars\_bulb\_sft##.dat’, ‘RTIGE\_2011\_CombesDiMatteo\_darkhalo\_sft##.dat’.

Je travaille sur le modèle sft00. J’utilise la carte d’âges pour donner aux particules du disque un âge entre 6.6 et 9.98 (9.5 Gyr) en  $\log_{10}(\text{yr})$ . Je donne un âge de 11.5 Gyr aux particules du bulbe et 12.5 Gyr aux particules du halo. Ensuite, pour chaque modèle plus récent, j’ajoute 50 millions d’années à chaque particule.

Je regarde les métallicités (voir *figures 2.7 et 2.8*) :

Pour le disque tout est ok, mais pour le bulbe, la métallicité devrait être beaucoup plus faible. Paola me dit que c’est normal, car ils n’ont pas considéré un bulbe comme celui de la voie lactée, formé avant le disque. Je vais donc modifier les valeurs de métallicité des particules du bulbe. Je regarde la variation de la métallicité avec le rayon (voir *figure 2.9*) : Je vais discrétiser les valeurs de métallicité afin d’alléger le modèle. J’en choisis 5 : 0.005, 0.01, 0.015, 0.02, 0.025, 0.03. Pour chaque valeur, je regarde quelles sont les particules qui ont une métallicité à  $\pm 0.0025$ , et je leur donne cette valeur. Je ne peux pas prendre les valeurs 0.035 et 0.04 car le site CMD ne permet pas de produire des isochrones à ces métallicités. La première tranche va donc de 0 à 0.0075, et la dernière de 0.0275 à 0.04.

Pour le halo et le bulbe, je prends trois valeurs :  $Z = 0.0004$ ,  $Z = 0.0008$  et  $Z = 0.0002$  que je réparties aléatoirement en tirant pour chaque particule l’une de ces trois valeurs selon une loi uniforme.

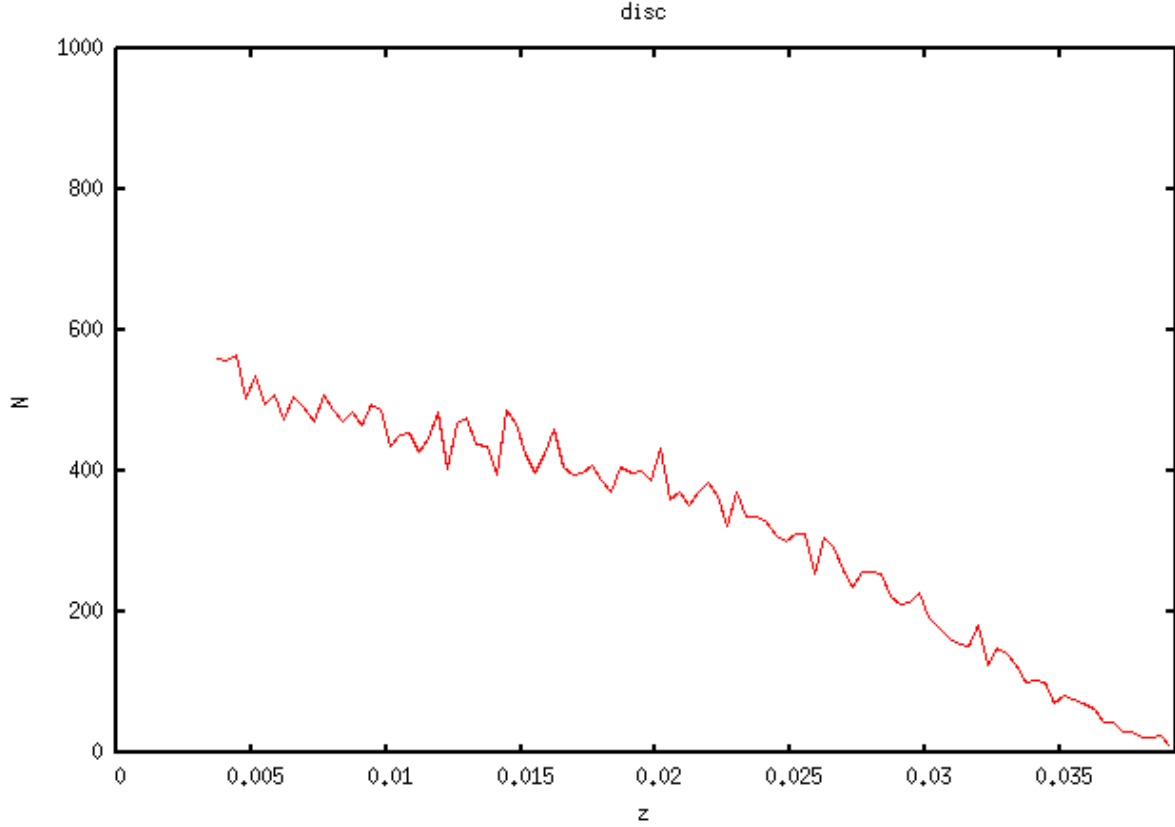


FIGURE 2.7 –

J'ai donc en tout  $5+3=8$  valeurs de métallicités. Je dois télécharger de nouvelles isochrones, et fabriquer de nouveaux cubes de paramètres stellaires. D'abord, le gaz.

#### Particules de gaz :

Toujours dans le programme 'RTIGE\_S\_catalogue.py', je modifie les données concernant les particules hybrides gaz/étoiles. Pour ce qui est des âges, je regarde la masse d'étoiles fabriquée par tranches de 50 millions d'années. Pour chaque modèle, j'ai  $M_{\text{etoiles}} = M_{\text{tot}} - M_{\text{gaz}}$ . La masse totale et la masse de gaz de la particule est donnée dans le fichier. Ainsi, je connais la masse d'étoiles pour chaque particule. Je connais par soustraction des  $M_{\text{etoiles}}$  d'un fichier à l'autre pour chaque particule, la masse de nouvelles étoiles pour chaque particule de chaque modèle. Ainsi je connais l'âge de toutes ces nouvelles étoiles à 50 millions d'années près. Au final, dans le modèle sft10 j'ai des étoiles de 0, 50, 100, 150, 200, 350, 400, 450 et 500 millions d'années dans les particules. Certaines qui ont commencé à former des étoiles un peu après le début de la simulation n'ont pas les plus vieilles étoiles, autrement dit, la proportion de masse d'étoiles de chaque tranche d'âge change selon la particule considérée. Dans les fichiers de modèle j'ajoute donc 11 colonnes, donnant chacune la masse d'étoiles de chaque classe d'âge. Je donne ces classes d'âge en  $\log_{10}(\text{yr})$ , la classe d'âge 0 a l'âge le plus jeune disponible dans les fichiers d'isochrones soit 6.6. Les valeurs en  $\log_{10}$  sont donc : 6.60, 7.70, 8.00, 8.18, 8.30, 8.40, 8.48, 8.54, 8.60, 8.65, 8.70.

Pour la métallicité, j'ai tracé  $z_m$  et  $z_{\text{new}}$ , métallicité moyenne et métallicité des nouvelles étoiles pour chaque particule en fonction de la distance au centre galactique (voir figures 2.10 et 2.11). J'obtiens grosso-modo la même chose dans les deux cas. Et il semble raisonnable de prendre les mêmes métallicités que pour les particules étoiles. Je fais donc la même discrétisation.

#### Paramètres stellaires :

Je télécharge les isochrones pour les nouvelles métallicités. Je modifie le programme 'RTIGE\_S\_isochrones.py',

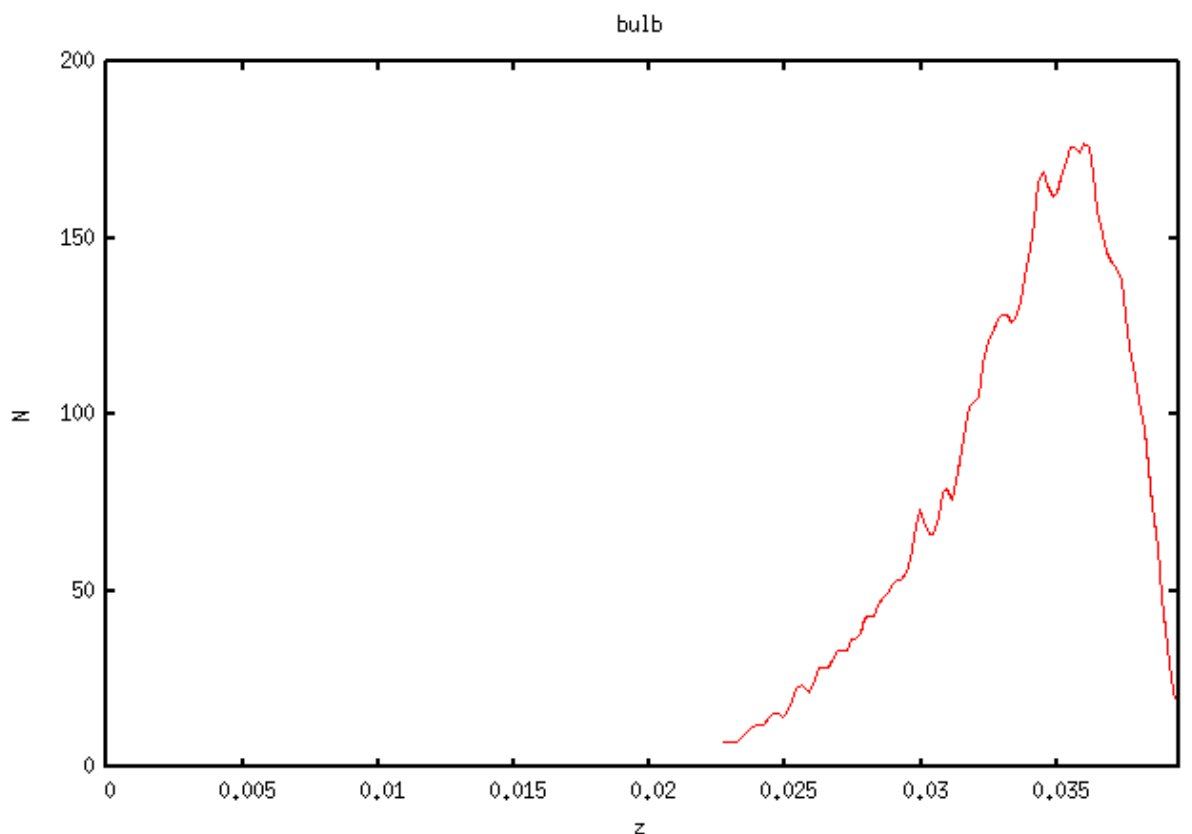


FIGURE 2.8 –

pour qu'il travaille en plus avec les nouvelles valeurs de métallicité (on garde les anciennes, on en a besoin pour les amas ouverts). Pour les paramètres stellaires des naines blanches, j'ai moins de valeurs de métallicité différentes. J'avais associé des chemins évolutif calculés pour une certaine métallicité aux métallicités choisies pour les isochrones. Voilà comment je les associe en prenant en compte les nouvelles valeurs :

```
Z=numpy.array(['0002','0004','0008','003','004','005','01','015','02','0.025','03'])
```

```
Z2=numpy.array(['0001','0004','001','004','004','004','008','008','008','04','04'])
```

J'ai lancé le programme sur aquilon pour les 4 nouvelles valeurs de métallicité : 0.005, 0.015, 0.02 et 0.025.

### 16 novembre 2011 :

Les fichiers d'isochrones sont écrits, j'ai lancé le programme 'RTIGE\_S\_HR\_cubes.py' après y avoir ajouté les nouvelles valeurs de métallicité. Les diagrammes HR sont OK, à part la présence d'étoiles autour des naines blanches ajoutées par mon algorithme. Je ne sais pas d'où viennent ces points, ils n'apparaissent que sur les nouvelles isochrones. Est-ce que quelque chose aurait changé sur leur site ? Il faudra creuser ce mystère plus tard.

Pour avoir le nombre d'étoiles par particules, il faut diviser la masse totale d'une particule par la masse moyenne donnée par la carte de masse.

J'ai donc tout ce qu'il faut pour les étoiles de champ. Je m'occupe maintenant des particules hybrides gaz/étoiles. Voir document "Milieu interstellaire".

Donc pour les particules hybrides gaz/étoiles, j'ai la masse de nouvelles étoiles pour 11 âges différents. J'ai donc besoin de diagrammes HR contenant plusieurs isochrones. Voilà ce qu'il faut faire :

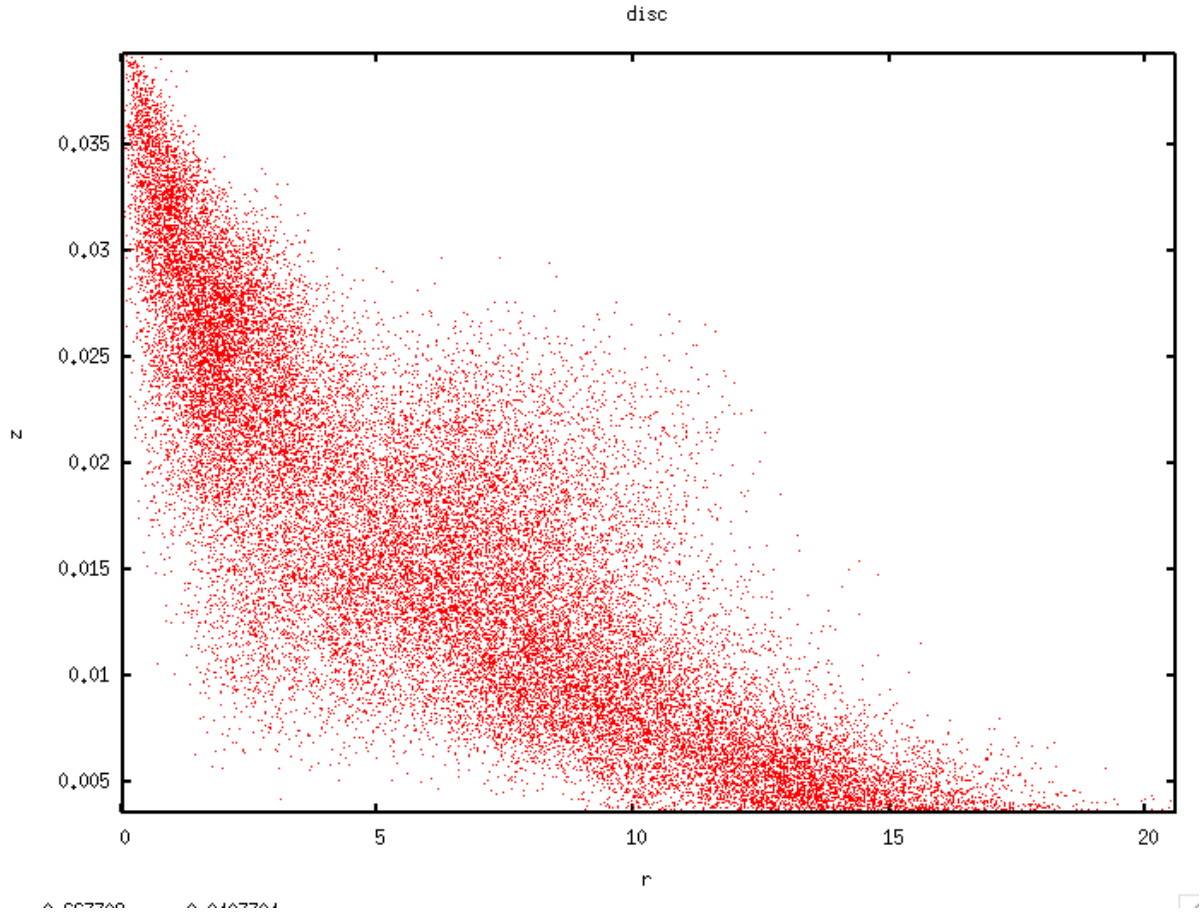


FIGURE 2.9 –

- 1) Pour chaque âge, lire le diagramme isochrone correspondant dans le cube.
- 2) Pour chaque âge, calculer la masse moyenne à partir de la carte masse associée.
- 3) Pour chaque âge, calculer le nombre moyen d'étoiles  $N_i$  en divisant la masse totale des étoiles de cet âge par la masse moyenne calculée.
- 4) Pour chaque âge, normaliser le diagramme d'isochrone  $iso_i$  à 1.
- 5) Construire un diagramme HR par cette opération :  $HR = iso_1 \times N_1 + iso_2 \times N_2 + \dots + iso_{11} \times N_{11}$
- 6) Diviser le diagramme HR par  $\sum N_i$
- 7) Multiplier le diagramme HR par  $10^6$  et prendre le log10 pour pouvoir l'enregistrer en int16.

PROBLEME : Il faut calculer un diagramme HR différent pour chaque particule selon les proportions de masses des étoiles des différentes tranches d'âge !

**17 novembre 2011 :**

J'ai modifié le programme 'RTIGE\_S\_catalogue.py'. J'ai ajouté une colonne "nombre moyen d'étoiles" donnant le nombre d'étoiles calculé pour une particule à partir de la masse moyenne de ses étoiles.

**18 novembre 2011 :**

Après correction d'erreurs de code et l'écriture d'une fonction "Nstars\_mean" dans RTIGEutil.py donnant le nombre moyen d'étoiles pour une masse totale, un âge, un cube de diagramme HR et un cube de paramètres, j'ai lancé 'RTIGE\_S\_catalogue.py'. C'est très lent.

**28 novembre 2011 :**

J'ai maintenant les nombres moyens d'étoiles pour chaque particule étoile et pour chaque tranche d'âge de chaque particule hybride. J'ai voulu voir une image de galaxie en utilisant la méthode que j'ai développée avec Mikaël pour les couleurs. (voir partie méthode). Le bulbe apparaît blanc, le disque rose clair, et les particules hybrides rose foncé. Je ne parviens pas à trouver d'où ça vient. J'ai écrit



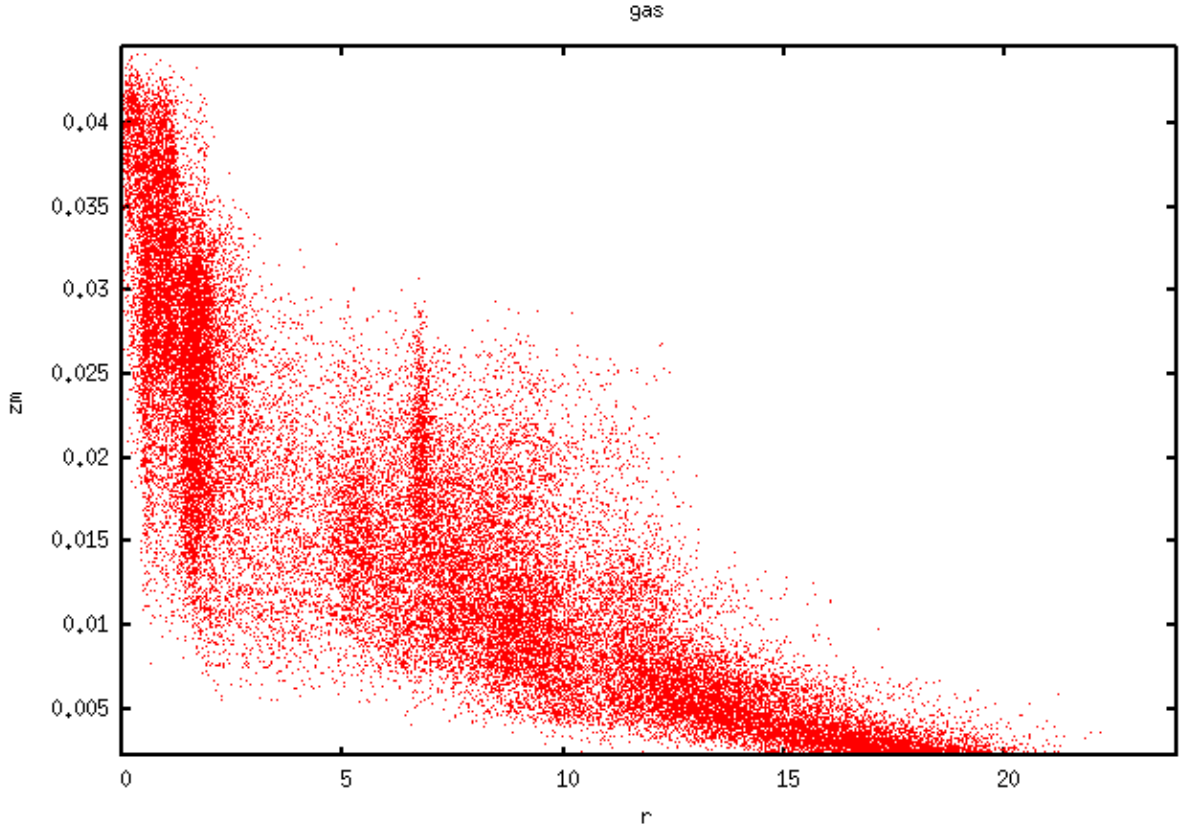


FIGURE 2.10 –

un programme 'galaxy3D2.py' qui fait ça, et un programme 'test4.py' qui le fait pour 2000 particules de chaque ensemble. Pour le calcul des luminosités dans les trois canaux de couleurs, j'ai écrit deux fonctions dans 'RTIGUtil.py', une pour les particules étoiles : 'particle\_colors', et une pour les particules hybrides : 'particle\_colors\_2'.

Je viens de corriger la gestion des cases vides des diagrammes HR : au lieu de mettre les cases à 0 pour tous les vecteurs, je ne fais les calculs que sur les cases non-vides. Les images ne sont plus roses, mais blanches. J'ai remis la multiplication par le nombre d'étoiles dans le calcul de luminosité dans les deux fonctions, et relancé le programme. On va voir. C'est toujours blanc. voir *figures* 2.12 (disque), 2.13 (bulbe), 2.14 (particules hybrides) et 2.15 (tout).

#### Amas ouverts :

Pour les particules hybrides, j'ai besoin de savoir combien elles contiennent d'amas ouverts : les nouveaux et les anciens. Je dispose de la masse totale d'étoiles de chacun des 11 âges. Il me faut donc la fonction de masse initiale d'amas ouverts : c'est-à-dire une fonction qui donne la proportion d'amas de chaque classe de masse qui naissent à un moment donné. Hui-Chen me dit que la proportion d'amas par classe de masse est :

$$\frac{dN}{dM} \propto M^\alpha \quad (2.16)$$

avec  $\alpha = -2$ , et  $M \in [10^4, 10^6 M_\odot]$ . Références :

A&A 494, 539-551 (2009)

DOI : 10.1051/0004-6361 :200811212

Mon. Not. R. Astron. Soc. 405, 857-869 (2010)

doi :10.1111/j.1365-2966.2010.16511.x

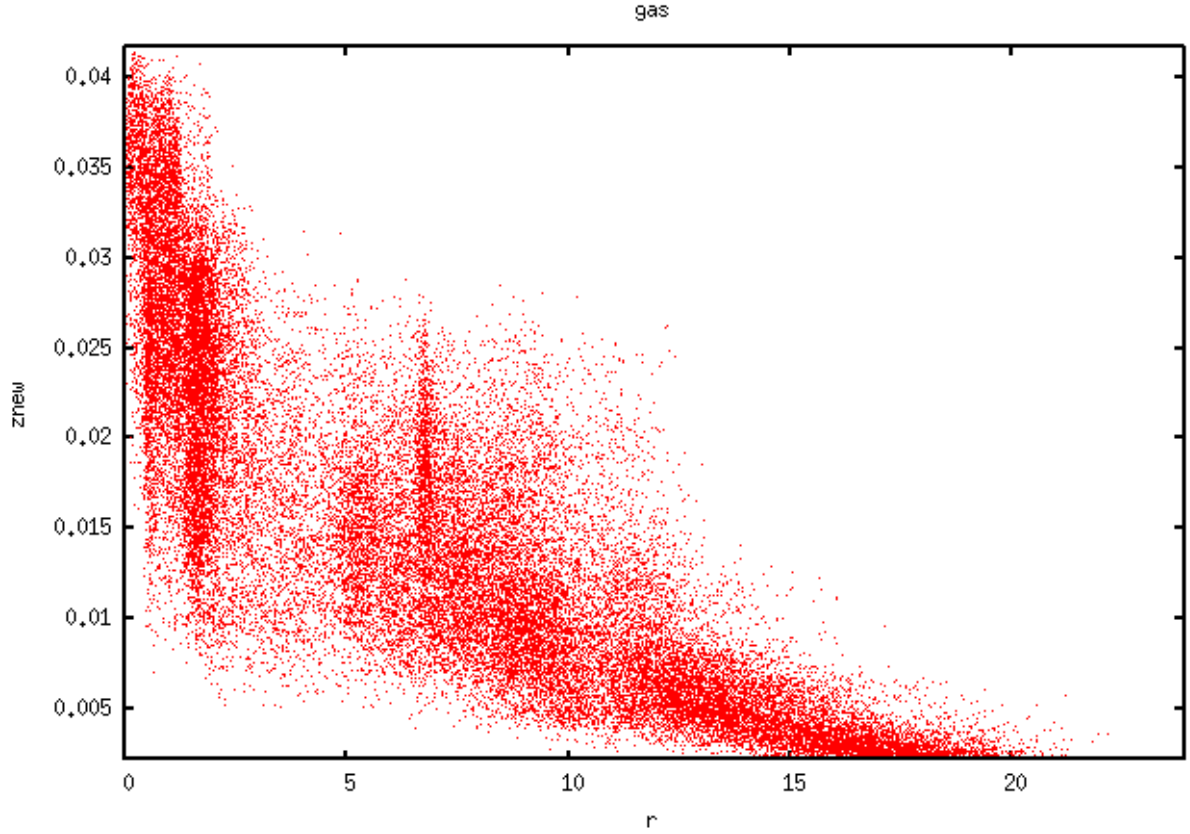


FIGURE 2.11 –

The Astronomical Journal, 135 :823–835, 2008 March  
doi :10.1088/0004-6256/135/3/823

Je peux prendre 5 classes de masse :  $10^2, 10^3, 10^4, 10^5, 10^6 M_{\odot}$ . Problème, quel est  $\alpha$  pour  $M < 10^4 M_{\odot}$  ? Les modèles que donne Paola ont une masse de 250 ou 2000  $M_{\odot}$ , donc je dois aller jusque  $10^2$ .

### 29 novembre 2011 :

Dans Parmentier et al. 2008, je lis :

”For the ICMF, we assume a power-law mass spectrum  $dN/dM_{cl} \propto M_{cl}^{-2}$  over the mass range  $10^2$ – $10^7 M_{\odot}$ .” Cela semble donc raisonnable de prendre  $\alpha = 2$  pour les 5 classes de masse d’amas.

### 30 novembre 2011 :

Dans Bik et al. 2003 ils prennent aussi  $\alpha = 2$  pour des masses entre  $10^{3.5}$  et  $10^{5.5} M_{\odot}$ . Pour chaque particule, je vais donc :

- 1) Prendre la masse totale de nouvelles étoiles
- 2) Tirer des masses d’amas suivant la loi par un algorithme de réjection
- 3) Dernière itération quand la masse totale est atteinte.

J’ai écrit un nouveau programme ‘RTIGE\_OC\_catalogue.py’, qui produit 11 fichiers ‘RTIGE\_2011\_OC\_catalogue\_####.txt’, #### étant l’âge de la simulation.

Pour chaque particule, j’ai 11x5=58 colonnes donnant le nombre d’amas à simuler, de 5 masses différentes, et de 11 âges différents. Pour l’instant les amas vieux tirés dans les modèles plus anciens restent dans les particules des modèles plus récents. Il faudra tenir compte de la dilution des amas dans le disque dans un second temps, simplement en ajoutant quelques lignes dans l’algorithme qui enlève des amas au bon moment.



FIGURE 2.12 –

Les plus petits amas ne doivent pas être représentés et laissés dilués dans la couche de particules hybrides, en utilisant les diagrammes HR. Les plus gros amas (très peu), peuvent être représentés sous forme de sprites plus ou moins brillantes, par dessus les particules. Autour des sprites, il faudra colorer et illuminer la fraction gaz des particules en rose pour représenter l'ionisation.

#### **5 décembre 2011 :**

Pour les âges, il faut raisonner en termes de classe d'âge, pas d'âge!!!! les âges des particules ou amas dépendent du modèle choisi, mais pas la classe d'âge. exemple : modèle 07 : contient les classes d'âge 00 à 07, et les âges de ces classes, sont 8.54 à 6.6.

#### **29 mars 2012 :**

Mikaël me fait part de points aberrants dans les couleurs moyennes des particules, pour un âge de 9.18 en log, et une métallicité de 0.015, 0.02 ou 0.025. Il y a effectivement des points bleus sur les diagrammes HR à cet âge. Je regarde si je retrouve ces points dans les isochrones de départ. Je prends le diagramme #258 (0 pour origine). Je vois deux points à 83,32 et 91,47 sur Qfitsview, et une série de points au-dessus des naines blanches, le plus brillant étant 53,47. Ces points ne devraient pas être là sauf s'il s'agit de la phase transitoire géante rouge naine blanche. Je rappelle le pas des diagrammes en Te : 0.014, et en LV : 0.105.

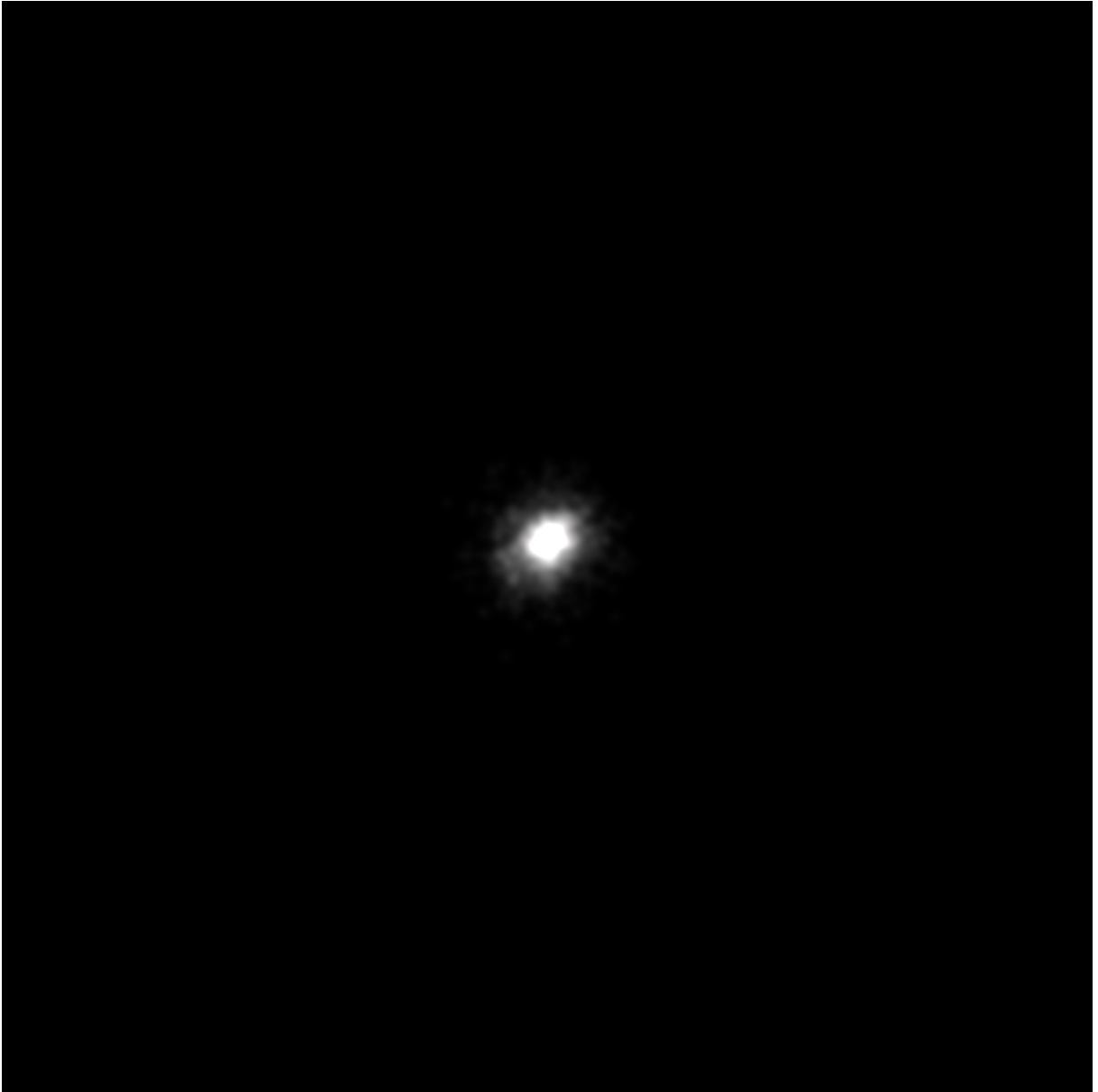


FIGURE 2.13 –

- Point 83, 32 :  $LV=4.715$ ,  $Te=3.374$ . Dans l'isochrone à l'âge 9.18 je vois une ligne à  $Te=3.7325$ ,  $L=4.6990$ , et  $MV=-6.907$ . Ce  $MV$  correspond à  $LV=0.4*(4.74-MV)=4.6588$ . Les deux points correspondent.
- point 91,47 :  $LV=5.555$ ,  $Te=3.958$ . Dans l'isochrone à l'âge 9.18, je vois une ligne à  $Te=3.9596$ ,  $L=5.6134$  et  $MV=-9.165$ . Ce  $MV$  correspond à  $LV=5.562$ . Les deux points correspondent.
- point 53,47 :  $LV=1.565$ ,  $Te=3.958$ . Dans l'isochrone à l'âge 9.18, je vois une ligne à  $Te=4.0360$ ,  $L=0.7520$  et  $MV=3.303$ . Ce  $MV$  correspond à  $LV=0.5748$ . Les deux points correspondent.

Les points 1) et 2) ont des voisins dans l'isochrones, qui sont rassemblés dans la même case de diagramme HR. Les points de la série au dessus- des naines blanches se retrouve au-dessous de la ligne du point 3 avec un température et luminosité proches.

#### 17 avril 2012 :

Je renomme le programme 'RTIGE\_galaxy\_models.py' en 'RTIGE\_galaxy\_models\_old.py' et 'RTIGE\_galaxy\_models' en 'RTIGE\_galaxy\_models\_GALMER09112011.py'.

'RTIGE\_galaxy\_models\_GALMER09112011.py' est alors le programme qui fonctionne avec le modèle GALMER 09/11/2011.

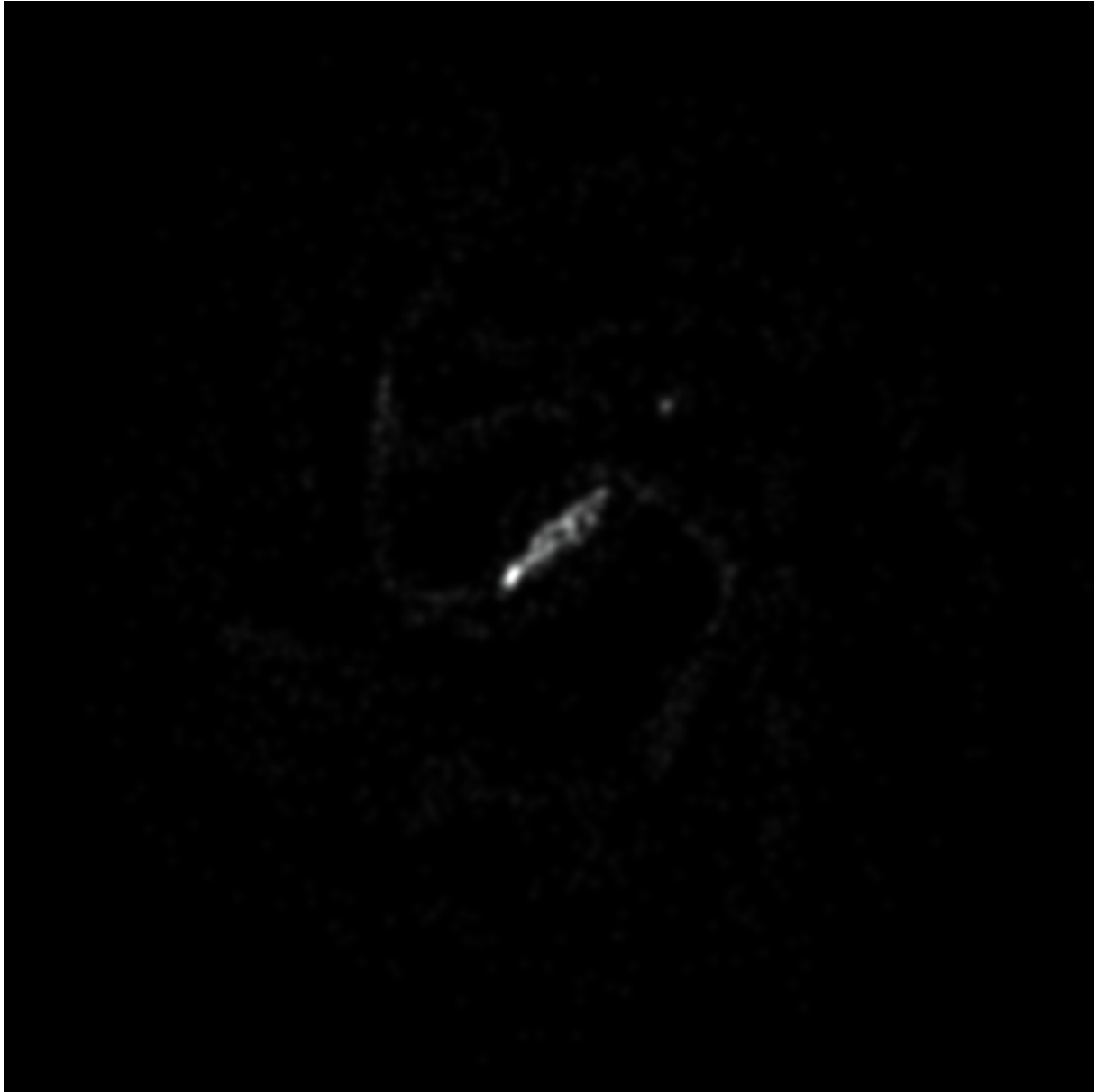


FIGURE 2.14 –

J'écris un programme 'RTIGE\_galaxy\_models\_GALMER05042012.py' qui fonctionne avec le nouveau modèle GALMER 05/04/2012 (30 Mpart).

**18 avril 2012 :**

Le programme 'RTIGE\_galaxy\_models\_GALMER05042012.py' est prêt à être lancé. Je mets à jour le programme 'RTIGE\_S\_catalogue.py' (j'ai ajouté 'old' au programme déjà existant).

**19 avril 2012 :**

J'ai mis à jour le programme 'RTIGE\_S\_catalogue.py'.

ATTENTION : j'ai changé les indices des fichiers GALMER pour qu'ils débutent à 00 au lieu de 01. Pour l'instant, je ne travaille que sur les 11 premiers modèles (00 à 10). Le modèle 00 correspond donc à une galaxie de 11.8 Gyr (13Gyr-120\*10Myr). Le pas est de 10 Myr, donc le modèle 10 correspond à une galaxie de 11.9 Gyr. A terme, il y aura 121 modèles et 121 classes d'âges. Je ne sais pas encore s'il vaut mieux toutes les regrouper dans un même cube pour les fichiers 'RTIGE\_2012\_GALMER\_gas\_stars\_params.dat' et 'RTIGE\_OC\_catalogue.dat', ou bien s'il faut les répartir dans des fichiers de 11 classes d'âges. A voir avec Mikaël pour le temps de lecture que ça demande

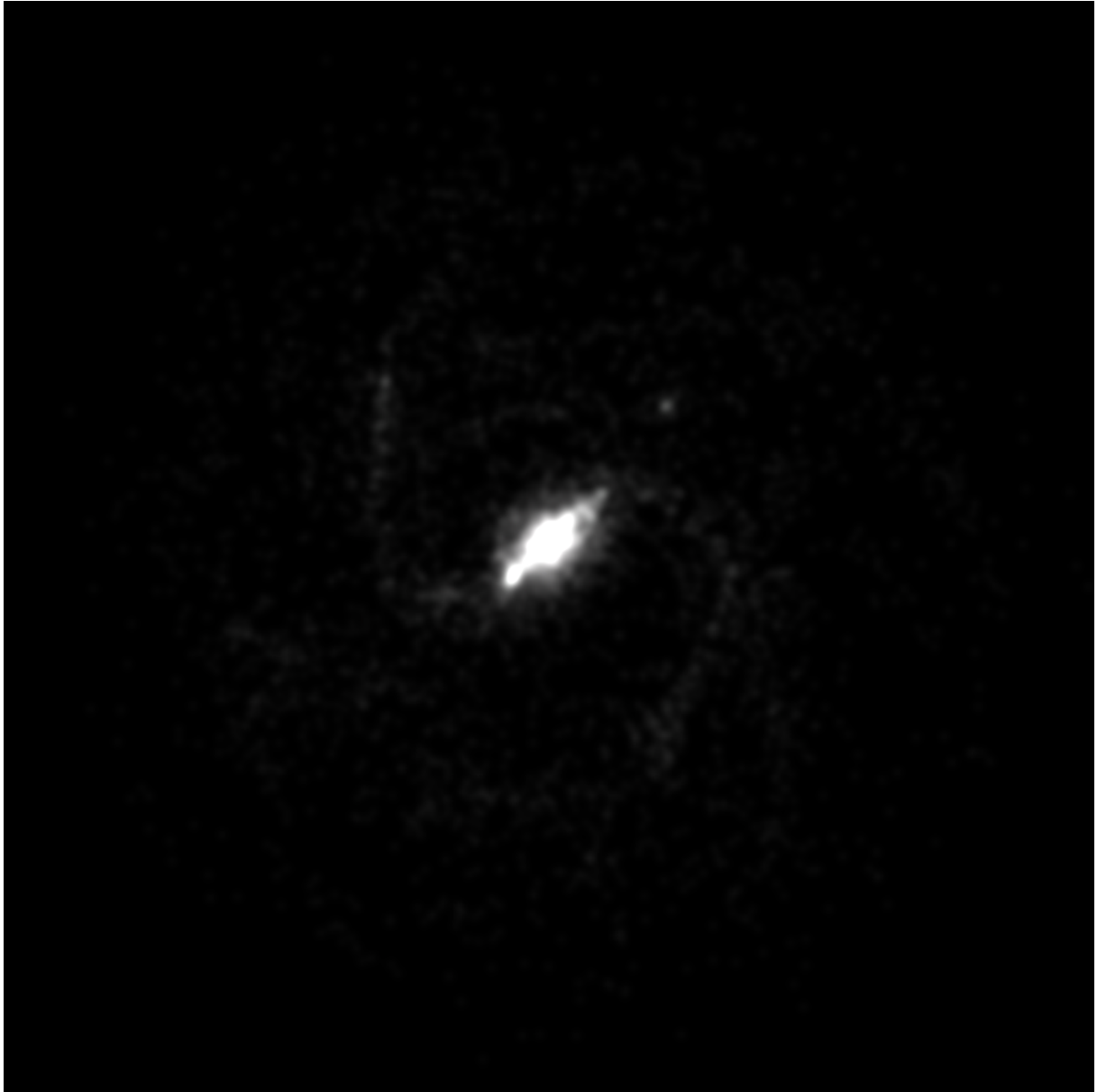


FIGURE 2.15 –

dans les deux cas.

#### 20 avril 2012 :

Mail de Mikaël :

*” Est-ce que tu penses que pour modéliser le dernier âge de la galaxie on aura besoin des 121 classes d’âge ? Ce sont surtout les particules jeunes qui nous intéressent dans ces classes d’âge, on peut peut-être s’en tirer sans utiliser des classes d’âge les plus anciennes ? Ou peut-être utiliser des plus grands pas, de 50Myr comme dans le modèle de démo. La précision à 10Myr n’est peut-être pas nécessaire.*

*Je propose ça car :  $10240000 \text{ particules} * 121 \text{ sous particules} * 7 \text{ données (métallicité, masse, 5 quantités d’amas ouverts)} * 4 \text{ octets} = 32 \text{ Go}$  Ça va pas rentrer en RAM et les calculs vont être vraiment très longs.”*

Voilà ce que je vais faire :

Je vais prendre un pas de 100 Myr, soit 10 fois moins de fichiers, toujours en gardant 1.2 Gyr en tout, soit 12 fichiers, que je renomme en 'out\_100Myr##.dat', avec '##' allant de 00 à 11. Cela correspond aux indices 01, 11, 21, 31 .... 121 dans les fichiers GALMER.

Mail de Paola :

”

*Salut Yannick,*

*pour les unites de dens et h elles sont tjrs dans les unites du code : donc densites en  $2.3 * 10^9 M_{\text{sun}}/\text{kpc}^3$  et h en kpc*

*pour la facon dont les densites sont calculees, j'attends confirmation d'une chose de la part de la personne qui a ecrit le code. j'ai un doute la dessus.*

*Pour le nombre de particules dans le disque, si tu utilises la simulation gSb\_q1p8\_BD0p25, les valeurs correctes de M et Mbul sont 20 et 5 respectivement.*

*Je suis desolee, mais mercredi, apres 10 jj de vacances :) , j'ai fait confusion entre la simulation que je t'ai donnee et d'autres ...*

*amities,*

*p.*

*Salut Yannick,*

*donc pour les densites et les h les uniques quantites qui tu dois regarder sont celles des particuls 1 a Ngas. Pour N ≠ Ngas le code, dans cette version, ne calcule pas de densite ni de h (je ne sais pas bien quelles valeurs il sauvegarde ..)*

*Pour N ≠ Ngas, la densite represente la densite de volume de chaque particule P0, ou cette densite est donnee par la contribution de toutes les particules "voisines" qui se trouvent a une distance de P0 ≤ h (avec h est le rayon qui contient 50 particules de gaz voisines a P0)*

*amities, p. ”*

Particles :

Ngas=10240000

Ndhalo=10240000

Nstars=10240000

Ntot=Nstars+Ngas+Ndhalo

Nbar=Nstars+Ngas

Ndisk=Nbar\*M/(M+Mbulb/facplum)

avec facplum=1.124915, M=20.,Mbulb=5.

Ndisk=numpy.round(Nstars\*20./(25./1.124915))

Nbulb=numpy.round(Nstars-Ndisk)

Ntot is composed of : (sorted)

ngas

ndisk

nbulb, with nbulb+ndisk = nstar

ndm

**23 avril 2012 :**

” *Salut Yannick,*

*je n'ai pas d'autres modes ou le rayon de kernel est calculé pendant la simulation pour les étoiles et la DM. \*Mais\* je pense que vous pouvez qd même utiliser les valeurs qui sont dans le fichier pour h. Benoit me disait que le code calcule h au départ pour toutes les particules, mais après il le recalcule, pendant la simu, seulement pour les particules de gaz. Donc l'h des étoiles et de la DM dans le fichier pourrait donner de tt façon une info ...*

*Pour la masse de la galaxie : oui, elle est la même que le modèle précédent.*

*amities, p."*

## **2 mai 2012 :**

Finalement je fais tourner le code python directement sur le serveur où se trouvent les fichiers. Je prends un pas de 50 millions d'années. Je prends donc les fichiers 01, 06, 11.... 116, 121. Une fois convertis dans le format RTIGE, je change la numérotation afin de travailler avec 25 fichiers, numérotés de 01 à 25.

mail envoyé à Mikaël :

*"Ca y est j'ai compris l'erreur de conception des âges des particules du disque. Je me disais bien qu'il y avait un truc qui clochait.*

*Au lieu de donner un âge selon l'altitude aux particules du modèle 00, le plus vieux / au présent et d'ajouter 50 Myr pour chaque modèle suivant, il faut que je fasse l'inverse :*

- Donner un âge en fonction de l'altitude pour le modèle 24, le plus jeune / au présent.*
- Enlever 50 Myr pour chaque modèle antérieur, et convertir les masses d'étoiles en gaz au fur et à mesure pour les plus jeunes particules.*

*Dans le modèle 24, le disque ne contiendrait alors pas de gaz, mais dans les modèles précédents, certaines particules seraient en fait composées de gaz. Pour les particules du disque je donne un âge moyen, donc il n'y aurait pas de classes d'âges (le taux de formation stellaire n'est pas simulé pour ces particules-là).*

*J'ai juste à inverser une boucle pour ça. Ça devrait pas être trop compliqué."*

## **3 mai 2012 :**

Je me suis trompé hier, les âges dans le disque doivent bien être donnés pour le modèle fictif 00 (âgé de 0 années), soit pour le modèle 01 en ajoutant 10 millions d'années à toutes les particules du disque (le modèle 01 est âgé de 10 millions d'années). Pour les modèles suivants, il faut leur ajouter 50 millions d'années à chaque fois. Et il n'y a plus de formation stellaire dans le disque dès le modèle 01 : les dernières étoiles formées l'ont été dans le modèle 00.

Prenons donc le disque du modèle 01. Quel âge donner aux particules du disque ? On veut que le disque contiennent des particules âgées au plus de 10 Gyr pour le modèle 25.

La simulation dure 1.21 Gyr avec un pas de 50 Myr. Donc, si dans le modèle 01, l'âge maximum est de 8.78 Gyr (9.94 en log10) et l'âge minimum de 10 Myr (7.00 en log10), dans le modèle 25, l'âge maximum est de 9.98 Gyr et l'âge minimum de 1.21 Gyr.  $8.78 \times 10^9 + 24 \times 5 \times 10^7 = 9.98 \text{ Gyr}$ . En prenant ces bornes pour le modèle 01, cela signifie que dans un modèle fictif 00, on aurait un âge maximum est de 8.77 Gyr et un âge minimum de 0 yr.

Si maintenant on voulait simuler le modèle "-01", il faudrait retirer 50+10 Myr aux particules du disque du modèle 01, et transformer les particules d'âge négatif en particules de gaz, afin de rendre compte de la formation stellaire de façon approximative.



Pour ce qui concerne l'âge des modèles, si le présent est 13Gyr pour le modèle 25, et si le pas est de 50 Myr, alors le modèle 01 a un âge de  $13e9-24*5e7=11.8Gyr$ , et le modèle fictif 00 aurait un âge de  $11.8e9-1e7=11.79Gyr$ .

Pour ce qui concerne l'âge du bulbe et du halo : On veut qu'au modèle 25, le bulbe ait un âge de 12Gyr et le halo un âge de 13Gyr. Cela correspond à un âge de  $12e9-24*5e7=10.8Gyr$  pour le bulbe et un âge de  $13e9-24*5e7=11.8Gyr$  dans le modèle 01.

### 10 mai 2012 :

Le programme 'RTIGE\_S\_catalogue.py' est trop lent. Je regarde ce qui est lent.

- Fonction RTIGEutil.Nstars\_mean :

Le where2D est beaucoup trop lent. Le calcul du nombre d'étoiles sur 1000 itérations : 2012-05-10 14 :52 :50.190498 2012-05-10 14 :52 :57.701820

1000 itérations 7.511321 s

Je remplace le where2D par une astuce vectorielle :

2012-05-10 16 :20 :57.629760 2012-05-10 16 :20 :58.117063

1000 iterations 0.48 s

Soit un facteur 16 en temps entre les deux. Je sors les définitions de la carte de masse et de l'isochrone (à partir des cubes) à l'extérieur de la fonction pour gagner un peu de temps.

Voilà la nouvelle fonction :

```
#Normalisation to 1. first step : log10 to relative number / 1e6
isoc2=10**(isoc/1000.)
#replace isoc2 boxes that are 1 by 0 (because they were 0 in isoc, meaning no stars).
#isoc2 has n' values and 1 values. Then isoc 3 has n'-1 values and 0 values
isoc3=isoc2-1
#isoc4 has n'(n'-1) values and 0 values
isoc4=isoc2*isoc3
#isoc5 has n' values and 0 values
isoc5=isoc4/isoc3
#Normalisation step two : /1e6
isoc5/=1e6
#avoid NaN values
ind=numpy.isfinite(isoc5)
#Mean mass
meanmass=numpy.sum(massmap[ind]*isoc5[ind])
#Mean numbers of stars
Nstars=numpy.int64(masstot/meanmass)
```

Je vérifie que dans les deux cas, les 100 premières valeurs de Nstars calculées sont les mêmes :

avec l'ancienne fonction :

```
array([ 18664., 18348., 18326., 18285., 18592., 18353., 18761., 18734., 18362., 18930., 18730., 18693.,
 18348., 18586., 18730., 18761., 18592., 18356., 18348., 18930., 18761., 18585., 18730., 18421., 18730.,
 17792., 18692., 18730., 18664., 18692., 18592., 17807., 18734., 18761., 18586., 18708., 18432., 18432.,
 18267., 18761., 18592., 18348., 18240., 18730., 18586., 18592., 18693., 18586., 18761., 18356., 18785.,
 18785., 18692., 18664., 18348., 18285., 17675., 18730., 18692., 18734., 18692., 18567., 18785., 18341.,
 18734., 18693., 18432., 18708., 18664., 18930., 18730., 18930., 18285., 18692., 18930., 18432., 18592.,
 18586., 18655., 18438., 18563., 18441., 17686., 18353., 18761., 18362., 18324., 18708., 18761., 18379.,
 18730., 18664., 18280., 18324., 18567., 18348., 18761., 18240., 18421., 18365.]])
```

Avec la nouvelle :

```
array([ 18664., 18348., 18326., 18285., 18592., 18353., 18761., 18734., 18362., 18930., 18730., 18693.,
```

18348., 18586., 18730., 18761., 18592., 18356., 18348., 18930., 18761., 18585., 18730., 18421., 18730., 17792., 18692., 18730., 18664., 18692., 18592., 17807., 18734., 18761., 18586., 18708., 18432., 18432., 18267., 18761., 18592., 18348., 18240., 18730., 18586., 18592., 18693., 18586., 18761., 18356., 18785., 18785., 18692., 18664., 18348., 18285., 17675., 18730., 18692., 18734., 18692., 18567., 18785., 18341., 18734., 18693., 18432., 18708., 18664., 18930., 18730., 18930., 18285., 18692., 18930., 18432., 18592., 18586., 18655., 18438., 18563., 18441., 17686., 18353., 18761., 18362., 18324., 18708., 18761., 18379., 18730., 18664., 18280., 18324., 18567., 18348., 18761., 18240., 18421., 18365.])

Le code avait calculé l'âge des particules du disque du modèle 01. Je sauve ce vecteur logagearr dans '/data2/yannick/RTIGE/data/Modeles\_galaxie/GALMER/05042012/RTIGE\_2012\_GALMER\_stars\_disc.logagearr01.dat', et je relancerai le code une fois optimisé, en omettant cette étape et en relisant ce fichier pour avoir les âges.

### 18 mai 2012 :

Test optimisation 'RTIGE\_OC\_catalogue\_hampi.py'

Essais avec les 100 premières particules.

1) Double boucle : extérieur n, intérieur i

d : 2012-05-18 15 :54 :27.180470      f : 2012-05-18 15 :55 :26.955834  
 $\Delta t=59.775364$  s

Double boucle : extérieur i, intérieur n

d : 2012-05-18 15 :55 :26.955845      f : 2012-05-18 15 :56 :25.340472  
 $\Delta t=58.384627$  s

Cas 2 très légèrement plus court.

2) En enlevant les if (amas tirés mais pas classés)

Double boucle : extérieur n, intérieur i

d : 2012-05-18 16 :33 :29.854530      f : 2012-05-18 16 :34 :26.791234  
 $\Delta t=56.936704$  s

Double boucle : extérieur i, intérieur n

d : 2012-05-18 16 :34 :26.791256      f : 2012-05-18 16 :35 :23.072634  
 $\Delta t=56.281378$  s

Très légèrement plus court.

3) En enlevant les if (amas tirés mais pas classés)

En enlevant le test sur la masse d'étoiles (si pas nul, on tire des amas)

Double boucle : extérieur i, intérieur n

d : 2012-05-18 16 :59 :09.590514      f : 2012-05-18 17 :00 :04.973615  
 $\Delta t=55.383101$  s

Très légèrement plus court. Il semble que ce ne soient pas les if qui ralentissent la boucle, car en les enlevant ça ne change presque pas le temps de calcul.

4) Double boucle : extérieur i, intérieur n

amsarr=params[ :, :1] avant la boucle

puis ams=amsarr[n,i] dans boucle

d : 2012-05-18 17 :51 :55.040055      f : 2012-05-18 17 :52 :49.447606  
 $\Delta t=54.407551$  s

Un peu mieux que le 1). On gagne 4 secondes.

En fait 4 secondes ça ne veut rien dire, quand je lance le programme plusieurs fois, le temps de calcul n'est jamais le même, à quelques secondes près.

**Problème :**

C'est normal que pour le modèle 121 (modèle 25 avec le pas de 50 Myr) je trouve des masses stellaires de moins de 100 Msun en soustrayant "masse totale" - "masse de gaz" ?

Par exemple, pour la première particule, j'ai :

Mtot=898.4375 Msun

Mgas=825.3089 Msun

=, Mstars=73.1285 Msun

Ca voudrait dire que dans chaque modèle successif, on forme une ou deux étoiles tout au plus par particule.

En pratique dans les 25 classes d'âges pour cette particule, j'ai en masse solaires :

Classe Z Msun

1 0.01 0.3979

2 0.01 2.8267

3 0.01 3.1901

4 0.01 4.4758

5 0.01 2.6082

6 0.01 1.679

7 0.01 4.6667

8 0.01 3.8364

9 0.01 2.3345

10 0.01 3.0751

11 0.01 4.6115

12 0.01 2.4472

13 0.01 2.7761

14 0.01 3.2982

15 0.01 2.4518

16 0.01 3.2568

17 0.015 2.7623

18 0.015 5.4211

19 0.015 2.9831

20 0.015 1.5157

21 0.015 3.6363

22 0.015 3.6317

23 0.015 1.5962

24 0.015 1.2305

25 0.015 2.4196

Si c'est vrai, alors les particules sont énormément moins massives que pour l'ancien modèle (je suppose que c'est logique vu l'augmentation du nombre de particules). Mais alors je ne peux pas donner de nombre d'amas par particules. Il faudrait alors considérer 1 amas fait de plusieurs particules ?

Si oui, alors il faut entièrement revoir la façon dont on catalogue les amas ouverts ...

Je bloque.

### **Voilà en fait le problème qui se pose à nous :**

Avec l'ancien modèle GALMER on avait un nombre de particules hybrides de l'ordre de 40 000, et leur masse était de 4MMsun je crois.

Pour chaque particule, je divisais la masse d'étoiles en amas d'étoiles, de chaque classe d'âge.

Les amas qui avaient une masse  $> 10^4$  Msun étaient représentés par des sprites sur l'image, pour ressortir sous forme de grumeaux. Les autres restaient dilués dans le fond diffus de l'image. En effet, la résolution spatiale du modèle n'était pas suffisante (distance entre les particules bien supérieure à la taille des amas) pour représenter des grumeaux d'amas d'étoiles. Il fallait donc le faire artificiellement. En sommant simplement la lumière des amas, on obtient une pâte floue à cause de ce problème de résolution. Dans la vraie vie, ce problème de résolution n'existe pas (les galaxies ne sont pas faites de particules), et donc les grumeaux sortent tous seuls (après ça dépend du télescope utilisé bien sûr).

Avec le nouveau modèle GALMER, on a 10 millions de particules hybrides, et leur masse est de 800 Msun. Forcément, on divise la masse en plusieurs particules. Après avoir soustrait le gaz, il ne reste que 73 Msun d'étoiles à répartir en amas dans 25 classes d'âges. Ce qui est impossible. Là aussi on a un problème de résolution : un amas doit être constitué de plusieurs particules ! Mais la distance entre deux particules dans le modèle GALMER ne dépend pas de l'appartenance ou non d'une masse stellaire à un amas. Bref, on ne peut plus employer la même méthode de génération d'amas.

Quand on somme simplement les luminosités des particules on obtient une pâte floue et bleue, sans grumeaux (ou presque). Impossible de représenter les amas sous forme de sprites car ils nous sont inconnus !

On ne peut pas se baser sur la densité de particules par élément de volume pour savoir où sont les amas, car la distance entre particules n'est pas liée à l'appartenance à un amas.

Une solution serait de fabriquer une nouvelle couche au modèle à partir de la partie stellaire des particules hybrides, et de dégrader la résolution afin d'avoir moins de particules et ainsi pouvoir les décomposer en amas.

Ou alors utiliser la densité locale des particules et le rayon de kernel donnés par le modèle ? mais je ne suis pas sûr de savoir ce que ça représente.

*Figure 2.16*, une image avec l'ancien modèle et les grumeaux, et *Figure 2.17* une image avec le nouveau modèle, nécessairement sans grumeaux.

### **21 mai 2012 :**

Je dois de toute façon fabriquer un disque d'amas ouverts d'une manière ou d'une autre. Le système développer avec le cube 'RTIGE\_OC\_catalogue.dat' permettait d'avoir les données des amas pour n'importe quel âge de modèle. Si je fais un disque d'amas, il sera nécessairement lié à un âge de modèle pour les positions. Il faut que je trouve un format "universel".

Ce dont je dispose c'est de la masse totale d'étoiles par particule et par classe d'âge. J'ai donc l'info pour simuler n'importe quel modèle. Problème : les particules sont trop petites, elles sont moins massives que le plus petit amas que nous avons choisi de simuler (100 Msun).

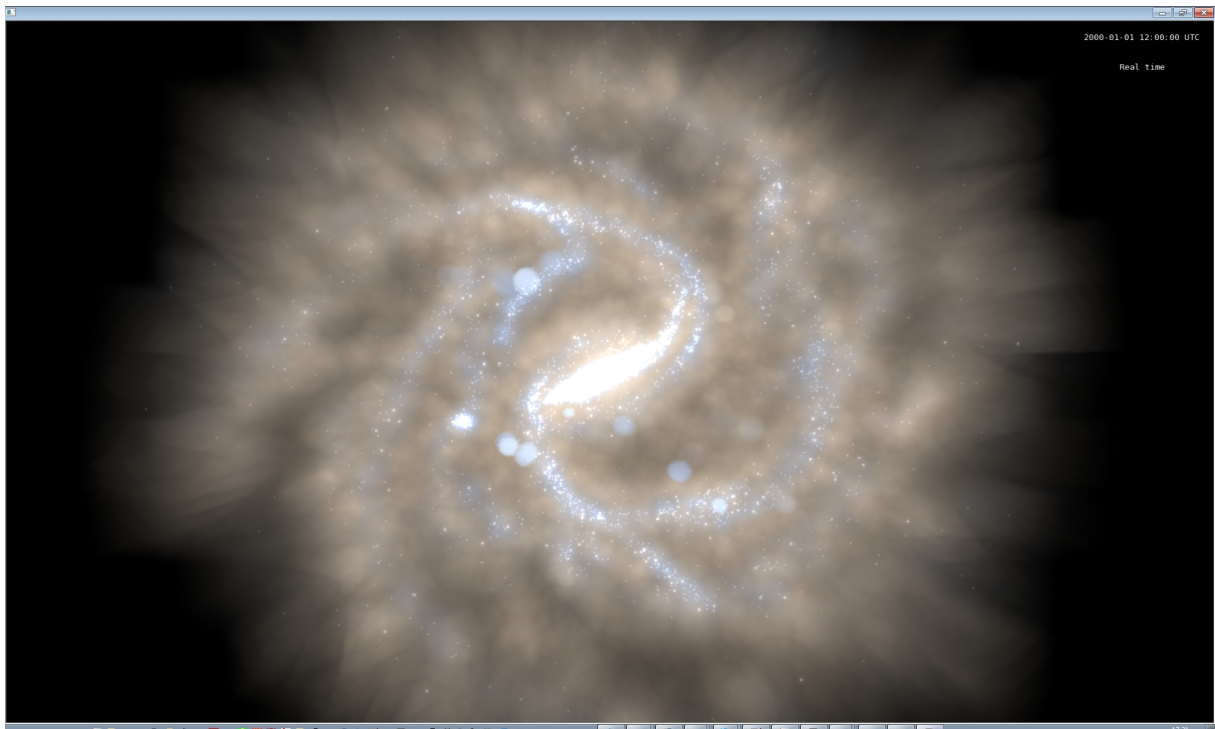


FIGURE 2.16 –

Solutions :

1) Prendre la partie stellaire des particules hybrides et dégrader la résolution pour avoir des particules plus grosses. Il faudrait pour cela regrouper des particules proches et sommer les masses totales d'étoiles dans chaque classe d'âge. Quels critères de regroupement ?

Problème : Deux particules côte à côte dans le modèle 01 ne le sont plus du tout dans le modèle 25. Donc un amas défini par un groupe de particules au modèle 01 a mécaniquement disparu au modèle 25. C'est possible mais ce n'est pas comme ça que disparaît un amas.

2) Faire une seule grosse particule en sommant toutes les masses totales d'étoiles pour chaque classe d'âge. Cette "particule" ne serait pas une sphère mais aurait la forme de tous les petits volumes élémentaires juxtaposés. Ensuite, il suffirait de tirer des amas pour chaque tranche d'âge. On aurait alors un disque d'amas pour chaque classe d'âge, et donc pour chaque modèle. Mais on ne tient plus compte de la densité ...

J'ai discuté avec Frédéric. Voilà les deux solutions qu'il en ressort :

1) Prendre une sphère contenant un certain nombre de particules et tirer des amas selon l'ICMF jusqu'à atteindre la masse totale de cette grosse particule.

Problème : Si on prend des grosses particules de  $10^6$  Msun, leur contenu en particules va considérablement changer d'un modèle à l'autre (la galaxie tourne).

2) Tirer un amas selon l'ICMF, et rassembler suffisamment de particules pour le former, puis renouveler l'opération jusqu'à ce que toutes les particules soient utilisées.

C'est le processus qui se rapproche le plus du processus naturel. **Je vais travailler sur cette seconde solution.**

**Pour chaque classe d'âge, tirer un amas selon l'ICMF, et rassembler suffisamment de particules pour le former, puis renouveler l'opération jusqu'à ce que toutes les particules soient utilisées.**

Une fois qu'un amas est formé à partir d'un groupe de particules, il faut l'associer à la particule centrale, et donner une masse stellaire nulle aux autres particules du groupe, pour la classe d'âge

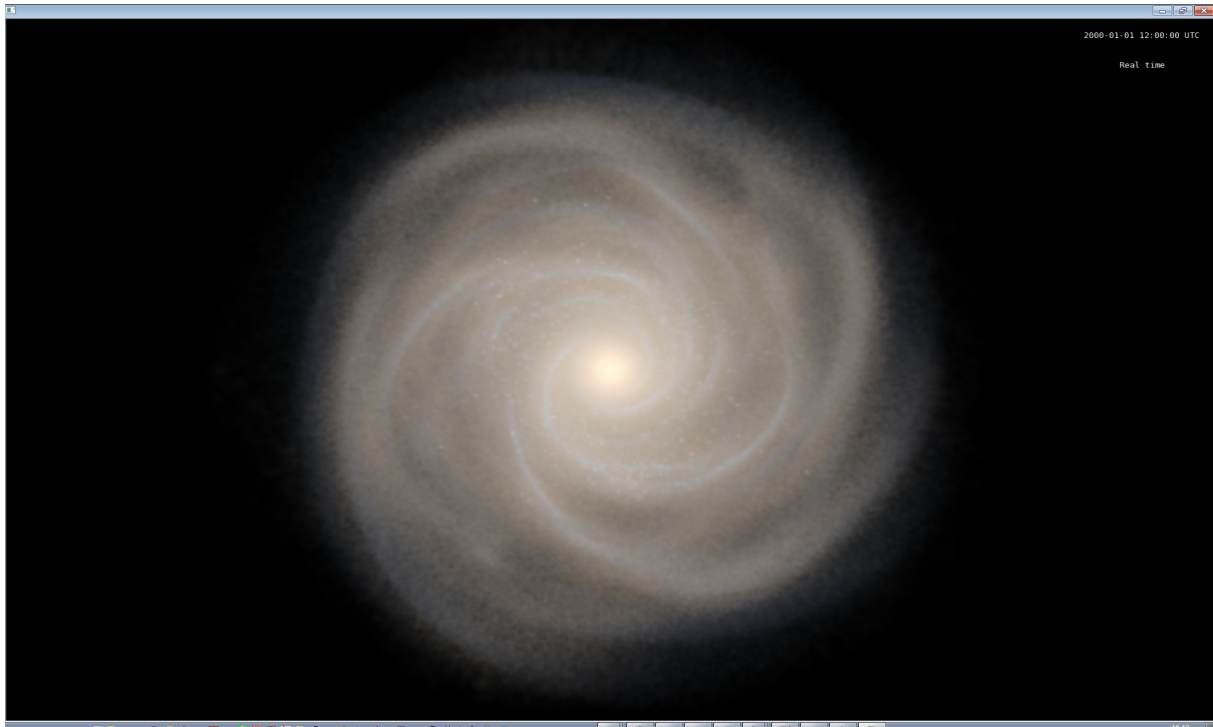


FIGURE 2.17 –

courante. Cet amas suivra cette particule pour les 25 modèles. Le gaz doit lui rester associé à toutes ces particules séparément.

**Questions :**

- Comment choisit-on la particule centrale à partir de laquelle on rassemble la masse ?
- Comment fait-on pour les amas de la classe d'âge suivante, est-ce qu'on garde la même particule centrale, ou bien est-ce qu'on change ?

**I) Pour le premier amas du modèle 01 :**

- a) Tirer une masse  $M$  d'amas.
- b) Tirer au hasard un rang de particule : ce sera la particule associée à cet amas.
- c) Ecrire un algorithme qui ajoute la particule la plus proche et qui somme la masse stellaire de la classe d'âge 01, jusqu'à ce que la masse de l'amas soit atteinte.
- d) Dans le catalogue d'amas (même format qu'avant, une matrice 25 classes d'âges x 5 classes de masse x 10240000 particules), ajouter un amas de masse  $M$  et de classe d'âge 01 à ma particule centrale, et inscrire 0 amas d'aucune classe de masse et de classe d'âge 01 pour les autres particules utilisées (leur masse stellaire est dorénavant contenue dans l'amas).
- e) Garder la masse de gaz répartie dans toutes ces particules.

**II) Pour les amas suivants du modèle 01 :**

- a) Tirer une masse  $M$  d'amas.
- b) Tirer au hasard un rang de particule : ce sera la particule associée à cet amas.
- c) Ecrire un algorithme qui ajoute la particule la plus proche et qui somme la masse stellaire de la classe d'âge 01, jusqu'à ce que la masse de l'amas soit atteinte. **Si à une itération la particule repérée est une particule déjà utilisée par un amas, tirer une nouvelle particule centrale : la particule centrale choisie était trop proche d'un autre amas.**
- d) Dans le catalogue d'amas (même format qu'avant, une matrice 25 classes d'âges x 5 classes de masse x 10240000 particules), ajouter un amas de masse  $M$  et de classe d'âge 01 à ma particule centrale, et inscrire 0 amas d'aucune classe de masse et de classe d'âge 01 pour les autres particules utilisées (leur masse stellaire est dorénavant contenue dans l'amas).

e) Garder la masse de gaz répartie dans toutes ces particules.

- Comment fait-on pour les amas de la classe d'âge suivante, est-ce qu'on garde la même particule centrale, ou bien est-ce qu'on change ?

### III Pour les amas du modèle suivant 02 :

a) Tirer une masse  $M$  d'amas.

b) Tirer au hasard un rang de particule : ce sera la particule associée à cet amas.

c) Ecrire un algorithme qui ajoute la particule la plus proche et qui somme la masse stellaire de la classe d'âge 02, jusqu'à ce que la masse de l'amas soit atteinte. **Si à une itération la particule repérée est une particule déjà utilisée par un amas, tirer une nouvelle particule centrale : la particule centrale choisie était trop proche d'un autre amas.**

d) Dans le catalogue d'amas (même format qu'avant, une matrice 25 classes d'âges x 5 classes de masse x 10240000 particules), ajouter un amas de masse  $M$  et de classe d'âge 02 à ma particule centrale, et inscrire 0 amas d'aucune classe de masse et de classe d'âge 02 pour les autres particules utilisées (leur masse stellaire est dorénavant contenue dans l'amas).

e) Garder la masse de gaz répartie dans toutes ces particules.

**S'il reste pour une quelconque classe d'âge des particules ne contribuant pas à des amas, considérer les étoiles contenues dans ces particules comme de tout petits amas se fondant dans les étoiles de champ.**

A la fin de ces opérations, on a un cube de données de 25 classes d'âges x 5 classes de masse d'amas x 10240000 particules, soit 10 millions de cartes donnant le nombre d'amas pour une particule, de chaque classe d'âge et de chaque classe de masse. Exactement comme avant, mais avec plus de particules et plus de résolution spatiale.

### 22 mai 2012 :

Echange mails avec Mikaël : Il faut tirer les particules centrales en prenant en compte la densité des particules dans le disque hybride, afin de localiser les amas le long des surdensités de gaz et des bras.

Deuxième échange de mails :

Mikaël :

*"Ceci dit, si on les choisit aléatoirement, on a plus de chance de tomber dans une zone dense étant donné qu'il y a plus de particules dans les zones denses (toutes les particules sont de masses à peu près équivalentes ?)"*

Moi :

*"Ah oui, tu as raison. J'ai vérifié, les particules hybrides ont bien toutes la même masse, donc la densité est uniquement représentée par la distance entre elles. En tirant aléatoirement on devrait donc automatiquement respecter les densités."*