

# Modélisation non linéaire inverse de spectres stellaires par réseaux de neurones pour l'extraction des paramètres fondamentaux

Jean-François TROUILHET, Claude CATALA

Laboratoire d'Astrophysique UMR 5572 - Observatoire Midi-Pyrénées  
14, Avenue Édouard Belin, 31400 Toulouse, France  
jean-francois.trouilhet@obs-mip.fr, catala@obs-mip.fr

**Résumé** – Le but de cet article est de présenter une méthode permettant d'obtenir de façon automatique les paramètres fondamentaux d'étoiles à partir de leur spectre. Pour cela, nous allons proposer des solutions aux problèmes rencontrés et plus particulièrement l'utilisation de réseaux de neurones.

**Abstract** – We present a methodology to obtain the stars's fundamental parameters from their spectrum. To make this automatically, we need to solve many problems, we present some solutions and more especially the use of neural networks.

## 1 Introduction

Les bases de données constituées en spectroscopie stellaire représentent un volume d'information considérable. Afin d'optimiser leur exploitation, il est important de disposer d'outils permettant d'extraire automatiquement les informations principales, et ainsi, faciliter le travail de recherche d'étoile ou groupe d'étoiles répondant aux critères de l'utilisateur.

On montre que le rayonnement stellaire correspond à un spectre de raies. Ce spectre est très chaotique, il comprend plusieurs milliers de raies de morphologies différentes qui se chevauchent les unes les autres. En contre partie, chacune de ces raies est fonction des caractéristiques de l'étoile. Les trois principaux paramètres caractérisant l'étoile sont :

- $T_{eff}$  : Température effective.
- $\log g$  : Gravité de surface.
- $[M/H]$  : Métallicité.

Les relations entre les paramètres fondamentaux et le spectre sont extrêmement complexes [1]. Malgré la redondance de l'information disponible, le problème de l'estimation de ces paramètres à partir du spectre reste délicat. Plusieurs approches sont envisageables :

La première consiste à réaliser une classification des spectres stellaires selon le système MK (Morgan-Keenan) [2] par exemple, le premier bilan sur ces méthodes a été dressé en 1978 [3]. Cette approche consiste à affecter chaque étoile à une classe à partir de l'examen de son spectre. Compte tenu du nombre limité de classes, la quantification introduite peut amener à séparer artificiellement des étoiles de caractéristiques très proches tout en regroupant des étoiles de caractéristiques beaucoup plus éloignées.

La deuxième approche fait l'objet de cette communication. Elle respecte la nature continue des caractéristiques des étoiles en essayant de réaliser une modélisation inverse, c'est à dire fournir une estimation de la valeur des paramètres fondamentaux à partir de l'examen du spectre [10]. En contrepartie, si pour la classification, la génération de frontières de décision non linéaires ne pose pas de problème, il y a très peu de résultats sur la modélisation non linéaire exploitables directement pour le problème envisagé.

## 2 Extraction des paramètres fondamentaux

### 2.1 Introduction

Afin de nous placer dans des conditions idéales et faire abstraction des bruits de mesure, de la réponse instrumentale et de tout autre problème lié à l'observation des étoiles, nous avons choisi de travailler sur des spectres synthétiques.

Plusieurs personnes ont proposé une solution au problème de synthèse, nous nous sommes restreint à l'hypothèse d'équilibre thermodynamique local [1] sachant que c'est un premier pas vers le traitement des spectres d'étoiles plus chaudes et surtout de spectres réels.

De plus, afin de diminuer la difficulté, nous avons limité l'ensemble des paramètres variables à la température effective  $T_{eff}$  et à la gravité  $\log g$  les autres restant constants. Ceci ne limite en rien les résultats car l'estimation de chacun des paramètres peut être faite de façon indépendante des autres. Ce premier travail doit donc être considéré comme une étude de faisabilité.

La synthèse des spectres ne pose pas de problème particulier :

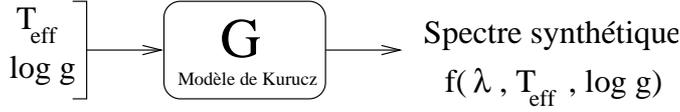


FIG. 1 – Problème direct : synthèse.

## 2.2 Modélisation

### 2.2.1 Recherche d'une structure adaptée

Nous pouvons considérer le problème de l'estimation des paramètres fondamentaux comme le problème inverse suivant :

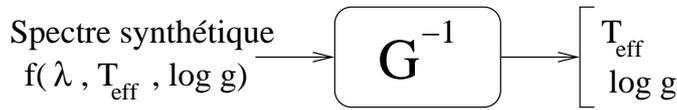


FIG. 2 – Problème inverse : Estimation des paramètres fondamentaux.

Si l'on adopte le principe de ne pas utiliser un modèle a priori qui conduirait à une solution extrêmement complexe, peu de méthodes de modélisation non linéaire génériques existent. C'est pour cela que nous avons pensé mettre à profit les caractéristiques non linéaires des réseaux de neurones [4]. Pour peu que le nombre de neurones soit suffisant, on montre qu'il existe des structures de réseaux de neurones permettant de réaliser un approximateur universel [5] [7] ce qui permet de résoudre le type de problème qui nous intéresse [6], Malheureusement, les capacités d'une telle structure sont limitées, notamment pour réaliser une multiplication, ce qui augmente la complexité du réseau et limite ses performances. L'exemple ci-dessous, bien que linéaire, illustre bien le problème :

Considérons la fonction  $f(x) = \alpha \cdot x + \beta$ , la connaissance de  $y = f(x)$  pour deux valeurs de  $x$  permet d'obtenir une solution au problème. Le calcul de la valeur du paramètre  $\alpha$ , donne :  $\alpha = \frac{y_1 - y_2}{x_1 - x_2}$ . Si l'on désire réaliser ce calcul avec un réseau de neurones conventionnel cela se fera au prix d'un nombre de neurones élevé et uniquement pour une certaine plage de variation des paramètres.

Un solution permettant de pallier ce problème sera soumise à la revue IEEE Trans. on Neural Network lorsque les résultats seront validés.

### 2.2.2 Calcul des poids du réseau

La figure de l'annexe n° 1 représente un réseau à  $M$  couches cachées.

Les poids synaptiques sont référencés par trois indices :  $w_{n^\circ \text{ étage}, n^\circ \text{ entrée}, n^\circ \text{ sortie}}$  par exemple, le poids synaptique entre l'entrée  $e_{0,2}$  et le premier neurone de la première couche cachée est nommé  $w_{0,2,1}$ . Les indices utilisés correspondent à :

- $i$  : Couche de départ.
- $j$  : Couche d'arrivée.
- $m_k$  : Couche  $n^\circ k$

Le critère retenu pour quantifier l'erreur globale en sortie (couche  $M + 1$ ) est :

$$\varepsilon = \sum_{m_{M+1}=1}^{N_{M+1}} |e_{M+1, m_{M+1}} - y_{m_{M+1}}|^2$$

où  $y_{m_{M+1}}$  est la valeur désirée de la sortie  $n^\circ m_{M+1}$ .

L'écriture d'un programme d'apprentissage ne se justifie pas dans le cas général. Il existe de très bons simulateurs comme SNNS [11] par exemple. Par contre, les modifications envisagées sur la structure du réseau nous ont amenés à réécrire un tel programme. Après une étude bibliographique des différents algorithmes d'apprentissage [12][13], nous avons, dans un premier temps, vérifié l'intérêt des modifications sur des réseaux de faible dimensions sous Matlab avec un apprentissage du type Levenberg-Marquardt. Puis, nous avons adopté le langage C pour des réseaux de taille plus conséquente avec un apprentissage basé sur des méthodes du type gradient conjugué [14].

Malgré un temps de convergence important, nous avons commencé à programmer un apprentissage par rétropropagation du gradient. La règle de modification des poids synaptiques est la suivante :  $w_{k,i,j} = w_{k,i,j} - \theta \cdot \frac{\partial \varepsilon}{\partial w_{k,i,j}}$  où  $\theta$  est le pas d'apprentissage.

Si la non linéarité du neurone  $n^\circ j$  de la couche  $k$  est notée  $S_{k,j}(x)$  et  $\tilde{e}_{k+1,j} = \sum_{i=0}^{N_k} w_{k,i,j} \cdot e_{k,i}$  la somme pondérée avant application de la non-linéarité du neurone d'indice  $k, j$ , on notera la dérivée  $\left. \frac{\partial S_{k,j}(x)}{\partial x} \right]_{x=\tilde{e}_{k+1,j}} = \Delta_{k,j}$ .

Si  $\Theta_j = e_{M+1,j} - y_j$  la différence entre la sortie  $n^\circ j$  et la sortie désirée ; on trouve les expressions de la dérivée de l'erreur par rapport aux poids à l'annexe n° 2

En général les non linéarités sont identiques pour tous les neurones, ceci permet de simplifier les expressions précédentes puisque toutes les dérivées sont identiques, seul l'argument est différent

Théoriquement, une couche cachée est suffisante, puisque toute fonction continue peut être approchée localement par un polynôme, la première couche permet de calculer le produit et la couche de sortie la somme. En pratique, tout comme en classification, le rajout de couches supplémentaires permet de diminuer le nombre de neurones sur les couches au détriment du temps d'apprentissage, il faut alors rechercher un compromis.

## 2.3 Information disponible

Les spectres synthétisés couvrent la plage  $\lambda = 380$  à  $680\text{nm}$  avec une résolution  $\frac{\lambda}{\Delta\lambda} = 45000$  ce qui correspond à 150000 points. Compte tenu de la puissance des calculateurs actuels et du coût calculatoire de l'algorithme d'apprentissage, il n'est pas envisageable de travailler directement sur les 150000 valeurs du spectre.

Compte tenu de l'étroite relation entre les paramètres fondamentaux et la classification MK, il existe un moyen sûr pour réduire le nombre d'entrées du réseau de neurones. Il s'agit de l'AFD, Analyse Factorielle Discriminante, qui s'applique aux problèmes de classification en mode supervisé. En effet le rôle de l'AFD est de rechercher et de limiter le nombre des axes discriminants c'est à dire les axes de projection pour lesquels la variance interclasses et la variance intraclasses sont respectivement maximale et minimale ce qui est le gage d'une classification optimisée mais non optimale car les projections restent du domaine de l'algèbre linéaire.

Ainsi, si le nouvel ensemble de variables permet une bonne classification, il est raisonnable de penser qu'elles conviendront pour le problème d'extraction des paramètres fondamentaux. Malheureusement l'AFD impose de manipuler plusieurs matrices de  $150000 \times 150000$  soit 180Go si les paramètres sont codés sur 8 octets. Cette valeur est largement au dessus des possibilités d'une station de travail, même si le principe est bon, il faudra affiner l'approche par une étude statistique sur la pertinence des paramètres pris un à un tel que le critère de Fisher [9][8].

## 3 Conclusions

Les expérimentations sont en cours et les résultats n'ont malheureusement pas pu être intégrés dans cette communication. Entre temps, nous avons eu connaissance des résultats obtenus par C. Bailer-Jones [10] qui montrent l'intérêt d'une telle approche sur des spectres très basse résolution.

Dans notre cas, compte tenu du nombre important de points en entrée du réseau, les temps de convergence de l'algorithme d'apprentissage sont très importants même pour un apprentissage du type gradient conjugué. Aussi, nous attendons beaucoup de la modification envisagée sur la structure du réseau utilisé.

## Références

- [1] Robert L. Kurucz *Model Atmospheres for G, F, A, B and O Stars*. The Astrophysical Journal Supplement Series 40, pp 1-340, May 1979.
- [2] Coryn Bailer-Jones *Neural Network Classification of Stellar Spectra*. Institute of Astronomy and Emmanuel College Cambridge, December 1996.
- [3] T. Schmidt-Kaler *Automatic Spectral Classification. Present State and Future Developments*. IAU Colloquium 47 on Spectral Classification of the future, 11-15 July 1978, vol 9, pp 285-286, 545-560, 1979.
- [4] S. Haykin *Neural Networks : a comprehensive foundation*. Prentice Hall, 1979.
- [5] K. Hornik, M. Stinchcombe, H. White *Multilayer feedforward networks are universal approximators*. Neural Networks, pp 551-560, 1990.
- [6] L. Jackowska-Strumillo, J. Sokolowski, A. Zochowski *The topological derivative method and artificial neural networks for numerical solution of shape inverse problem*. INRIA Lorraine, n° 3739, June 1999.
- [7] G. Cybenko *Approximation by superposition of sigmoidal functions*. Mathematics of Control, Signal and Systems, vol 2, pp 303-314, 1989.
- [8] Fisher, R.A. *The Statistical Utilization of Multiple Measurements*. Annals of Eugenics, vol 8, pp 376-386, 1938.
- [9] Fisher, R.A. *The Use of Multiple Measurements in Taxonomic Problems* Annals of Eugenics, vol 7, pp 179-188, 1936.
- [10] Coryn Bailer-Jones *Stellar parameters from very low resolution spectra and medium band filters  $T_{\text{eff}}$ ,  $\log g$  and  $[M/H]$  using neural networks*. Astronomy and Astrophysics, vol 357, pp 197-205, 2000.
- [11] Institute for Parallel and Distributed High-Performance Systems (IPVR), University of Stuttgart *Stuttgart Neural Network Simulator*. <http://www-ra.informatik.uni-tuebingen.de/SNNS/>
- [12] A.R. Webb, David Lowe, M.D. Bedworth *A Comparison of Nonlinear Optimisation Strategies for Feed-Forward Adaptive Layered Networks*. Royal Signals and Radar Establishment, Memorandum 4157, May 1988.
- [13] W. Schiffmann, M. Joost, R. Werner *Optimization of the Backpropagation Algorithm for Training Multilayer Perceptrons*. Institute of physics, University of Koblenz, September 29, 1994.
- [14] Martin F. Moller *A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning*. Computer Science Department, University of Aarhus, Denmark, November 13, 1990.
- [15] D. Katz, C. Soubiran, R. Cayrel, M. Adda, R. Cautain *On-line determination of stellar atmospheric parameters  $T_{\text{eff}}$ ,  $\log g$ ,  $[F_e/H]$  from ELODIE echelle spectra*. Astronomy and Astrophysics n° 338, pp 151-160, 1998
- [16] Guy Demoment, Jérôme Idier, Jean-François Giovannelli, Ali Mohammad-Djafari. *Problèmes inverses en traitement du signal et de l'image* Techniques de l'ingénieur - Traité d'électronique.
- [17] S. Chen, S.A. Billings, W. Luo. *Orthogonal least squares methods and their application to non-linear system identification*. Int. J. Control, vol 50, n° 5, pp 1873-1896, 1989.
- [18] Qinghua Zhang. *Using Wavelet Network in Nonparametric Estimation*. Rapport interne INRIA n° 2321, June, 1994.

## Annexe n° 1 : Schéma du réseau neuromimétique

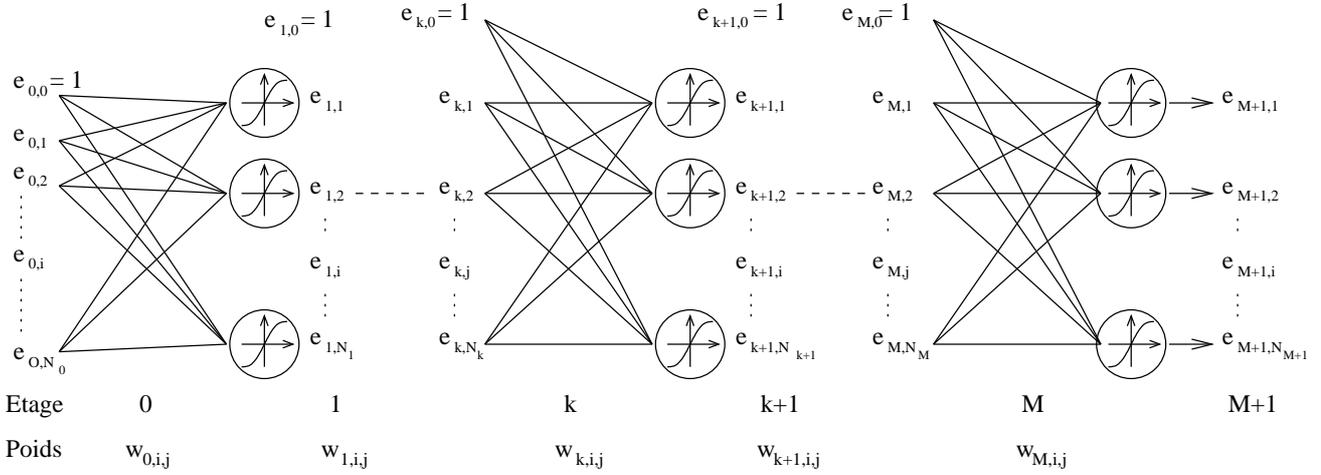


FIG. 3 – Réseau

## Annexe n° 2 : Expressions des dérivées de l'erreur

– pour la couche  $n^\circ M$  :

$$\frac{\partial \varepsilon}{\partial w_{M,i,j}} = 2 \cdot e_{M,i} \cdot \Delta_{M,j} \cdot \Theta_j$$

– pour la couche  $n^\circ M - 1$  :

$$\frac{\partial \varepsilon}{\partial w_{M-1,i,j}} = 2 \cdot e_{M-1,i} \cdot \Delta_{M-1,j} \cdot \left\{ \sum_{m_{M+1}=1}^{N_{M+1}} \Theta_{m_{M+1}} \cdot \Delta_{M,m_{M+1}} \cdot w_{M,j,m_{M+1}} \right\}$$

– pour la couche  $n^\circ M - 2$  :

$$\frac{\partial \varepsilon}{\partial w_{M-1,i,j}} = 2 \cdot e_{M-2,i} \cdot \Delta_{M-2,j} \cdot \left\{ \begin{array}{l} \sum_{m_{M+1}=1}^{N_{M+1}} \Theta_{m_{M+1}} \cdot \Delta_{M,m_{M+1}} \\ \cdot \left\{ \sum_{m_M=0}^{N_M} \Delta_{M-1,m_M} \cdot w_{M-1,j,m_M} \cdot w_{M,m_M,m_{M+1}} \right\} \end{array} \right\}_2$$

– Par généralisation, pour la couche  $n^\circ k$  :

$$\frac{\partial \varepsilon}{\partial w_{k,i,j}} = 2 \cdot e_{k,i} \cdot \Delta_{k,j} \cdot \left\{ \begin{array}{l} \sum_{m_{M+1}=1}^{N_{M+1}} \Theta_{m_{M+1}} \cdot \Delta_{M,m_{M+1}} \\ \cdot \left\{ \sum_{m_M=0}^{N_M} w_{M,m_M,m_{M+1}} \cdot \dots \right. \\ \cdot \Delta_{l,m_{l+1}} \cdot \left\{ \sum_{m_l=0}^{N_l} w_{l,m_l,m_{l+1}} \cdot \dots \right. \\ \cdot \Delta_{k+2,m_{k+3}} \cdot \left\{ \sum_{m_{k+2}=0}^{N_{k+2}} \Delta_{k+1,m_{k+2}} \cdot w_{k+1,j,m_{k+2}} \cdot w_{k+2,m_{k+2},m_{k+3}} \right\} \dots \end{array} \right\}_{M-k} \dots \right\}_{M+2-l} \right\}_2$$