Structure Interne des Étoiles :
Modèles Polytropiques

G. Niccolini

11 novembre 2019

1 Considérations théoriques

1.1 Équation de Lane-Emden

On considère un étoile que l'on supposera à symétrie sphérique de rayon $R_\star$ et de masse $M_\star$. On suppose l'étoile à l'équilibre hydrostatique décrit par l'équation suivante :

$$\frac{dP(r)}{dr} = -G\,\frac{m(r)}{r^2}\,\rho(r)\ , \tag{1}$$

où G est la constante de gravitation, $m(r)$ la masse comprise dans une sphère de rayon r, $\rho(r)$ la densité en r et $P(r)$ la pression.

On a de plus l'équation suivante pour $m(r)$:

$$\frac{dm(r)}{dr} = 4\pi\rho(r)r^2\ . \tag{2}$$

Les équations (1) et (9) forment un système de deux équations pour trois inconnues. Dans un étude plus poussée, une équation décrivant le transport d'énergie (radiatif, convectif, diffusion) ainsi que sa production ainsi que des équations d'état sont nécessaire. Dans l'approche simplifiée que nous souhaitons entreprendre on suppose une relation de fermeture, dite équation polytropique liant P à ρ de la manière suivante :

$$P = K\,\rho^{\frac{n+1}{n}}\ , \tag{3}$$

où K et n sont des constantes, cette dernière appelée « indice polytropique ».

1- On introduit une nouvelle variable θ définie par :

$$\rho(r) = \rho_{\rm c}\ [\theta(r)]^n\ , \tag{4}$$

où $\rho_{\rm c}$ est la densité au centre de l'étoile, c.a.d $\rho_{\rm c}\hat{=}\rho(0)$. Montrer qu'en combinant les Éq. (1) et (9) et à l'aide du changement de variable ci-dessus (4) on obtient l'équation suivante :

$$\frac{K\,(n+1)\,\rho_{\rm c}^{\frac{1}{n}-1}}{4\pi\,G}\,\frac{1}{r^2}\,\frac{d}{dr}\left(r^2\,\frac{d\theta}{dr}(r)\right) = -\theta^n(r)\ . \tag{5}$$

2- En introduisant la longueur $\lambda_n = \left(\frac{K\,(n+1)\,\rho_c^{\frac{1}{n}-1}}{4\pi\,G} \right)^{\frac{1}{2}}$ et la variable sans dimension $x \hat{=} \frac{r}{\lambda_n}$ montrer que l'on obtient l'équation de Lane-Emden suivante :

$$\frac{d}{dx}\left(x^2\,\frac{d\theta}{dx}(x) \right) = -x^2\,\theta^n(x)\ . \tag{6}$$

3- Montrer que l'on a les conditions aux bords suivantes :

$$\frac{d\theta}{dx}(0) = 0 \tag{7}$$
$$\theta(0) = 1\ . \tag{8}$$

4- À l'aide de la définition de la masse $M_\star = 4\pi \int_0^{R_\star} \rho(r)\,r^2\,dr$ montrer que l'on a :

$$M_\star = -4\pi\rho_c\,\lambda_n^3\,x_\star^2\frac{d\theta}{dx}(x_\star)\ . \tag{9}$$

5- On suppose que la pression vérifie $P(R_\star) = 0$, ou de manière équivalente que $\theta(x_\star) = 0$. On peut donc déterminer $x_\star$ à partir de la solution $\theta(x)$ de l'Éq. (6). Montrer alors à partir de l'Éq. (9) et de $R_\star = x_\star\,\lambda_n$ que l'on a les relations suivantes :

$$\rho_c = \frac{M_\star}{4\pi\,R_\star^3}\,\frac{x_\star}{|\frac{d\theta}{dx}(x_\star)|}\ , \tag{10}$$
$$P_c = \frac{G}{4\pi(n+1)}\,\frac{1}{|\frac{d\theta}{dx}(x_\star)|^2}\,\frac{M_\star^2}{R_\star^4}\ . \tag{11}$$

6- Montrer que les variations de ρ, P et m en fonction de x sont données par :

$$\rho(x) = \rho_c\,[\theta(x)]^n\ , \tag{12}$$
$$P(x) = P_c\,\,[\theta(x)]^{n+1}\ , \tag{13}$$
$$m(x) = M_\star\left(\frac{x}{x_\star}\right)^2\frac{\frac{d\theta}{dx}(x)}{\frac{d\theta}{dx}(x_\star)}\ . \tag{14}$$

1.2 Équation d'état

On suppose que la pression totale au sein de l'étoile est donnée par la somme de la pression cinétique et de la pression de radiation :

$$P = \frac{\rho k_B T}{\mu\,m_H} + \frac{1}{3}\,a\,T^4\ , \tag{15}$$

où μ est une quantité sans dimension appelé (improprement) « poids moléculaire » moyen, m_H la masse atomique de l'hydrogène et a la constante radiative.

On introduit $\beta = \frac{P_{\rm cin}}{P}$ où $P_{\rm cin}$ est la pression cinétique $P_{\rm cin} = \frac{\rho k_{\rm B} T}{\mu\, m_{\rm H}}$. On supposera ce facteur constant et indépendant de r au sein de l'étoile (modèle d'Eddington).

7- Montrer que la température est donnée par :

$$T(x) = \frac{\mu m_{\rm H}}{k_{\rm B}}\, \beta\, \frac{P_{\rm c}}{\rho_{\rm c}}\, \theta(x)\ . \tag{16}$$

8- Montrer alors que la pression totale peut se mettre sous la forme polytropique suivante :

$$P = K\, \rho^{\frac{4}{3}}\ , \tag{17}$$

avec K donnée par :

$$K = \left[\left(\frac{k_{\rm B}}{\mu\, m_{\rm H}} \right)^4 \frac{3}{a}\, \frac{1-\beta}{\beta^4} \right]^{\frac{1}{3}}\ . \tag{18}$$

L'équation d'état (17) correspond donc à une équation polytropique d'indice $n = 3$ lorsque β est constant.

En remarquant que $K = \frac{P_{\rm c}}{\rho_{\rm c}^{\frac{1+n}{n}}}$ et en considérant les équations (10) et (11) donnant la densité et la pression au centre de l'étoile, on obtient l'équation non linéaire suivante pour β :

$$\frac{1-\beta}{\beta^4} = \frac{a}{3} \left(\frac{\mu\, m_{\rm H}}{k_{\rm B}} \right)^4 \frac{P_{\rm c}^3}{\rho_{\rm c}^{\frac{3(n+1)}{n}}}\ . \tag{19}$$

Le membre de droite ne contient implicitement que des grandeurs caractérisant l'étoile : $M_\star$, $R_\star$, μ, n et la solution de $\theta(\star) = 0$, $x_\star$ restant à déterminer.

2 Aspects numériques

Afin de résoudre le modèle stellaire, il est nécessaire de résoudre l'équation de Lane-Emden (6) ainsi que l'équation (19) donnant β our des paramètres stellaires : $M_\star$, $R_\star$, μ donnés.

2.1 Résolution de l'équation de Lane-Emden

On va utiliser une méthode numérique simple permettant de remplacer l'Éq. (6) par un équivalent discret. On introduit pour cela une grille de valeurs discrètes pour x définie de la manière suivante :

$$x_i = \left(i - \frac{1}{2} \right) \Delta x\ , \tag{20}$$

où Δx est le pas d'échantillonage spatial et $i \geq 1$. On notera que le centre de l'étoile $x = 0$ correspond à un indice $i = \frac{1}{2}$ demi entier. On imagine les valeurs x_i comme représentant les centre de cellules en x délimitées par 2 murs $x_{i-\frac{1}{2}}$ à gauche et $x_{i+\frac{1}{2}}$ à droite.

La valeur x_0 est donc en dehors du domaine physique et appartient à ce que l'on appelle une cellule « fantôme » ; elle sera utilisée pour assurer les conditions aux bords (7) et (8). Notons les solutions exactes de l'Éq. (6) par $\theta_i = \theta(x_i)$ et les approximations numériques que nous allons obtenir par $\bar{\theta}_i$.

1- L'équation différentielle (6) comporte des dérivées que l'on souhaite estimée. Pour cela montrer à l'aide de développement de Taylor que l'on a :

$$\frac{\theta_{i+1} - \theta_i}{\Delta x} = \frac{d\theta}{dx}(x_{i+\frac{1}{2}}) + \frac{1}{24}\frac{d^3\theta}{dx^3}(x_{i+\frac{1}{2}})\,\Delta x^2 + O\left(\Delta x^4\right) \ , \tag{21}$$

ce qui nous permettra d'estimer $\left[x^2 \frac{d\theta}{dx}(x)\right]_{x=x_{i+\frac{1}{2}}}$ par l'approximation suivante :

$$x_{i+\frac{1}{2}}^2 \frac{d\theta}{dx}(x_{i+\frac{1}{2}}) \approx x_{i+\frac{1}{2}}^2 \frac{\bar{\theta}_{i+1} - \bar{\theta}_i}{\Delta x} \ . \tag{22}$$

On obtient à l'aide de cette approximation le « schéma [1] » numérique suivant :

$$\frac{1}{\Delta x}\left(x_{i+\frac{1}{2}}^2 \frac{\bar{\theta}_{i+1} - \bar{\theta}_i}{\Delta x} - x_{i-\frac{1}{2}}^2 \frac{\bar{\theta}_i - \bar{\theta}_{i-1}}{\Delta x}\right) = -x_i^2\, \bar{\theta}_i^n \ , \tag{23}$$

qui représente une version discrète de l'Éq. (6). La méthode est dite d'ordre 2 car l'erreur introduite est un $O\left(\Delta x^2\right)$ comme on peut le voir dans l'Éq. (21).

La méthode Éq. (23) nous donne une relation de récurrence entre les $\bar{\theta}_i$:

$$\bar{\theta}_{i+1} = \bar{\theta}_i + \left(1 + \frac{\Delta x}{2\,x_i}\right)^{-2} \left[\left(1 - \frac{\Delta x}{2\,x_i}\right)^2 (\bar{\theta}_i - \bar{\theta}_{i-1}) - (\bar{\theta}_i)^n\,\Delta x^2\right] \ . \tag{24}$$

On admettra [2] que le procédé de calcul des $\bar{\theta}_i$ ci-dessus est stable dès lors que $\Delta x < 1$.

La relation de récurrence (24) doit être amorcée à l'aide de deux valeurs $\bar{\theta}_0$ et $\bar{\theta}_1$. On notera que $\bar{\theta}_0$ est en dehors du domaine physique ($x_0 < 0$). C'est une valeur « fantôme » dont le seul but et de nous permettre d'assurer les conditions initiales (7) et (8) dont les versions discrètes d'ordre 2 sont :

$$\frac{\bar{\theta}_1 - \bar{\theta}_0}{\Delta x} = 0 \ , \tag{25}$$

$$\frac{\bar{\theta}_1 + \bar{\theta}_0}{2} = 1 \ , \tag{26}$$

dont les solutions triviales $\bar{\theta}_1 = \bar{\theta}_0 = 1$, nous donnent les valeurs nécessaires au démarrage de (24).

2.2 Détermination de $x_\star$

On utilise le procédé (24) jusqu'à l'obtention d'une valeur de $\bar{\theta}_i$ négative, obtenue pour l'indice $i = i_\star$. Une interpolation d'ordre 2 de $\theta(x)$ est alors donnée

1. Traduction hasardeuse du mot anglais *scheme*.
2. La démonstration de ce résultat n'est pas aisée et fait appel en particulier à un théorème sur certaines propriétés des racines des polynômes dû à Miller 1971 ...

par la formule d'interpolation de Lagrange suivante :

$$\theta(x) \approx \bar{\theta}_{i_\star} \frac{(x - x_{i_\star-2})(x - x_{i_\star-1})}{2\Delta x^2} - \bar{\theta}_{i_\star-1} \frac{(x - x_{i_\star})(x - x_{i_\star-2})}{\Delta x^2} + \bar{\theta}_{i_\star-2} \frac{(x - x_{i_\star-1})(x - x_{i_\star})}{2\Delta x^2} \ . \quad (27)$$

2- L'approximation numérique de $x_\star$, notée $\tilde{x}_\star$ sera obtenue en annulant le membre de droite de l'Éq. (27). Montrer que l'on obtient l'équation quadratique suivante :

$$\begin{aligned}&\left(\bar{\theta}_{i_\star} - 2u_{i_\star-1} + \bar{\theta}_{i_\star-2}\right) x^2 \\ &\quad - x \left[\bar{\theta}_{i_\star}(x_{i_\star-2} + x_{i_\star-1}) - 2\bar{\theta}_{i_\star-1}(x_{i_\star} + x_{i_\star-2}) + \bar{\theta}_{i_\star-2}(x_{i_\star} + x_{i_\star-1})\right] \\ &\qquad + \bar{\theta}_{i_\star} x_{i_\star-2} x_{i_\star-1} - 2\bar{\theta}_{i_\star-1} x_{i_\star} x_{i_\star-2} + \bar{\theta}_{i_\star-2} x_{i_\star} x_{i_\star-1} = 0 \ . \quad (28)\end{aligned}$$

Cette équation est du type $ax^2 + bx + x = 0$ qui se résout de manière numérique à l'aide de la formule bien connue :

$$x_\star = \frac{-b \pm \sqrt{b^2 - 4ac}}{2\,a} \ , \quad (29)$$

où l'on ne retiendra que la racine pertinente pour laquelle $x_{i_\star-1} \leq x_\star \leq x_{i_\star}$.

3- Dans les considérations qui suivent, nous avons également besoin d'une estimation de la quantité $\frac{d\theta}{dx}(x_\star)$. Pour cela, en dérivant (27) et en utilisant l'estimation $\tilde{x}_\star$ montrer que l'on peut prendre :

$$\frac{d\theta}{dx}(x_\star) \approx \bar{\theta}_{i_\star} \frac{2\tilde{x}_\star - x_{i_\star-2} - x_{i_\star-1}}{2\Delta x^2} - \bar{\theta}_{i_\star-1} \frac{2\tilde{x}_\star - x_{i_\star} - x_{i_\star-2}}{\Delta x^2} + \bar{\theta}_{i_\star-2} \frac{2\tilde{x}_\star - x_{i_\star-1} - x_{i_\star}}{2\Delta x^2} \ . \quad (30)$$

2.3 Détermination de β

L'équation (19) est une équation non linéaire que l'on peut résoudre par la méthode de Newton-Rapshon. Pour cela on introduit la fonction $f(\beta)$ définie de la manière suivante :

$$f(\beta) = \delta\,\beta^4 - 1 + \beta \ , \quad (31)$$

avec

$$\delta = \frac{a}{3}\left(\frac{\mu\, m_{\rm H}}{k_{\rm B}}\right)^4 \frac{P_{\rm c}^3}{\rho_{\rm c}^{\frac{3(n+1)}{n}}} \ . \quad (32)$$

La méthode de Newton-Raphson est une méthode itérative pour trouver les racines d'une équation $f(\beta) = 0$ qui consiste à déterminer des approximation successives, notées β^k, à l'aide du procédé suivant :

$$\beta^{k+1} = \beta^k + \frac{f(\beta_k)}{f'(\beta_k)} \ , \quad (33)$$

qui démarre pour une valeur initiale β^0 donnée. On pourra prendre $\beta^0 = 1$.

On poursuit se procédé jusqu'à ce qu'un critère d'arrêt soit vérifié. On pourra prendre par ex :

$$\left| \frac{\beta^{k+1}}{\beta^k} - 1 \right| \leq \epsilon \ , \tag{34}$$

où ϵ est une constante. Ce critère est basé sur la différence relative entre deux itérés successifs. On prendra typiquement une valeur $\epsilon \sim 10^{-8}$.

3 Développement

On souhaite développer un programme résolvant le problème d'une étoile polytropique correspondant au modèle d'Eddington pour le Soleil décrit dans les sections précédentes. On comparera les résultats obtenus à un modèle dit « standard » prenant en compte plus de physique, notamment les processus de transport d'énergie, les réactions nucléaires, etc .... Dans ce qui suit nous présentons quelques étapes essentielles du programme, en mettant l'accent sur des difficultés techniques liées à l'allocation dynamique de la mémoire.

On développera ce programme en `C` dans le standard `C11`. Pour cela, il suffit de rajouter avec `gcc` l'option de compilation `-std=C11`. On commence par écrire un programme de base, comportant la fonction `main` et quelques définitions :

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <assert.h>

/*****************************/
/********** Constants *********/
/*****************************/

#define REALLOC_FACTOR 1.25
#define TRUE 1
#define FALSE 0

const double Msun = 1.9891e30; /* kg */
const double Rsun = 6.95508e8; /* m */
const double G = 6.673e-11;  /* Nm^2kg^{-2} */
const double kb = 1.3806503e-23; /* JK^{-1} */
const double a = 7.565767e-16; /* Jm^{-3}K^{-4} */
const double amu = 1.66053873e-27; /* kg */
const double pi = 3.14159265358979323846;

int main(void) {
  return EXIT_SUCCESS;
}
```

Nous allons tout d'abord définir les paramètres stellaires :

```
  /****************************************/
  /********** Stellar parameters *********/
  /****************************************/

  double Rstar = Rsun;
```

```c
double Mstar = Msun;
double X = 0.7; /* H abundance */
double mu = 4. / (3. + 5. * X);
double n = 3.;
```

Ensuite, il nous faut allouer la mémoire contenant les valeurs de x et $\theta(x)$ de la manière suivante :

```c
size_t nx = 1024;
double* theta = (double*) calloc(nx, sizeof(double));
double* x = (double*) calloc(nx, sizeof(double));
```

Lorsque nous intégrons l'équation pour $\theta(x)$ nous avançons en aveugle et ne nous pouvons pas connaître à l'avance la taille des tableaux servant à stocker les résultats. Pour gérer cette situation, il sera peut être nécessaire d'ajuster la taille des tableaux et donc de ré-allouer de la mémoire, ainsi qu'éventuellement copier son contenu, de manière dynamique. Afin de nous aider dans cette tache, on utilisera la fonction suivante, à placer avant `main` :

```c
/*************************/
/********** resize *********/
/*************************/

int resize(size_t size, double** data) {
  /* This method is used to resize an array and handling
     properly
     whenever we cannot allocate the memory */

  double* new_data = realloc(*data, size * sizeof(double));

  if(new_data) {
    *data = new_data;
  } else {
    return FALSE;
  }
  return TRUE;
}
```

Il nous reste à écrire une fonction qui sera l'intégrateur de l'équation de Lane-Emden (6). On donne ci-dessus les éléments concernant la ré-allocation de mémoire sans donner l'implémentation du schéma numérique[3]. Si lors de l'intégration nous atteignons la taille maximale du tableau, nous utilisons la fonction `resize` ci-dessus, en demandant une fraction (ici $\frac{1}{4}$) de la mémoire en plus. Voici une possible implémentation d'une telle fonction :

```c
/*****************************************************/
/********** Lane-Emden equation 2nd order scheme **********/
/*****************************************************/

int lane_emden(size_t* nx, double** x, double** theta, double
    n, double dx) {
  size_t size = *nx;
  double dx2 = dx * dx;
```

3. Il faut bien travailler un peu jeunes padawanes.

```
/* TODO : you might need some definitions */
for(size_t k = 1;; ++k) {
  /* If we reach the size of the arrays, we need to re-
      allocate
     memory */

  if(k >= size - 2) {

    /* We increase the size by a factor 'REALLOC_FATOR') */

    size_t new_size = (size_t) (size * REALLOC_FACTOR);
    ierr = resize(new_size, x);

    /* Checking if memory could be allocated */
    if(!ierr) {
      return ierr;
    }

    /* Checking if memory could be allocated */
    ierr = resize(new_size, theta);
    if(!ierr) {
      return ierr;
    }
    size = new_size;
  }

  /****************************/
  /* 2nd order numerical scheme */
  /****************************/

  (*theta)[k + 1] = /* TODO : you need to code the scheme
      here /*

  /*********************************/
  /* We reached the outer boundary */
  /*********************************/

  if((*theta)[k + 1] <= 0.) {
    *nx = k + 2;
    /* We realloc again to the actual size, in order to free
        the
       useless memory */
    ierr = resize(*nx, x);
    if(!ierr) {
      return ierr;
    }

    ierr = resize(*nx, theta);
    if(!ierr) {
      return ierr;
    }
    break;
  }
}
```

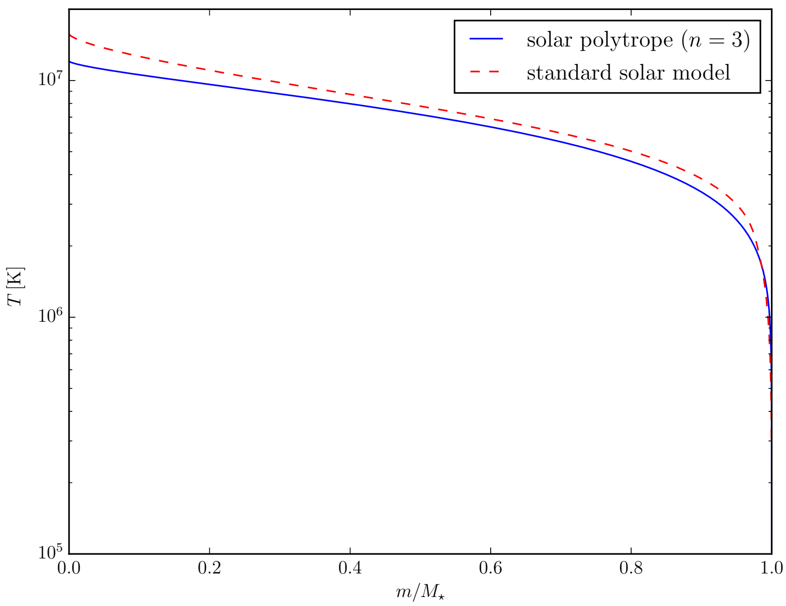

FIGURE 1 – Temperature en fonction de la masse m dans le modèle solaire polytropique.

```
   return TRUE;
}
```

3.1 Ce n'est pas terminé . . .

Nous n'avons ici présenté qu'une possible implémentation d'une partie seulement du code. Il reste à calculer β, $x_\star$, $\frac{d\theta}{dx}(x_\star)$. Il faut également calculer $\rho(x)$, $T(x)$, $P(x)$, $m(x)$ et écrire tout cela dans un fichier de sortie.

Vous aurez certainement besoin de fonctions pour calculer β :

```
double compute_beta(double delta, double beta0, double epsilon
    ) {
 double beta = beta0;
 /* TODO : code the Newton-Raphson method */
 return beta;
}
```

d'une autre pour calculer $x_\star$ et $\frac{d\theta}{dx}(x_\star)$:

```
void boundary_values(size_t nx, double* const x, double* const
     theta, double dx, double* xstar, double* dtheta_dxstar) {
 /* TODO */
}
```

Pour ouvrir un fichier, procédez de la sorte :

```
FILE* file;
file = fopen("pstar.txt", "w");
```

N'oubliez pas de le « fermer » à la fin de la fonction `main` :

```
fclose(file);
```

Vous devez également désallouer toute la mémoire allouée, par exemple pour les tableaux `x` et `theta` :

```
free(x);
free(theta);
```

Enfin, pour visualiser les résultats des modules pythons vous seront fournis. Vous devrez les adapter afin de générer des figures telle que la Fig. (1) par exemple où l'on compare la température, représenté en fonction de la masse, du modèle solaire polytropique au modèle solaire standard [4].

4. On peut trouver les fichiers de résultats ici : `https://www.sns.ias.edu/~jnb/SNdata/Export/BP2004/bp2004stdmodel.dat`