

スパコン上で C 言語を用いた複素指数関数計算について

平成 27 年 5 月 11 日

プログラム名：

プログラムの概要：

原子核の励起状態の計算

助言、提案の欲しい問題：

日立 SR16000 上で C を用いた計算において、複素計算特に `cexp` が異常に遅い

使用する計算機 (予定も含む)：

日立 SR1600@京大基研

用いる言語：

C,

質問など：

基研の SR16000 上の IBM 製 `xlC` で複素数の計算を行いたいと考えています。しかし同等の Fortran のコードと比較して、C のコードは 4 倍以上も遅くなるという問題が生じています。原因を調べた結果、特に標準ライブラリ `libm.a` 中の複素数 `cexp` が Fortran と比べて異常なぐらい遅い事がわかりました。おそらく `cexp` を自分で置き換えたりするのが一般的な方法でしょうが、その場合は複素数 `cexp` が提供されているその他のライブラリをご存知ないでしょうか？またこのような場合、他に何か解決策があるのでしょうか？同じく C をお使いの方が、このような問題が生じた時に、一般的にどのように対処されているのか、お知恵を拝借させてください。

回答

他社製のスパコンも含めて調査しましたが、残念ながら現時点では一般的に、C 言語での複素計算を含んだコードの最適化は Fortran と比較してかなり遅れている様です。特に C09 規格を満たす C 言語での `CEXP` の実装(`libm.a`)は、大部分のメーカーのスパコンでもかなり遅くなっています (Fortran 比数倍～十倍程度)。

これを解消する一つの手段は、`CEXP` を自分で用意して置き換える方法があります。GNU

Scientific Library (GSL)には、優れた複素関数のコレクションがあり、ソースコードの参照が可能です。GSLでは CEXP の実装にオイラー公式が使われており、 $e^{x+iy} = e^x(\cos y + i \sin y)$ と展開する事により、実数の早い指数関数、sincos 関数を利用する事ができます。この方法により、幾分パフォーマンスは改善されます。(なお SR16000 上において、この方法は Fortran に対しても有効でした。)

また、昔から良く行われている、自分自身で複素数を構造体や配列などを用いて定義する方法もあります。C 言語に対しては今もこの方法はかなり有効である事を確認しました。この方法とインライン展開を用いる事で、Fortran とほぼ同等の演算速度である事を SR16000 上で確認しました。おそらく Fortran の最適化レベルが高いコンパイラは、この方法と同等のバイナリを生成していると考えられます。

結論として、一般的に C 言語での複素計算を用いたコードへの最適化は Fortran と比較して各社まだかなり遅れている様です。C 言語のソースを注意深く書き直す事で、Fortran と同等のパフォーマンスを得る事ができますが、かなりの労力が必要です。初心者はこの辺りの事情に注意して言語を選択してください。