

一般的非経験分子軌道計算プログラム Gaussian98 の使い方

システム運用掛

一般的非経験分子軌道計算プログラム Gaussian98 を超並列型スーパーコンピューター SR8000/MPP 及びベクトル並列型スーパーコンピューター SR8000/128 (バルク利用システム) 上のプログラムライブラリーとして 2001 年 4 月 2 日に公開しました。本稿では Gaussian98 を実行するためのジョブの作成と使用方法について簡単に説明します。但し、Gaussian98 自体の内容や計算方法、入力ファイルの書き方に関しては触れませんので詳細は以下に示す文献を参照して下さい。

本センターにおける Gaussian98 のバージョンは次の通りです。

Gaussian98 Revision A.9

本ライブラリーは 1 ノードによる逐次実行版 (並列版ではありません) で、プログラムは要素並列化されており、32bit モードで動作します。また、ソースプログラムは非公開です。

参考文献

- [1] Gaussian98 User's Reference Second Edition (Manual Version: 6.1 Jan.1999)
- [2] Gaussian98 Programmer's Reference (Manual Version: 6.0 Aug.1998)
- [3] 電子構造論による化学の探求 第二版
James B.Foressman、Eleen Frisch 共著 田崎健三 訳

Gaussian Inc. (ガウシアン社) から上記のマニュアルと解説書が出版されています。[1]、[2] は英文です。[2] はソースプログラムの解説書なので一般利用者には必要ないでしょう。[3] は入門用解説書です。これらは本センタープログラム相談室に常備しています。

また、以下の Web ページから Gaussian98 に関する様々な情報を得ることができます。

<http://www.gaussian.com/>

なお、Gaussian98 を使用して得られた計算結果や成果を公表する場合には、以下の Web ページに従って文献の引用を行うようにして下さい。

<http://www.gaussian.com/citation.htm>

1. スクリプトファイルの作成

Gaussian98 をバッチジョブ (NQS) で実行する方法を紹介します。バッチジョブで実行

するためには環境変数の設定や入出力ファイルを記述するスクリプトファイルを作成する必要があります。以下にスクリプトファイルの例を挙げ説明します。

```
#!/bin/csh                ... (1) シェルの宣言
#@$-q parallel           ... (2) キューの選択
#@$-N 1                  ... (3) ノード数
#@$-IT 00:30:00         ... (4) 経過制限時間 (ここでは 30 分)
#@$-IM 2GB               ... (5) メモリーサイズ
#@$-ls 192MB            ... (6) スタックサイズ
setenv g98root /usr/local ... (7) 環境設定
source $g98root/g98/bsd/g98.login ...      "
setenv GAUSS_SCRDIR /var/tmp ... (8) スクラッチディレクトリー
cd gaussian/test        ... (9) 入力ファイルの場所
g98 < test178.com > test178.log ... (10) プログラムの起動
```

- (1) スクリプトファイルを C シェル「/bin/csh」で記述することを宣言します。(本スクリプトでは setenv、source 等 C シェルのコマンドを使用しています。)
- (2) パイプキューは「parallel」を指定します。これはプログラムが要素並列化されているため、1 ノード占有可能なバッチキュー P001 で実行させるためです。
- (3) 並列版ではないためノード数は「1」ノード固定です。
- (4) ジョブの実行時間を制限します。計画停止時刻(qstat コマンドで確認)が迫っているときはこれを残り時間以内に設定しないとジョブが実行しません。
- (5) ジョブの実行に必要なメモリーの大きさです。入力ファイルの %Mem で指定した値より十分大きい値を設定する必要があります。
- (6) スタックサイズは通常「192MB」として下さい。
- (7) コマンドパスなど各種環境変数を設定します。
- (8) スクラッチファイルを置くディレクトリーを指定します。通常は「/var/tmp」です。
- (9) 入力ファイルのある場所までディレクトリーを移動します。
- (10) プログラム g98 を起動します。入力ファイル及び出力ファイルを以下のようにリダイレクションで指定することができます。

```
g98 < 入力ファイル > 出力ファイル
```

“ > 出力ファイル ” を省略すると実行結果はジョブの標準出力ファイルに出力します。

(7)、(8)の設定及び unlimit コマンドで各種メモリーサイズを最大にしておくことによりインタラクティブ環境 (TSS) の mpp-p または bulk-p でも実行できます。但し、スタックサイズ不足で実行できないことがありますので特別な事情を除いてはバッチジョブで実行するようにして下さい。

2. プログラムの実行

スクリプトファイルが完成したらファイル (以下の例では job.csh) に保存し、バッチシステムにジョブを投入します。ジョブの投入は以下のように qsub コマンドで行います。実行状況は qstat コマンドで確認して下さい。

```
% qsub job.csh
```

```
Request 4575.mpp-s.cc.u-tokyo.ac.jp submitted to queue: parallel.
```

```
% qstat
```

```
2001/06/08 (Fri) 19:09:09: BATCHPIPE REQUESTS on mpp-bt.cc.u-tokyo.ac.jp
```

```
NQS schedule stop time : 2001/06/29 (Fri) 08:30:00 (Remain: 493h 20m 51s)
```

REQUEST	NAME	OWNER	QUEUE	PRI	NICE	CPU	MEM	STATE
4575.mpp-s.cc	job.csh	a30000	P001	31	20	UNLIM	13824	RUNNING

1. のスクリプトファイルの場合、ジョブが終了すると以下のファイルが作成されます。エラーメッセージが出ていないか各ファイルを確認して下さい。

```
job.csh.o4575    ... ジョブの標準出力  
job.csh.e4575    ... ジョブの標準エラー出力  
test178.log      ... Gaussian98 の出力ファイル
```

エラーがなければ出力ファイルに計算結果が出力され、標準出力及び標準エラー出力のファイルは空になります。

ジョブをキャンセルする場合には以下のようにします。

```
% qdel -k 4575.mpp-s.cc.u-tokyo.ac.jp
```

```
Request 4575.mpp-s.cc.u-tokyo.ac.jp is running, and has been signalled.
```

(mpp-s からキャンセルする場合は qdel -k 4575 に省略可)

3. スクラッチファイルについて

Gaussian98 は実行時にスクラッチファイルと呼ばれる巨大な一時ファイル(サフィックスが.rwf、.scr、.chk等のファイル)を作成します。しかもこれらのファイルはジョブが正常終了しなかった場合、消されずにディスク上に残ります。通常、スクラッチファイルは/var/tmp ディレクトリー*を使用しますが、大きなジョブが同時に実行したときなど場合によってはディスク容量不足によるエラーとなることがあります。必要に応じて GAUSS_SCRDIR を変更して/home や/short ディレクトリー**を使用するようにして下さい。なお、計算によってはスクラッチファイルが 20GB 程度にまで達することがありますが、本 Gaussian98 ではこの大きさのファイルを扱えずエラーとなります。この場合はアルゴリズムを変更する等で対応して下さい。アルゴリズムについては参考文献[1]又は Web ページ <http://www.gaussian.com/> を参照して下さい。

* /var/tmp ディレクトリーはスクラッチファイルを残さないように実行中に作成したファイルをジョブ終了時に自動的に削除します。ジョブ終了後も必要なファイルは別のディレクトリーに出力して下さい。

**/home、/short ディレクトリーに作成されたスクラッチファイルはジョブ終了後も保存されます。

例 setenv GAUSS_SCRDIR /short/mpp-bt/a30000

/short は保存期間と容量によって課金されますので不要なファイルは必ず削除するよう御注意下さい。また、/home を使用する場合はファイル使用量上限値にも注意して下さい。

4. 例題

Gaussian98 が用意しているテスト用の入力ファイルが以下にあります。参考文献[1]の「Chapter 6 Running Gaussian98 Test Jobs」にこれに関する記述があります。

`/usr/local/g98/tests/com/testxxx.com` (xxx は 000 ~ 453)

また、参考文献[3]の例題、演習に使用している入力ファイルが以下のディレクトリーにあります。ファイル名は文献中の「インプットファイル表」を参照して下さい。

`/usr/local/g98/explore/quick` (クイックスタート)

`/usr/local/g98/explore/examples` (例題)

`/usr/local/g98/explore/exercise` (演習)

5. 実行時のトラブル

Gaussian98 使用時に起こりやすいトラブルと対処法の事例を以下に挙げます。ジョブの出力ファイル (Gaussian98 の出力、標準出力、標準エラー出力) のメッセージを確認して下さい。なお、問題解決には、入力ファイル中のルートセクションの記述を # (標準) や #T (簡略化した出力) ではなく #P (詳細の出力) としてみることも必要です。

A) `segmentation violation occurred.`

メモリー不足です。入力ファイルのメモリー制限値 `%Mem` を大きくします。

例 `%Mem=1024MB`

このときジョブのメモリー制限値 `##$-1M` は `%Mem` で指定した値より大きく設定 (`%Mem`+数百 MB 以上) しなければなりません。

B) `galloc: could not allocate memory.`

メモリー不足です。ジョブのメモリー制限値 `##$-1M` を大きくします。

例 `##$-1M 3GB`

また、`%Mem` の限界 (2GB 弱) 以上を指定したときもこのエラーとなります。

C) `Error termination in NtrErr: NtrErr called from FIOcnc.`

スクラッチファイルが大きすぎます。ディスクを使用しないアルゴリズムの変更等を検討して下さい。(3.スクラッチファイルについて参照)

例 `#P CIS/3-21G` を `#P CIS-direct/3-21G` に変更する

D) `Write error in NtrExt1: Disc quota exceeded`

`/home` にファイル使用量上限値以上のファイルを作成しようとしている場合にこのエラーとなります。不要なファイルを削除するか、`newuser` 手続きで上限値を増やします。

E) `Error termination via Lnk1e`

原因は特定できませんが、他にメッセージがないときはメモリー不足を疑います。計算が 2GB を超えてしまうとこのエラーとなります。スクラッチファイルが実行途中で削除されたときにもこのエラーとなります。