



JMMC-MIN-2300-0003

Date : 05/03/2007

JMMC

REUNION DES PREMIERS FUTURS UTILISATEURS DE LITPRO

CRAL-Observatoire, 26-27 février 2007

Participants:

Armando Domiciano
Nicolas Nardetto
Michel Tallon

Florentin Millour
Romain Petrov
IsabelleTallon-Bosc

Denis Mourard (26.02)

1 Programme des 2 journées

- Tour de table des besoins et objectifs de chacun quant à l'utilisation du logiciel et l'apport souhaité de la réunion.
- Tutorial : démonstration de LITpro sur un exemple de fit et démonstration du GUI développé par Guillaume Mella au LAOG.
- Démonstration de la manière d'implémenter une fonction utilisateur et de compléter le fichier descripteur : interfaçage du modèle de céphéide de Nicolas avec LITpro.
- Lecture des fichiers de données AMBER de Florentin => implantation de la SED dans le code et tentative d'ajustement du modèle de Florentin de γ 2Vel
- Ecriture d'une « road map » : liste des tâches à effectuer

Addendum : non discuté en réunion

- comment garantir une pérennité au logiciel

2 Points vus

2.1 Introduction

Le premier objectif de la réunion est rappelé : expliquer LITpro, l'architecture qui le sous-tend et son fonctionnement, en illustrant son utilisation avec un exemple de fit. Le second objectif est de pouvoir répondre le plus possible aux questions et besoins des participants, qui portent sur :

- l'implantation d'un modèle dit « utilisateur »,
- le fit de données multichromatiques, du type AMBER ou VEGA,
- le format Oifits des données (comment ce dernier est lu actuellement par LITpro, comment doit-il évoluer, questions qui pourraient aider à la réflexion menée pour les données VEGA).

2.2 Tutorial : LITpro et GUI

LITpro est décrit à partir d'un exemple de fit, l'un de ceux présentés à l'école de Goutelas le 14 juin dernier, sur une binaire de calibration.

A été rédigé un papier pour l'école qui sera mis prochainement sur la page Wiki du groupe (après entrée des dernières corrections) et qui décrit précisément le tutorial.

L'interface graphique écrite en JAVA par Guillaume est montrée : elle lit le fichier descripteur (transformé en fichier xml) et lance le fit yorick. Le GUI utilise les fonctionnalités de TOPCAT.

2.3 Implantation du modèle de Nicolas

A partir du modèle de céphéide de Nicolas est écrit en séance le fichier descripteur : opération simple qui permet de sélectionner les données, le modèle, définir les paramètres de ce dernier, et leurs caractéristiques (bornes, fixé ou libre, ...).

La fonction modèle, écrite en Yorick par Nicolas, n'a pu être testée du point de vue de l'ajustement, faute de données multichromatiques à fitter. Il a été néanmoins mis en évidence qu'elle devrait gagner en rapidité de calcul et en robustesse.

2.4 Lecture des données de Florentin

Avant la réunion avait été entreprise la lecture des données AMBER envoyées par Florentin. La méthode choisie est de lire les différents fichiers fits produits par le pipeline d'AMBER et d'organiser ces données sous forme d'une « hash table » yorick ayant la même structure que celle produite lors d'une étape de lecture des données oifits. La « hash table » est ensuite sauvegardée dans un fichier « yhdf » (yorick hierarchical data file). Le code a été modifié pour pouvoir traiter ces fichiers « yhdf ». Le programme qui produit les fichiers « yhdf » à partir des données AMBER s'appuie sur des fonctions de Florentin modifiées, en particulier pour le calcul des lignes de base.

Cette méthode (conversion données non-oifits vers « yhdf ») peut maintenant facilement être utilisée pour d'autres types de données.

Cette fonctionnalité n'a pas abouti avant la réunion, à cause de l'incertitude sur la définition des variables telles que VISdiff, VISclos, etc. mais elle a pu aboutir en séance : la prise en compte de la SED est maintenant effective.

Elle n'a pu être testée avec la fonction utilisateur de Florentin faute de temps mais aussi parce que le fichier descripteur écrit ne pouvait être lu : les paramètres ne peuvent pour l'instant pas passer sous forme de vecteur.

2.5 Liste des tâches identifiées par ordre de priorité décroissante

En comité plus restreint (Armando, Romain et Denis étant partis) a été dressée une liste des actions à mener, compte tenu des discussions de la réunion. Le vocabulaire employé ne sera sans doute compréhensible que par les participants de la réunion ; ceux-ci devant être les seuls acteurs, ce n'est pas trop gênant.

Il a été également décidé de nous tenir au courant de l'avancement par le biais de réunions téléphoniques régulières, par ex. tous les 2-3 mois (→ la première de ce genre, 1ère quinzaine de mai ? –date à fixer-).

Liste des tâches (toutes urgentes... mais classées de super-urgentes à urgentes tout court...)

- Mettre à disposition le logiciel.
 - o si urgence : cvs (le logiciel a été diffusé de cette façon lors de la réunion).
 - o faire un package épuré
 - numéro version (correspondant à un tag du module cvs)
 - ftp ou page web.
 - o pserver.
- Amélioration du jargon pour plus de clarté
 - o changer dans le code SED -> flux uniquement dans les CR
 - o residuals are not residuals
- Faire marcher / tester la SED.
- Fichiers descripteurs
 - o nouveau parseur avec mémorisation/relecture des résultats.
 - models -> modeling(s)
 - targets -> data
 - functions -> model, model = set of functions.
 - residuals: changer logique.
 - pas de default
 - commenté -> skipped
 - noms des fonctions appelées explicites.
 - data+model+residuals = "group"
 - o E/S xml.
- Finaliser importateur de données AMBER.
 - o importation SED.
 - o contrôler échantillonnage SED = f(lambda).
- Importer différents formats de fichiers spectre. (→ Nicolas + Florentin)
 - o reconnaissance automatique du format à partir du fichier.
 - o => génération de la bonne hash table.
- Paramètres vectoriels.
- LA DOC !!!
 - o premier jet CRAL.
 - o collaboration indispensable ensuite.

- Visibilité différentielle
 - o Spécifier une classe visibilité différentielle où la méthode de calcul du canal spectral de référence est définie précisément
 - o Devrait idéalement permettre différentes méthodes (AMBER, VEGA, etc.)
- Développeurs ?
 - o lorsque prise en charge d'une fonctionnalité ou d'une partie du code.
 - o gestion des bugs (→ Nicolas + Florentin).
- Simulation de données -> oifits.
- DB (Data Blocks) avec drivers.
 - o calcul des résidus
 - o simulation des données.
 - o lecture des données.
 - o contrôle des données.
 - o ... (voir code)
- Fit sur données hétérogènes : fichier.yhd généré préalablement.
 - o publier la hash table.
 - o nouvelle classe de DB si type de données non prévu.
- Flags
 - o fonction pour ré-appliquer les flags.
 - o fonction pour modifier les flags (→ Florentin)
- Moteurs de fit supplémentaires à l'OCA ?
 - o interfaçage avec LITpro.
- Parallélisation ?

Parallèlement à ces actions : poursuite et finalisation du GUI.

3 Addendum à la réunion

LITpro a été conçu de manière à ce que son utilisation soit relativement facile, ainsi que son enrichissement. L'intérêt de chacun est de pouvoir bénéficier d'un logiciel robuste qui réponde certes à ses besoins mais qui bénéficie aussi de l'expérience des autres (autres fonctionnalités, remontée de bugs,...).

J'ai suffisamment confiance dans la motivation « scientifique » et le respect mutuel des uns et des autres pour que la collaboration qui a vu concrètement le jour avec cette réunion soit fructueuse et bénéfique au sein du groupe.

Nous n'avons pas discuté des modalités à suivre pour ouvrir le logiciel à l'extérieur : est-ce qu'une personne peut utiliser LITpro pour ses travaux en ajoutant ou non une fonctionnalité et oublier de faire connaître ou remonter l'info. aux membres du groupe original ?

Nous n'avons pu mener cette réflexion durant ces 2 journées, l'accent ayant été mis à forte raison sur le tutorial et l'implantation du modèle chromatique. Peut-être pourrions-nous la mener lors d'une prochaine rencontre, ou par courrier électronique.