

# GUIDE D'UTILISATION DES ROUTINES

## v3.4, 12/nov/2006, BC

### **Lancer une simulation**

Se placer dans le répertoire `simu_JUNO` (le répertoire qui contient le répertoire des routines de simulation `pro_v3.4/`).

Lancer la commande

```
> ./idl_juno_v34
```

NB: Cela revient à saisir la commande

```
> idl pro_v3.4/JUNO_batch.pro
```

Le fichier batch `pro_v3.4/JUNO_batch.pro` contient les commandes nécessaires à la compilation et au lancement de la simulation définie par la procédure `JUNO_init`.

### **Syntaxe de la routine d'initialisation**

La syntaxe de `JUNO_init` est la suivante :

- définition de la longitude (+ inclinaison) jovicentrique du Soleil (à t=0)

```
SET_LOCAL_TIME, longitude=0., inclination=0., parameters=params
```

- durée totale et pas de la simulation

```
SET_SIMULATION_TIME, simul_length=480., simul_step=.25,  
parameters=params
```

- Observateur fixe ou mobile

```
SET_OBSERVER, name="fixe", motion=0b, obs_params=[3000.,0.,351.], $  
parameters=params,verbose=verbose  
; ici, obs_params : position fixe de l'observateur  
;                  = [distance, latitude, longitude] (a t=0?)
```

ou bien

```
SET_OBSERVER, name="orbiter", motion=1b, obs_params=  
[20.31,6.5,351.,90.,90.],$  
parameters=params,verbose=verbose  
; ici, obs_params : definition de l'orbite et de la position initiale  
de  
; l'observateur  
;                  = [semi_major_axis, semi_minor_axis, apoapsis longitude,  
;                  orbit inclinaison, initial phase]  
; NB initial phase = angle de phase sur l'orbite a t=0 par rapport au  
centre  
; de l'ellipse avec phase=0 a l'apojove
```

- Definition des fréquences d'observation

```
SET_FREQUENCY, fmin=.4, fmax=45., fstep=0.2, parameters=params
```

- Ajout des sources

```
ADD_SOURCE, name="Ovale Nord", directory="OvaleN", active_LO=
[0,360,3], $
    verbose=verbose,parameters=params,cone_app=70., cone_wid=.2,
shape=0.2
ADD_SOURCE, name="Ovale Sud", directory="OvaleS", active_LO=
[0,360,3], $
    verbose=verbose,parameters=params,cone_app=70., cone_wid=.2,
shape=0.2

ADD_SOURCE, name="Io Flux Tube", directory="IFT", /satellite, $
    pos_t0=[5.95, 105.], lag=[30.,-30.], shape=[0.1,0.1], cone_app=
[70.,70.], $
    cone_wid=[.2,.2], parameters=params,verbose=verbose,pole=[1,-1]
```

Les paramètres de ADD\_SOURCE sont les suivants :

- cas "source aurorale" :

+ name :	nom
+ directory :	répertoire où sont rangées les données précalculées
+ active_LO :	secteur de longitude actif [début, fin, pas] en degrés
+ active_LT :	secteur de temps local [début, fin, pas] en degrés (pas encore implémenté).
+ cone_app :	ouverture du cône d'émission en degrés (une valeur)
+ cone_wid :	épaisseur du cône d'émission en degrés (une valeur)
+ shape :	sélection du type d'émission

- cas "source satellite" :

+ name :	nom
+ directory :	répertoire où sont rangées les données précalculées
+ satellite :	doit être à 1 pour sélectionner ce type de source
+ pos_t0 :	position du satellite à t=0 (distance, longitude) en degrés
+ lag :	décalage en degrés entre la longitude active et celle du satellite (Nord, -Sud)
+ shape :	sélection du type d'émission (Nord, Sud)
+ cone_app :	ouverture du cône d'émission en degrés (Nord, Sud)
+ cone_wid :	épaisseur du cône d'émission en degrés (Nord, Sud)
+ pole :	pole de l'émission. doit être [1,-1] dans le cas décrit ici où on donne d'abord l'info Nord puis l'info Sud. [1= nord, -1=sud]

- tous les cas :

+ parameters :	la structure contenant les paramètres de la simulation, passée depuis la routine principale, <code>juno.pro</code> .
+ verbose=verbose :	ne pas supprimer (permet d'avoir des info complémentaires en lançant la simulation avec <code>juno,/verbose</code> ; l'exécution est alors environ 20-30% plus lente).
+ gradb_test :	mot clé permettant d'activer le test sur le gradient de B au

piéd de la ligne de champ active. Ce test ne semble pas fonctionner correctement pour l'instant.

les paramètres de la simulation sont enregistrés dans la structure `params` dont les différents éléments sont remplis par les différentes routines.

### ***Modifications par rapport aux versions précédentes***

Les common blocks ont été supprimés au profit du passage d'une structure contenant toutes les infos nécessaires au déroulement de la simulation. Les différentes sources ont été homogénéisées. Elles sont maintenant toutes traitées de la même manière (plus de distinguo source N, source S dans le cas satellite, par exemple). Les mots clés permettant de définir la manière de coder les intensités sur le spectre dynamique final ont été supprimés. On enregistre dans le tableau `spdyn` tous les spectres dynamiques. Les dimensions de ce tableau, dans la procédure `JUNO` sont donc : Nombre de fréquences x Nombres de pas de temps x Nombre de Sources. Le nombre de Sources est défini par le paramètre `Active_IO` ou `Active_IT` dans le cas d'une source oval aurorale, et il vaut 2 dans le cas d'une source satellite.

Le gain en temps de calcul par rapport est d'environ x25 sur Charybde, grâce à deux optimisations :

- les calculs sont faits en parallèle sur toutes les sources (gain de ~x4.5). Il y avait auparavant un appel de la routine de calcul de l'émission (`field_line_emission`) par source. Cette routine a été modifiée pour traiter tout d'un bloc.
- les données magnétiques précalculées sont chargées une fois pour toutes au moment de l'initialisation (gain jusqu'à ~x6). Cela est bénéfique surtout dans le cas des ovales auroraux, où la ligne de champs d'une longitude donnée pouvait être chargée de nombreuses fois au cours de la boucle sur le temps.

### ***A faire.***

- + l'option `Active_IT` pour sélectionner un secteur de temps local actif dans le cas des ovales auroraux. Pour mettre en place cette option, voir les commentaires dans `add_Source.pro`.
- + vérifier le test sur le gradient de B (dans `field_line_emission.pro`)
- + randomisation
- + variation linéaire de l'ouverture du cone d'émission en fonction de la fréquence.