

Conseils de bonne utilisation de Gaussian sur les centres de calcul nationaux

Décembre 2015

Utilisation optimale du logiciel Gaussian à l'IDRIS

Le logiciel Gaussian est installé à l'IDRIS sur la machine généraliste Ada, et parallélisé en OpenMP. Ce document rassemble les règles de bonnes pratiques permettant d'obtenir les performances les meilleures sur cette machine, en complément de la documentation de notre site : <http://www.idris.fr/su/chimie.html>.

1. Script de lancement Loadleveller

Le mode d'exécution OpenMP et le nombre de cœurs de calcul sont choisis respectivement par les directives :

@ job_type = serial

@ parallel_threads = *n*, avec *n* compris entre 1 et 32 sur Ada

2. Fichier d'entrée Gaussian

Afin d'utiliser au mieux la machine, il existe deux paramètres Gaussian gérant l'utilisation des cœurs de calcul et de la mémoire dynamique : **%NProcShared** et **%Mem**, à placer en tête du fichier d'entrée.

%NProcShared doit être égal au nombre de cœurs de calcul réservés (*n*).

%Mem détermine la quantité de mémoire virtuelle que Gaussian va utiliser pour les allocations dynamiques. Elle dépend aussi du nombre de cœurs de calcul que l'on souhaite utiliser. Par défaut, la valeur est de 256 mb, ce qui convient uniquement aux systèmes de petite taille, et pour un seul cœur de calcul. Il est recommandé de valoriser ce paramètre en utilisant la formule empirique suivante :

%Mem = (*n* × 3,5) – (*n* + 1) gb, soit par exemple pour 16 cœurs : **%Mem = 39gb**

Il est également conseillé d'ajouter « **#P** » devant la ligne de paramètres de calcul, afin d'obtenir des informations supplémentaires (notamment lors de l'entrée dans les différents « Links »).

3. Messages d'erreur et paramètres inadaptés

- Erreurs sur **%NProcShared**

- **%NProcShared** > *n*

- ▶ Plantage avec le message :

- Inconsistency in NProc: NTCur= 32 but real number of threads= 4*

- ▶ Lancement d'un nombre supérieur de *threads* (performances fortement dégradées)

- **%NProcShared** < *n*

- ▶ Les cœurs restant ne seront pas utilisés pour le calcul, mais restent comptabilisés (« facture temps » = *t* × *n*)

- Le paramètre **%NProc** est obsolète.

Exemples de sorties où **%NProcShared** n'est pas adapté au calcul :

PrsmSu: requested number of processors reduced to: 10 ShMem 1 Linda.

CalDSu: **requested number of processors reduced to: 28 ShMem** 1 Linda.
 PrsmSu: **requested number of processors reduced to: 7 ShMem** 1 Linda.
 DoSDTr: **NPSUse= 13**
 JobTyp=1 Pass 1: l= 1 to 5 **NPSUse= 1** ParTrn=F ParDer=F DoDerP=T.

• Erreurs sur %Mem

• %Mem trop élevé

- ▶ Plantage immédiat avec le message :
Insuffisant virtual memory

• %Mem faible ou pas adapté au type de calcul

- ▶ Plantage au bout d'une phase de calcul, avec un message explicite de type :
Insufficient memory for ...
- ▶ Réduction du nombre de cœurs utilisés pour les calculs, par le message :
GetIJB would need an additional 45990731 words of memory to use all 32 processors.

4. Utilisation spécifique

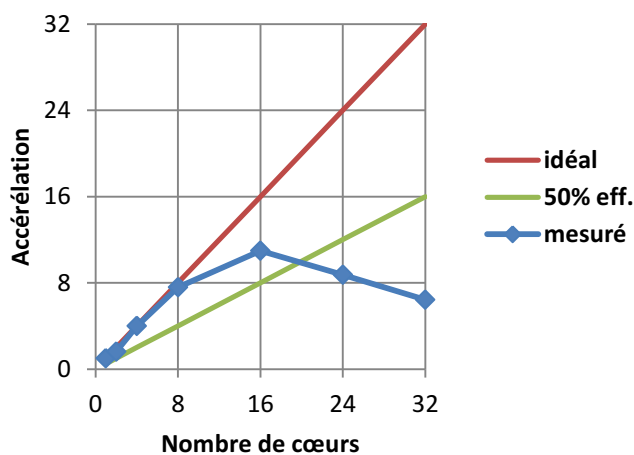
Dans de **rares** cas, l'utilisation des nœuds à mémoire large peut être nécessaire. Ces nœuds sont accessibles avec la directive Loadleveller :

@ as_limit = x Gb, avec $x = 7,0 \text{ Gb} \times n$

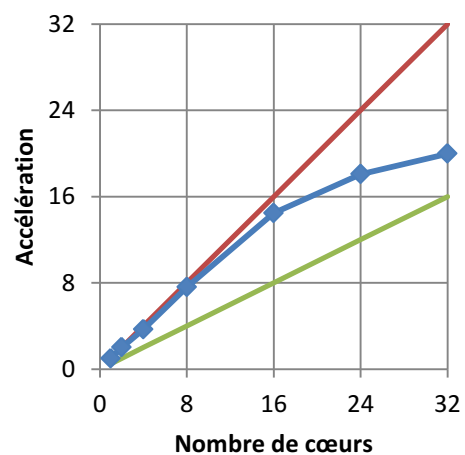
Dans la formule calculant %Mem, il faut alors remplacer la valeur 3,5 par 7.

5. Performances

Les graphiques suivants présentent les performances de deux systèmes en fonction du nombre de cœurs utilisés sur la machine Ada :



Test Gaussian n° 397



Supercellule de zéolithe (MOR)

Le système traité et le type de calcul ont une incidence directe sur les performances. Ainsi, il est recommandé de faire des tests de performances pour connaître le nombre de cœurs maximal à utiliser (efficacité > 50%).

Utilisation de Gaussian au TGCC

Le logiciel Gaussian est installé au TGCC sur la machine Curie, et parallélisé en OpenMP. Ce document rassemble les règles de bonnes pratiques permettant d'obtenir de bonnes performances sur cette machine.

Script de soumission

La bonne parallélisation est à l'utilisation de tous les coeurs d'un (ou deux) socket(s) pour Gaussian (suivant les "Links").

Néanmoins il y a deux contraintes supplémentaires que sont l'utilisation mémoire (%MEM) et la taille du fichier temporaire (MaxDisk) exprimées en Giga-Word (= Giga Octet). Les raisons sont décrites plus loin. Voici leur influence:

Condition	NB_SOCKET	GAUSS_SCRDIR
Max(%Mem,MaxDisk) <= 28GW	1	/tmp
28GW < Max(%Mem,MaxDisk) <= 56GW	2	/tmp
MaxDisk >= 56GW	1	\$SCRATCHDIR

soit:

```
#!/bin/bash
#MSUB -n 1 #
#MSUB -c <NB_CORES> #
#MSUB -T 86400 # durée de 24h

# Load Gaussian
module load gaussian
source ${g09root}/g09/bsd/g09.profile

# Set the number of threads
export OMPI_NUM_THREADS=<NB_CORES>

# Execute gaussian
ccc_mprun g09 < input.com > output.log
```

Avec
<NB_CORES> = 8 (ou 16 si 2 sockets) sur Curie standard
<NB_CORES> = 8 (ou 16 si 2 sockets) sur Curie xlarge
<NB_CORES> = 8 (ou 16 si 2 sockets) sur Airain standard
<NB_CORES> = 8 (ou 16 si 2 sockets) sur Airain large
<NB_CORES> = 10 (ou 20 si 2 sockets) sur Airain ivybridge

Fichier d'entrée Gaussian

Dans le fichier d'entrée de Gaussian deux paramètres sont importants:

- %NprocShared le nombre de threads OpenMP utilisé par Gaussian.

Il doit être égal au nombre de cœurs de calcul réservés (<NB_CORES>).

- %Mem détermine la quantité de mémoire virtuelle utilisée par Gaussian pour ses allocations.

Sur Curie, nous avons ratio de 3,5 GB par coeur utilisé soit la formule suivante:

%Mem = (n × 3,5) gb, soit par exemple pour 8 cœurs : %Mem = 28gb et pour 16cœurs : %Mem = 56gb

- \$GAUSS_SCRDIR et "-R- Maxdisk=<size>" déterminent l'emplacement du fichier temporaire et sa taille.

Par défaut, il est recommandé d'utiliser le /tmp est local au noeud qui est le SSD du noeud avec le même ratio de 3,5gb par coeur.

Cas spécifique

Fichier temporaire nécessaire de taille trop importante

Dans ce cas exporter

```
export GAUSS_SCRDIR=$SCRATCHDIR
```

Attention: Cet export ralentira votre exécution