

量子化学計算における計算資源予測機能の設計と実装

西川 武志[†] 長嶋 雲兵[†] 関口 智嗣[†]

これまでにグリッド技術を用いた応用ソフトウェアの提供例として Quantum Chemistry Grid (QC Grid) を構築してきた。ポータルサービスの第1弾として量子化学計算プログラム Gaussian を Web インタフェースから利用者がアルゴリズムの負荷や計算資源について意識することなく迅速に結果を得られる QC Grid/Gaussian Portal の設計と構築を行ってきた。QC Grid/Gaussian Portal では依頼された計算ジョブの負荷に応じて適切な計算資源を選択するメタスケジューラを備えている。本論文ではメタスケジューラの計算資源予測機能がどのように計算ジョブの内容から計算資源を予測するか方式、その設計および量子化学計算プログラム Gaussian 98 に対する実装について報告する。計算時間、メモリやディスク要求量の推定には基底関数数を変数とした指数関数式を用い、計算時間の推定では実測結果から構築した予測式と理論式に基づいた予測式を組み合わせることが効果的であることを示した。同様にメモリやディスク要求量の指数関数式での推定に関しても実測予測式と理論予測式を組み合わせることで様々な分子に対して良好な結果を得た。また過剰なメモリ指定は計算時間が増大することを明らかにした。

Design and Implementation of Estimating Resource Requirements for Quantum Chemistry Calculation

TAKESHI NISHIKAWA,[†] UMPEI NAGASHIMA[†] and SATOSHI SEKIGUCHI[†]

We are developing Quantum Chemistry Grid (QC Grid) as a framework of Grid Application Service Provider. QC Grid/Gaussian Portal is the first implementation of the portal service. Users can get the computed results as fast as possible without knowing the specifications of each system environment and the computing loads of algorithms for quantum chemistry calculation. QC Grid/Gaussian Portal has a Meta-scheduler which chooses the most appropriate computing resources from contents of user requested jobs. In this paper, we report the method how to estimate resources by the function of predicting in the Meta-scheduler. And we describe design and implementation for the quantum chemistry calculation program, Gaussian 98. For the estimation of the calculation time, the amount of memory, and the amount of disk requirement, the equation of the exponential function with coefficients were defined based on the theory and the actual surveys. It was effective to combine the equation of estimation of the computing time based on the surveys or based on the theory. Our equations for the estimation of the amount of memory requirement and the amount of disk requirement got a good result for the calculation to the various molecules. It was shown the excessive memory allocation brought increase in the calculation time.

1. はじめに

これまでにグリッド技術を用いた応用ソフトウェアの提供例として Quantum Chemistry Grid (QC Grid) を構築してきた^{1),2)}。利用者が応用ソフトウェアに対して簡単にアクセスするためのポータルサービスとして量子化学計算プログラム Gaussian³⁾ を Web インタフェースから利用できる QC Grid/Gaussian Portal の設計と構築を行ってきた。QC Grid/Gaussian Por-

tal では、利用者がアルゴリズムに依存して変化する計算資源要求量について意識することなく、迅速に依頼された計算ジョブに応じて適切な計算資源を選択するメタスケジューラを備えている。QC Grid/Gaussian Portal を実際に運用し、新たな計算需要を掘り起こしている^{1),2)}。

量子化学計算プログラムが要求する計算機資源量は、分子を構成する原子数と基底関数系の組合せから定まる基底関数数や計算アルゴリズムに依存して際立って変化する。一方、近年のスーパーコンピュータセンタでは複数のアーキテクチャ(ベクトル計算機、大規模共有メモリを備えたスカラー計算機、PC クラスタ)等

[†] 産業技術総合研究所グリッド研究センター
Grid Technology Research Center, National Institute of
Advanced Industrial Science and Technology (AIST)

からなるため、様々なアーキテクチャから所望の計算に適切な計算資源を選択することは利用者にとって困難である。特に量子化学計算を材料設計や化学反応解析に役立てたい実験化学者等にとっては量子化学や計算機科学の専門的な知識に疎い場合が多く、計算機資源選定で困難に直面することが予想される。

QC Grid/Gaussian Portal の構成要素であるメタスケジューラは、計算依頼内容を解析し必要となるメモリやディスク容量、計算時間コスト等の推定を、量子化学のノウハウや計算機資源の性能情報等を蓄積したデータベースを参照して行う。その推定結果に応じた計算資源配分を行い、利用者が個々の計算実行環境の詳細を知らなくても計算資源を能率的に利用することが可能にする。

本論文では、メタスケジューラがジョブの要求する計算資源量 (CPU 時間, メモリ容量, ディスク容量) の予測を行うための設計と実装について報告する。

2. 量子化学計算手法と計算機使用量

QC Grid の主たる構成要素の量子化学計算プログラムは多体のシュレーディンガー方程式を非線形の一様問題に近似し自己無撞着場 (Self Consistent Field: SCF) 法 (別名 Hartree-Fock-Roothan 法, HF 法) により解くことを基本としている。この手法はおおまかに次のような手順を踏む。

- 初期値生成
- 一電子積分
- 二電子積分から Fock 行列生成
- Fock 行列対角化
- 諸性質の計算

SCF 法の演算時間の 95% から 99% を二電子積分から Fock 行列生成が占めることが分かっており、行列の次元数 (積分の基底関数数) の 4 乗に比例する計算量となっている^{4)~12)}。

HF 法では多電子のシュレーディンガー方程式を一電子問題に帰結したため電子相関の効果を十分に取扱いえず、電子相関が重要となる系では満足いく結果を与えない。そのため post-SCF 法と呼ばれる電子相関を何らかの形で取り込む様々な理論的方法が開発されている。代表的なものとして Moller-Plesset 摂動法 (MP n 法)^{3)~16)}, Coupled Cluster 法 (CC 法)¹⁷⁾, Quadratic Configuration Interaction 法 (QCI 法)¹⁸⁾ 等がある。post-SCF 法では演算時間を積分変換と配置生成のどちらかの時間が支配する。積分変換は基底関数数五乗以上に比例する計算量となっており、配置生成は励起される電子数と電子占有軌道と空軌道 (厳

表 1 量子化学計算手法と資源使用量

[文献 24) 第 6 章 p.124 「量子化学計算手法と資源使用量」より抜粋]

Table 1 Resource usage of quantum chemistry calculation.

方法	理論値			実際値	
	CPU	メモリ	ディスク	CPU	ディスク
HF					
慣用 SCF	N^4	N^2	N^4	$N^{3.5}$	$N^{3.5}$
Direct SCF	N^4	N^2	—	$N^{2.7}$	N^2
MP2					
慣用	N^5	N^2	N^4	N^5	N^4
Direct	N^5	N^3	—	N^5	N^2
SemiDirect	N^5	N^2	N^4	N^5	N^3
MP4,					
QCISD(T)	N^7	N^2	N^4	N^7	N^4

N: 基底関数数

密には仮想軌道) の組合せ, 基底関数数 N に対して $O(N!)$ の計算量となり, 基底関数数が増大するとともに急激に増大する。一例として文献 24) 第 6 章 p.125 に C_5H_{12} の計算例で同じ大きさの基底関数で HF 法と QCISD(T) 法では Cray T94 上の実行時間において最大 387 倍の開きがあることが紹介されている。同文献にある「量子化学計算手法と資源使用量」の表の抜粋を表 1 に示す。

表 1 で示される理論値も実際値もすべて文献 24) の初版が出版された当時, およそ 10 年前の計算機で扱える範囲での値であり, 理論値といっても当時の計算では無視できる次数の項を落としたものである。post-SCF 法では, そもそも電子配置が問題になるような計算は最初から無理だとして, 1 あるいは 2 電子励起に限定したため, 積分変換が支配的であるような計算量となっている。組合せ演算の影響は励起電子数に相当する N のべき乗次数の増加に収まっている。一方, 現在の計算機で計算可能な領域は, post-SCF 法で組合せ演算に比例する演算を無視できない系にまで達しており, 理論値もすでに修正が必要となっている。

さらに電子相関を電子密度の汎関数で表現することで基底関数数の三乗程度の計算資源コストで電子相関を取り扱えるとして, ここ 10 年急速に普及した密度汎関数法 (DFT 法) がある^{19)~23)}。実際には基底関数に対する計算量の依存性は, これまで紹介した HF 法, post-SCF 法, DFT 法のいずれでも基本となるアルゴリズムに対し様々な改良が加えられているため計算量の基底関数依存性の累乗数は低減されている^{4)~12)}。さらに Gaussian 03 では Fast Multipole Method (FMM) が導入され^{25)~30)} ある原子数以上では FMM が自動的に適用されクーロン相互作用の計算量を低減し, より大きな系での計算を可能にして

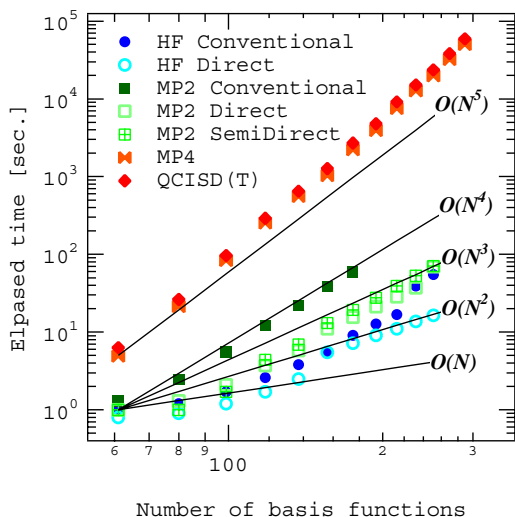


図 1 Pentium III 1.2 GHz CPU システム上 Gaussian 98 Rev. A9 での 1 点エネルギー計算に必要な経過時間と基底関数数との関係

Fig. 1 Size dependence of electronic structure methods.

いる。

図 1 に Pentium III 1.2 GHz 512 KB CPU キャッシュ 2way-SMP 主記憶 2 GB ディスク 36 GB 構成でシリアル実行で測定した種々の計算手法での 1 点エネルギー (SP) 計算に必要な経過時間と基底関数数との関係を示す。計算に必要な中間データをディスクに保存する慣用手法ではおおむね表 1 のとおりであるが、中間データをディスクに保存せずに必要に応じてそのつど再計算する Direct 手法では種々の計算性能向上の試み^{4)~12)} が功を奏し、基本アルゴリズムの理論値よりも計算時間の基底関数数依存性の次数が低減していることが分かる。特に基底関数数が 180 まででは表 1 の CPU 時間の理論値の N 累乗数より次数が低くなっている範囲が 512 KB CPU キャッシュ容量に対応している。実際の計算では SP 計算を基に、エネルギーの一次微分を利用し分子を構成する原子に働く力の計算 (Force 計算) とその応用である構造最適化 (Opt 計算)、二次微分を利用した振動数計算 (Freq 計算) や NMR 化学シフト (NMR 計算)、溶媒中での計算に必要な分子の体積を求める計算 (Vol 計算) 等がある。一次微分や二次微分を利用した計算では累乗数がそれぞれ微分の階数分だけ増加し、計算時間が増大する。SP 計算でも遷移金属を含む系等、それらを含まない系より収束が悪い系での SCF 計算の回数の予測は理論的には困難である。同様に構造最適化計算では最適化の回数も理論的には事前に予測し難い。しかしながら QC Grid では、1 つの計算資源のジョ

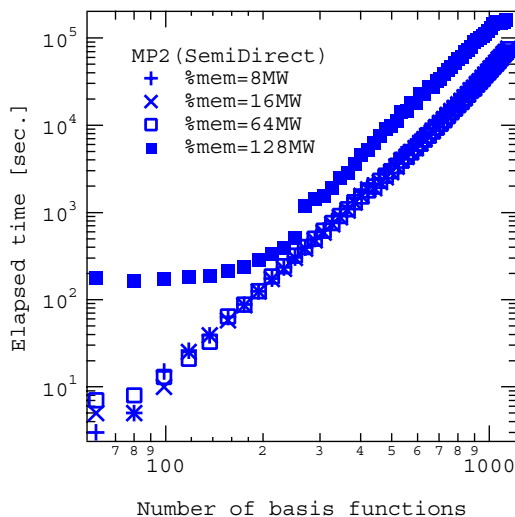


図 2 MP2 (SemiDirect) アルゴリズムにおけるメモリ指定量と経過時間と基底関数数との関係

Fig. 2 Memory allocation size dependence.

ブキューの時系列に多くのジョブを割り当てるより、グリッド上に存在する多種多様の膨大な計算資源の中から適切な計算資源を見出すことに主眼を置いている。それでもできる限りジョブ割当てを最適化したいという要望には、各サイトの運用ポリシーによって定められた SCF や最適化の繰返し数の上限の情報を利用すれば最長打ち切り実行時間の予測も可能である。したがって、ジョブのスケジューリングには十分な情報を得ることができる。

次に図 2 に MP2 SemiDirect 法では Gaussian の Link 0 コマンド (%mem=値) によるメモリ指定量を過剰に指定すると経過時間が増加してしまうことを示す。この原因の 1 つは、Gaussian は複数の実行モジュールからなるマルチオーバーレイ実行形式を採用していることにある。各オーバーレイ間での変数の受け渡しは RWF ファイルと呼ばれるスクラッチファイルやチェックポイントファイルを経由して行われ、そのメモリからディスクへの転送量はメモリ指定量に比例することにある。量子化学計算では、基底関数数の累乗でデータが増大するため、すべてをメモリ上に保持できないため、ディスク上にデータをいったん保存するアウトコアアルゴリズムが一般的である。ディスクのデータ転送性能はメモリ比べて圧倒的に劣ることも一般的である。データがすべてメモリ上に保持できれば高速になるはずであるという判断をする傾向が量子化学計算プログラムのある程度習熟した利用者にある。この傾向は最低限必要なメモリ容量よりある程度多くメモリを割り当てると計算時間が短縮するという事実

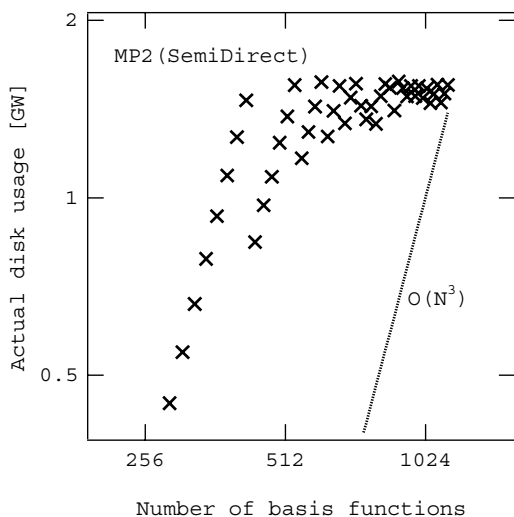


図 3 MP2 SemiDirect 法における標準設定でのディスク使用量の実測値

Fig. 3 Actual disk usage in MP2 by default.

によって補強されている．しかしながら図 2 に示すようにメモリを過剰に指定しても速くならないという事実があるが，一般に認識は低い．過剰にメモリを指定することで，グリッド上に多種多様な計算資源が存在しても大きなメモリを備えて利用できる資源は比較的少ないため，大規模メモリ資源にジョブが過剰集中する可能性がある．したがって，計算に最低限に必要なメモリ量の算出と適切なメモリ量の指定がジョブスケジューリング上では重要である．現在の実装では利用者に過剰なメモリ指定であることを示し変更することを促すか，強制的に運用側が指定したメモリ指定に置換することで解決を図っている．この際に示すメモリ量は最低限必要な量よりも多く与え，計算時間が短くなるよう配慮する．図 2 を例にとれば，128 MW を指定してきたような入力に対して 64 MW 以下の選択肢を提示する．

さらに実測値が重要であることを示す例として図 3 の MP2 Semi-Direct 法での SP 計算において系の大きさとディスク使用量との関係がある．表 1 に従えば MP2 Semi-Direct 法でのディスク使用量は $O(N^3)$ に比例する．実測してみるとディスク使用量が $O(N^3)$ に比例するものの系のサイズが大きくなり基底関数数が 422 を超えるところで最大使用量が 1.7 GW 未満に抑制されていることが分かる．Gaussian には MaxDisk オプションという限られたディスク容量でできる限り大きな系の計算が可能ないようにディスク容量に応じて計算量を分割する機能が存在するが，マニュアルの記述では Gaussian98 以降ではデフォルトでは制限がな

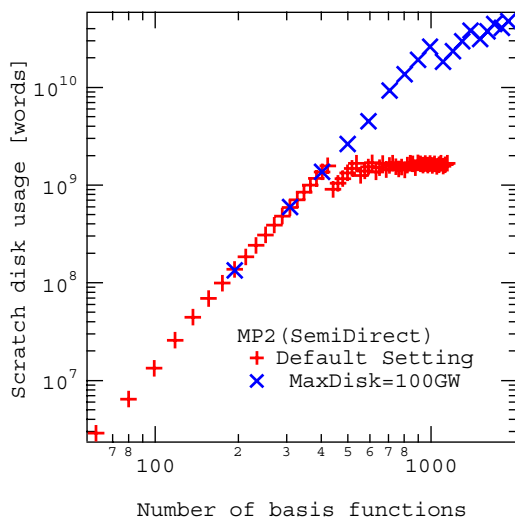


図 4 MP2 SemiDirect 法における MaxDisk=100 GW 設定でのディスク使用量の実測値と標準設定でのディスク使用量の比較

Fig. 4 Actual disk usage in MP2 by MaxDisk=100 GW.

いことになっている．ところが実測とマニュアルの記述が異なっている．なお MaxDisk オプションでディスク使用量の上限値を 100 GW と設定すると標準の設定ではディスク使用量が抑制された系のサイズでも図 4 に示すように 100 GW 近傍に達するまではディスク使用量が $O(N^3)$ に比例し増加している．すなわちデフォルトの設定では 2 GW のディスク容量が確保できる計算資源にジョブのスケジューリングが可能であるが，マニュアルの記述どおりにスケジューリングポリシーを設定するとより大きなディスクを備えた資源にしかスケジューリングできないことになる．

このように量子化学計算の基本的な理論に基づいた計算資源要求量と具体的な実アプリケーション，今回は Gaussian，が必要とする計算資源要求量との乖離は無視し難い．したがって，計算資源の利用効率（選択可能性）を上げるには，あるジョブが必要とする計算機資源使用量の見積りには，理論値だけでなく，実測値が重要である．

量子化学グリッド上の計算資源へのジョブスケジューリングにおいて，あるジョブが要求するメモリやディスクの資源要求量を満たす計算資源を選択し，さらに計算資源が提供する計算時間上限値とジョブの計算時間予測値を比較して最終的なジョブ投入先を決定する方針を採用した．特に SCF 収束の悪い系や構造最適化等，計算時間が予測困難な場合はプログラムの繰返し回数上限までの時間を算出し，その時間に適合する計算資源を選択する．

ありとあらゆる場合について資源要求量を実測することは限られた資源制約が存在するため不可能である。実測が困難な場合については理論に基づいた資源要求量予測を行い実測値に基づく予測と組み合わせた。以下で我々のアプローチの詳細について述べる。

3. 計算負荷予測機能の設計と実装

3.1 計算負荷予測機能の設計

Grid Application Service Provider フレームワークとして QC Grid を設計しておりポータルとして取り扱う量子化学計算プログラムは Gaussian にのみ限定しているわけではない。GAMESS³¹⁾, NWChem³²⁾ 等の量子化学計算プログラムにも対応可能なように設計している。したがって、量子化学計算の基本的な理論に基づいた資源予測式を採用するのは自然なことである。一方、前に述べたように個別アプリケーションに応じた資源予測ができないと有効な計算資源割当てが困難である。

したがって、QC Grid のメタスケジューラの計算負荷予測機能は一般的な量子化学アルゴリズムに応じた部分と個別のアプリケーションの実測値に応じた部分からなる。中核となる予測資源量 v の予測式は基底関数数 N の関数として、計算時間 (t : time), メモリあるいはディスク容量 (a : amount), すべてに対して

$$v(N) = d + CN^\alpha \quad (1)$$

という式で表すことで、単純ながら様々なアルゴリズムに対応可能である。実際に用いた予測資源量 v の予測式は、一般的な量子化学アルゴリズムの理論に応じた予測式 (T : Theoretical)

$$v_T = d_T + C_T N^{\alpha_T} \quad (2)$$

を基本として係数を用意し、個別のアプリケーションでの実測に応じた予測式 (S : survey)

$$v_S = d_S + C_S N^{\alpha_S} \quad (3)$$

で求めた係数を、系の大きさ、理論的導出の困難さ、実測の有無等に応じて、 v_T あるいは v_S の各要素を組み合わせて構築した。

計算時間に対する係数の詳細は次式のとおりでである。

$$t_{est} = d + C_s C_m C_j N^{m+j} \quad (4)$$

t_{est} : 予測時間

d : 遅延時間

C_s : システム性能係数

C_m : 手法種別係数

C_j : ジョブ種別係数

N : 基底関数数

m : 手法種別累乗数

$+j$: ジョブ種別累乗増分

メモリ容量およびディスク容量に対する係数の詳細は次式のとおりでである。

$$a_{est} = d + C_s C_m C_j N^{m+j} \quad (5)$$

a_{est} : 予測容量

d : 最小限必要容量

C_s : システム依存データ単位

C_m : 手法種別係数

C_j : ジョブ種別係数

N : 基底関数数

m : 手法種別累乗数

$+j$: ジョブ種別累乗増分

3.2 実装例

3.1 節で述べた、それぞれの変数について現在の実装では次のように定めている。遅延時間は現実のネットワークのトラフィックの変化やプログラムの起動時間やバッチキューイングシステムの応答時間等のシステムのオーバーヘッドに対応する時間の実測値から定めている。産業技術総合研究所先端情報計算センターで運用中の実装例では、十分に余裕をとり $d = 10$ 秒としている。システム性能係数としては量子化学計算アプリケーション (今回は Gaussian) での特定のベンチマーク問題 (付属の test397) に対する実測値の性能比を採用し、実測値がないシステムに関してはそのシステムの SPEC CPU ベンチマークスコアと実測値のあるシステムの SPEC CPU ベンチマークスコアの比から定めている。手法種別係数 C_m , 手法種別累乗数 m は SP 計算 (ジョブ種別係数 $C_j = 1$ とする) の実測値を基準として式 (4) にあてはめて最小自乗法により定めている。

ジョブ種別係数およびジョブ種別累乗増分は前述のとおりの SP 計算の係数を基準として、Force 計算, Freq 計算, NMR 計算等の目的とする諸性質の計算との実測から最小自乗法により求めている。

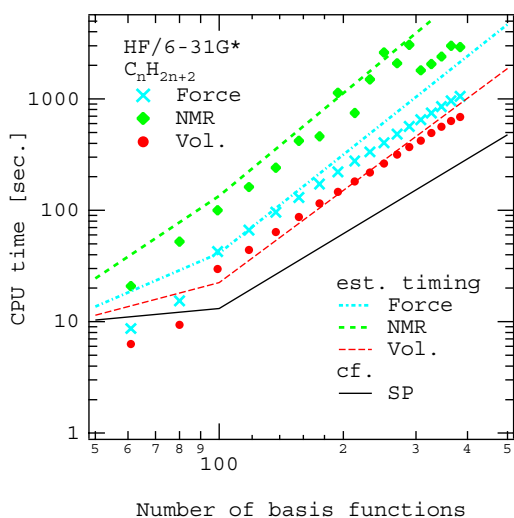
メモリとディスクの最小限必要容量は、システム依存のデータ単位や Gaussian のデフォルト設定値から定めている。システム依存データ単位はアウトコアでディスク上に記録する倍精度実数をアクセスするための整数サイズ等から定まる。たとえば 64 bit アドレッシングシステムでは整数サイズが 32 bit アドレッシングシステムの倍の値となっている。基底関数数に依存して変化するメモリやディスクの必要容量は少しでも不足すると計算できないため、原則として表 1 等の理論値を採用し、MaxDisk の隠れた制限値等は実測を反映させ切り替えている。

産総研で運用中の QC Grid/Gaussian Portal の計

表 2 Pentium III 1.2GHz システム上での Gaussian 98 の実測による予測時間の手法種別累乗数 m と累乗係数 C_m

Table 2 Power coefficients of the timing prediction for Gaussian 98 based on executed results on Pentium III 1.2GHz system.

方法	m 理論	m :実測	C_m :実測
HF	4	3.11	1.88e-6
MP2	5	3.33	2.11e-6
MP4	7	6.57	3.74e-11
CCSD	6	6.00	1.60e-10
QCISD	6	5.93	1.78e-10
QCISD(T)	7	6.55	4.90e-11
DFT	3	1.62	1.49e-2



方法	C_j	累乗増分 $+j$
Vol.: 排除体積計算	4	0
Force: エネルギー勾配計算	10	0
NMR: NMR 化学シフト	25	0.1
SP: エネルギー 1 点計算	1	0

図 5 Pentium III 1.2 GHz CPU システム上 Gaussian 98 Rev. A9 での HF/6-31G*モデルにおけるジョブ種別ごとの経過時間の実測値 (プロット) および予測式のジョブ種別係数 C_j とジョブ種別累乗増分 $+j$

Fig. 5 Estimating and actual CPU times by job types.

算資源を構成する Pentium III 1.2 GHz システム上での Gaussian 98 Rev. A9 を用いて直鎖アルカン C_nH_{2n+2} , ($n = 3, \dots, 15$) の 6-31G*基底関数系での種々の方法における SP 計算時間の実測 (図 1) に基づき最小自乗法により決定した予測時間式の累乗係数 m , 手法別係数 C_m の一例を表 2 に示す. また図 5 に実測値を基に最小自乗法により決定したジョブ種別係数 C_j とジョブ種別累乗増分の係数 $+j$ の一例を示す. 同様の手続きで決定したディスク使用量予測式の係数の一例を表 3 に示す.

図 6 に直鎖アルカン C_nH_{2n+2} , ($n = 3, \dots, 15$)

表 3 ディスク使用量係数

Table 3 Disk usage coefficients for Gaussian 98.

方法	d	m	C_m
HF Direct SCF			
Single Point	40 MB	2	4.7e-4
Volume	40 MB	2	4.7e-4
Force	40 MB	2	4.9e-4
NMR	40 MB	2	5.1e-4
SCF=QC	40 MB	2	1.4e-4
MP2 SemiDirect	40 MB	3	2.0e-4
MP4	60 MB	4	3.5e-6
CCSD	60 MB	4	7.8e-6
QCISD(T)	60 MB	4	7.2e-6
DFT	40 MB	2	4.5e-4

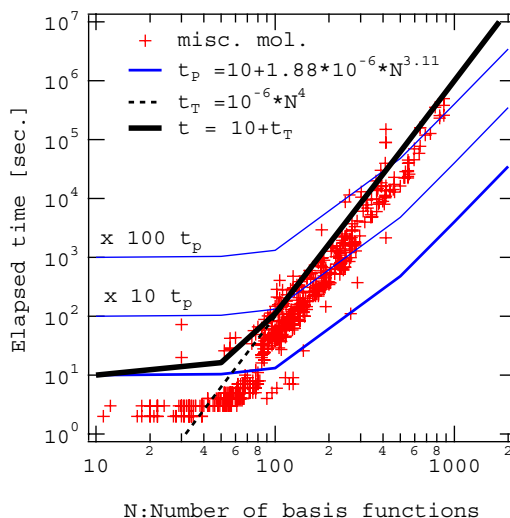


図 6 HF 法における経過時間予測式と、様々な分子の様々な基底関数系での実測値との比較

Fig. 6 Comparison of estimating and actual elapsed time in HF method.

での実測に基づく HF 法エネルギー 1 点計算の経過時間予測式 (太実線) と様々な分子 (benzen, C_{60} fullerene, H_2SO , H_2SO_2 , OMS, OTS, acetylene, adamantane, anthracene, azulene, benzene, bicyclo[2,2,1]heptane, bicyclo[2,2,2]octane, biphenyl, butane, cis-decalin, coronene, cyclobutane, cycloheptane, cyclohexane-boat, cyclohexane-chair, cyclooctane, cyclopentadiene, cyclopentene, cyclopropane, fluorene, formaldehyde, formamide, formic acid, furan, hydrogen cyanide, naphthalene, phenanthrene, pyrene, pyridazine, pyridine, pyrimidine, pyrrole, tetrahydrofuran, thiophene, trans-decalin) に対する HF 法での代表的な基底関数系 (STO-3G, 6-31G*, 3-21G, 6-31G**, 6-31+G**, cc-pVDZ, cc-pVTZ, 6-311G*, 6-31G, 6-311+G

(3df,2p), CEP-31G, CEP-31G*, CEP-121G, CEP-121G*, LANL2DZ, SDD, aug-cc-pVDZ, aug-cc-pVTZ) を用いた実測値と, 図 6 の実測に基づき累乗数 m や手法係数手法別係数 C_m を求めた予測式 (t_P : 太実線), その 10 倍 ($\times 10 t_P$), 100 倍の曲線 ($\times 100 t_P$), 理論上の累乗数 (4 乗) に基づき, 手法別係数 C_m を最小自乗法により求めた経過時間予測式 (t_T : 太破線), 前者に実測に基づいた遅延時間 10 秒を加えたもの ($t_T + 10$: 極太線) との比較結果を示す. 今回の直鎖アルカン C_nH_{2n+2} , ($n = 3, \dots, 15$) での 6-31G* 基底関数系に基づく予測式の算出では基底関数数が 61 から 289 までの範囲での実測を基にした.

まず, t_T を検討すると, 基底関数数が 80 以上の領域にわたって良好な予測となっている. 実測値に安全係数を乗じた遅延時間を加えた $t_T + 10$ では基底関数数 80 以下での t_T の過小評価の難点を克服している. 一方, t_P を検討するとそのままでは, 様々な分子と様々な基底関数系での計算では基底関数数が 11 から 1080 までの範囲に及ぶため, 外挿となる領域では計算時間が最大で約 1000 倍もの過小評価となってしまった. 過小評価回避法として実測に基づく予測式に安全係数を考慮して一定係数倍にするという解決策が考えられる. t_P を 10 倍および 100 倍した式と様々な分子での計算結果を比較すると系のサイズが小さな領域では無駄が多いことは明白である. ただし計算資源の選択という観点で計算資源 (計算時間) の予測という観点では, サイトの運用ポリシーに応じて, 理論に重きを置くか, 実測に重きを置くかは QC Grid の管理者が選択すればよい.

様々な分子を用いた計算と直鎖アルカンでの計算では, 分子の形状, 構成元素の種類によって SCF 計算の収束状況が異なるという問題もあるが, Gaussian 自体の特性, 基底関数が Pople 系 (STO-3G, 6-31G*, ..., etc.) とその他の系では同じ基底関数数でも前者に比べて後者の計算性能が低いと, 基底関数数が 61 から 289 までの範囲でも最大で約 200 倍の過小評価が生じている. また, 計算時間の予測は非線型方程式を繰返し法によって解くことになるので収束がどうなるのかは事前に予測が困難であり, 計算時間の高精度な見積りも難しいということが改めて確認された結果となった.

しかしながら SCF 収束回数や構造最適化の回数は計算機アーキテクチャが変わっても大幅に変化することはなく, 類似の分子では同じような収束傾向を示すため, 実測結果を将来のジョブの計算時間の予測に利用する QC Grid にとって深刻ではない. また 1 つの

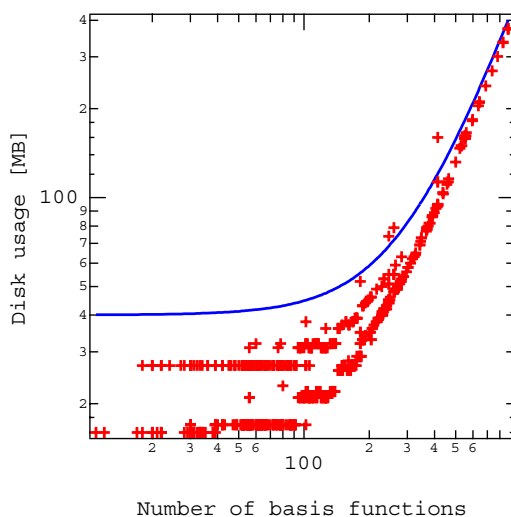


図 7 HF 法における, 実測に基づく予測ディスク容量と, 様々な分子での様々な基底関数系での実測値との比較
Fig. 7 Estimating and actual disk usages in HF method.

計算資源の時系列 (ジョブキュー) の中により多くのジョブを詰め込むという場合には高精度の計算時間の見積りは重要であるかもしれないが, 多数の様々な計算資源の存在が前提となるグリッド環境では高精度の計算時間の見積りは重要ではない. さらに QC Grid はこうした様々な分子での計算結果を反映して実測による計算予測式を再構築する概念を持つため, 基底関数の Pople 系とその他の系の採用頻度等は経験の蓄積とともに予測時間の誤差は遞減可能である. 今後の課題としてはこの仕組みを自動化する手法を検討することが存在する.

図 7 に HF 法における, 実測に基づく予測ディスク容量と, 様々な分子での様々な基底関数系での実測値との比較を示すが, こちらは計算時間とは状況が異なり, 非常に良好な予測となっている. メモリ容量についても同様の予測式を用いており, メモリが足りなくて計算が中断されたということは生じなかった. ディスク容量とメモリ容量はアルゴリズムと系の大きさが決まれば最低限必要な容量が定まるため, 定数で与えられる式 (5) の d の値を十分にとれば過小評価は回避できるのは当然といえば当然である.

3.3 Gaussian 03 での問題

前節までは Gaussian 98 に対する実装とその適用例について述べたが, 2003 年に新バージョンである Gaussian 03 が発表された. Gaussian 03 では周期境界条件への対応, ダイナミクス計算への対応, DFT 法における密度近似等の新機能の追加, 新たな基底関数系の追加等がなされ, 現状ではこれらの新機能には

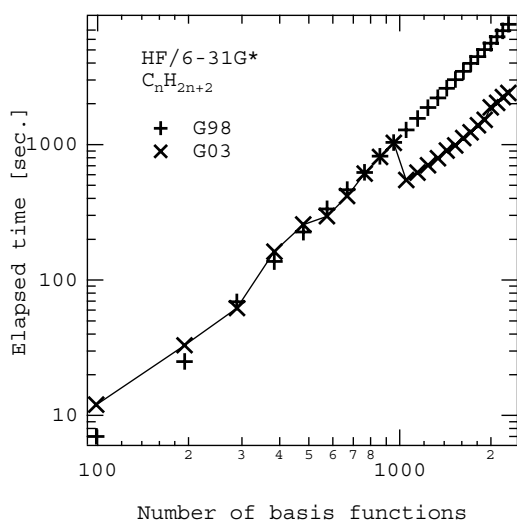


図 8 G98 と G03 の差異: HF/6-31G* 一点エネルギー計算
Fig. 8 Difference between G98 and G03 by HF/6-31G* SP.

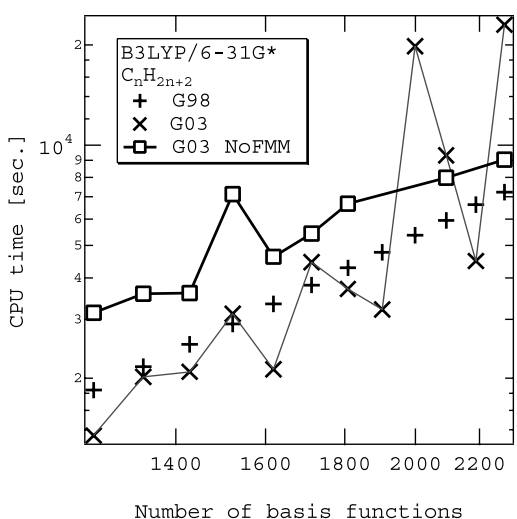


図 9 G98 と G03 の差異: B3LYP/6-31G* 一点エネルギー計算
Fig. 9 Difference between G98 and G03 by B3LYP/6-31G* SP.

対応できていない。既存の手法であっても根源的なアルゴリズム、初期値生成手法や SCF の基本アルゴリズムに大幅な変更が加えられ、Gaussian 98 までと計算の前提条件が変わってしまうという重大な問題が生じた。

たとえば図 8 に示すように Gaussian 03 で大きな分子系に対して自動的に導入される Fast Multipole Method^{25)~30)} によって HF 法では 900 基底関数以上の大きさでは Gaussian 98 に比べて計算時間の低減が著しい。一方、図 9 に示すように DFT 法 (B3LYP

法)ではいくつかの系においてデフォルトの計算条件設定では著しい計算時間の増大を招いている。FMM を適用しないオプション NoFMM を指定するとこの現象は抑制されるが、1524 基底関数の系では NoFMM オプションの副作用なのか、計算時間が極端に増大するという現象が見られている。今回示した現象は極端な例だが、実測値を基にし、今回のような特異値にどのように対応するかは今後の検討課題の 1 つである。

4. まとめ

計算資源の予測手法として本論文で提唱した理論値を実測値に基づいて修飾した予測式を利用する方法は妥当であることが分かった。ジョブをスケジューリングするための計算資源の予測という観点からは計算資源を過剰に見積もることは予定した資源より要求量が少なく済むので資源に遊びができる可能性があるものの致命的な問題ではない。この資源要求量の精度をどこまで求めるかは、運用ポリシーの問題であるため明確な結論は出ない。しかしながらグリッド上の様々な計算機資源の中から量子化学計算ジョブに適切な資源を選択するには本論文で述べた仕組みで十分対応可能である。

実測からの予測式の構築の際に用いるベンチマーク計算のための分子の種類、基底関数種別等は十分な分布を持つことが重要である。QC Grid/Gaussian Portal を実際に運用して蓄積された結果データベースから性能情報を抽出すれば、直鎖アルカン C_nH_{2n+2} , ($n = 3, \dots, 15$) の 6-31G* 基底関数系での実測に基づく、いわば貧弱な実測値から予測式を作成しても、システムを運用していくことで予測精度が向上していくことが期待できる。この際にどのようなタイミングで蓄積結果データベースから情報を抽出し、メタスケジューラの資源予測機能に反映させるか、その手法等は今後の研究課題である。

また Gaussian 98 から Gaussian 03 への例のように、バージョンアップにともなう資源要求量の変化等への対応は Gaussian に限らず他の量子化学計算プログラムでも必要となる可能性がある。その対応を個別にしていたのでは時間や費用がいくらあっても足りないということになる。この時間的経済的制約への対応策として我々が開発し運用している QC Grid/Benchmarking Portal システム³³⁾ によって量子化学計算プログラムベンチマーク問題の配布・結果の収集を行い、様々な計算環境でのベンチマーク性能情報への対応を図ることが有効であると考えられる。

Gaussian 03 は量子化学の様々な成果を取り込んだ

巨大なプログラム(Fortran ソースコードで約 120 MB 弱)となっており, 1 人の科学者がすべての機能を把握することが困難なアプリケーションとなっている. 今回報告した MP2 でのメモリ指定量と性能の関係や計算規模とスラッチディスク使用量の関係の知見のように広く知られていない性能情報を上手に共有する手段が必要である. ある計算ジョブが失敗した原因について量子化学の専門家や先んじて当該ジョブに遭遇した利用者等からの意見を収集して共有する仕組みとして QC Grid/Benchmarking Portal システム³³⁾ を拡張することも今後の重要課題である.

謝辞 メタスケジューラ的设计についてご議論いただいたグリッド研究センター伊藤智グリッド応用チーム長に感謝する. ベンチマークのための計算機資源を提供していただいた産総研先端情報計算センター, 日本ヒューレット・パッカード株式会社に感謝の意を表する.

参 考 文 献

- 西川武志, 長嶋雲兵, 関口智嗣: Quantum Chemistry GRID/Gaussian Portal の構築, 情報処理学会研究報告, 2002-HPC-90-8, pp.43-48 (2002).
- 西川武志, 長嶋雲兵, 関口智嗣: Quantum Chemistry GRID/Gaussian Portal Phase 2, 情報処理学会研究報告, 2002-HPC-92-8, pp.43-48 (2002).
- Frisch, M.J. et.al: Gaussian 03, (Revision B.3), Gaussian, Inc., Pittsburgh PA (2003).
- Frisch, M.J., Head-Gordon, M. and Pople, J.A.: *J. Chem. Phys.*, Vol.141, No.189 (1990).
- Head-Gordon, M., Pople, J.A. and Frisch, M.J.: *Chem. Phys. Lett.*, Vol.153, No.503 (1988).
- Frisch, M.J., Head-Gordon, M. and Pople, J.A.: *Chem. Phys. Lett.*, Vol.166, No.275 (1990).
- Frisch, M.J., Head-Gordon, M. and Pople, J.A.: *Chem. Phys. Lett.*, Vol.166, No.281 (1990).
- Almloff, J., Korsell, K. and Faegri Jr., K.: *J. Comp. Chem.*, Vol.3, No.385 (1982).
- Head-Gordon, M. and Head-Gordon, T.: *Chem. Phys. Lett.*, Vol.220, No.122 (1994).
- Head-Gordon, M. and Pople, J.A.: *J. Chem. Phys.*, Vol.89, No.5777 (1988).
- Gill, P.M.W.: *Adv. Quant. Chem.*, Vol.25, No.143 (1994).
- Taylor, P.R.: *Int. J. Quant. Chem.*, Vol.31, No.521, (1987).
- Moller, C. and Plesset, M.S.: *Phys. Rev.*, Vol.46, No.618 (1934).
- Pople, J.A., Seeger, R. and Krishnan, R.: *Int. J. Quant. Chem. Symp.*, Vol.11, No.149 (1977).
- Krishnan, R. and Pople, J.A.: *Int. J. Quant. Chem.*, Vol.14, No.91 (1978).
- Krishnan, R., Frisch, M.J. and Pople, J.A.: *J. Chem. Phys.*, Vol.72, No.4244 (1980).
- Pople, J.A., Krishnan, R., Schlegel, H.B. and Binkley, J.S.: *Int. J. Quant. Chem.*, Vol.XIV, No.545 (1978).
- Pople, J.A., Head-Gordon, M. and Raghavachari, K.: *J. Chem. Phys.*, Vol.87, No.5968 (1987).
- Hohenberg, P. and Kohn, W.: *Phys. Rev.*, Vol.136, No.B864 (1964).
- Kohn, W. and Sham, L.J.: *Phys. Rev.*, Vol.140, No.A1133 (1965).
- Slater, J.C.: *Quantum Theory of Molecular and Solids. Vol.4: The Self-Consistent Field for Molecular and Solids*, McGraw-Hill, New York (1974).
- Pople, J.A., Gill, P.M.W. and Johnson, B.G.: *Chem. Phys. Lett.*, Vol.199, No.557 (1992).
- Becke, A.D.: *J. Chem. Phys.*, Vol.98, No.5648 (1993).
- Foresman, J.B. and Frish, A.E. (著), 田崎健三 (訳): 電子構造論による化学の探求, 第 2 版, ガウシアン社 (1998).
- Millam, J.M. and Scuseria, G.E.: *J. Chem. Phys.*, Vol.106, No.5569 (1997).
- Burant, J.C., Scuseria, G.E. and Frisch, M.J.: *J. Chem. Phys.*, Vol.195, No.8969 (1996).
- J.C. Burant, M.C. Strain, G.E. Scuseria, and M.J. Frisch, *Chem. Phys. Lett.*, Vol.258, No.45 (1996).
- Burant, J.C., Strain, M.C., Scuseria, G.E. and Frisch, M.J.: *Chem. Phys. Lett.*, Vol.248, No.43 (1996).
- Strain, M.C., Scuseria, G.E. and Frisch, M.J.: *Science*, Vol.271, No.51 (1996).
- Daniels, A.D., Millam, J.M. and Scuseria, G.E.: *J. Chem. Phys.*, Vol.107, No.425 (1997).
- Schmidt, M.W., Baldridge, K.K., Boatz, J.A., Elbert, S.T., Gordon, M.S., Jensen, J.H., Koseki, S., Matsunaga, N., Nguyen, K.A., Su, S., Windus, T.L., Dupuis, M., Montgomery, J.A.: *J. Comput. Chem.*, Vol.14, pp.1347-63 (1993).
<http://www.msg.ameslab.gov/GAMESS/>
- High Performance Computational Chemistry Group: NWChem, A Computational Chemistry Package for Parallel Computers, Version 4.1, Pacific Northwest National Laboratory, Richland, Washington 99352, USA (2002).
<http://www.emsl.pnl.gov/docs/nwchem/>
- 西川武志, 長嶋雲兵, 関口智嗣: Quantum Chemistry Grid/Benchmarking Portal の構築, ハイパフォーマンスコンピューティングと計算科

学シンポジウム, HPCS2003 (2003).

(平成 15 年 10 月 10 日受付)

(平成 16 年 1 月 24 日採録)



西川 武志 (正会員)

1969 年生. 1992 年慶應義塾大学
理工学部計測工学科卒業. 1998 年
慶應義塾大学大学院理工学研究科計
測工学専攻博士課程修了. 博士(工
学). 同年日本学術振興会未来開拓

学術研究推進事業リサーチ・アソシエイト(分子科学研究
所理論研究系). 2001 年 4 月より独立行政法人産業
技術総合研究所. 現在同所センターグリッド応用チ
ーム研究員. 化学物理, 計算機性能評価技術, グリッド
アプリケーションサービスの研究開発に従事. 日本物
理学会, 日本化学会, フラーレン・ナノチューブ研究
会各会員.



長嶋 雲兵 (正会員)

1955 年生. 1983 年北海道大学
大学院理学研究科博士後期課程化学第
二専攻修了. 理学博士. 同年岡崎国
立共同研究機構分子科学研究所電子
計算機センター助手. 1992 年お茶

の水女子大学理学部情報科学科助教授. 1996 年同教
授を経て, 1998 年通産省工業技術院物質工学工業技
術研究所基礎部理論化学研究室長. 1999 年同産業技
術融合領域研究所計算科学研究グループ長, 2001 年
4 月独立行政法人産業技術総合研究所に改組. 同所先
端情報計算センター情報基盤研究開発室長. 2002 年
より同所グリッド研究センター総括研究員. 筑波大学
連携大学院大学教授. 計算化学, 情報化学, 大規模数
値計算, 広域分散並列処理の研究開発に従事. 日本化
学会, IEEE, 応用数理学会, 計算工学会, 化学工学
会, 日本コンピュータ化学会各会員.



関口 智嗣 (正会員)

1959 年生. 1982 年東京大学理学
部情報科学科卒業. 1984 年筑波大学
大学院理工学研究科修了. 同年電子
技術総合研究所入所. 以来, データ駆
動型スーパーコンピュータ SIGMA-
1 の開発等の研究に従事. 2001 年独立行政法人産業技
術総合研究所に改組. 2002 年 1 月より同所グリッド
研究センターセンター長. 並列数値アルゴリズム, 計
算機性能評価技術, グリッドコンピューティングに興
味を持つ. 市村賞, 情報処理学会論文賞受賞. グリッド
協議会会長. 日本応用数理学会, ソフトウェア科学
会, SIAM, IEEE, つくばサイエンスアカデミー各
会員.