週報

平岡立成

1　今週やったこと

- 『神経回路シミュレーション』山崎匡 読み進め

2　『神経回路シミュレーション』要約

第 12 章　高性能神経計算入門

ここまで学習してきた神経回路シミュレーションは，高々数千個のニューロン，数十万個のシナプスといった規模だが，実際の脳は約 860 億個のニューロン，約 100 兆個のシナプスからなり，通常のコンピュータでは，膨大な計算時間，格納できるメモリ量といった問題からシミュレーションができない．そこでスーパーコンピュータ（スパコン）を用いる．現在のスパコンは，膨大な数の計算機をネットワークで接続することで実現する．例えば理化学研究所計算科学研究センターのスパコン「富岳」は約 16 万台の計算機により構成される．また，GPU という並列計算のためのハードウェアを搭載したスパコンもある．このようなスパコンを活用することを考える研究が，高性能計算（High Performance Computing, HPC）である．

神経回路シミュレーションにおいても，このような並列計算による高速化を目指す「高性能神経計算」について，第 12 章では説明していく．

12.1 スパコンの性能指標

スパコンの性能指標として次のようなものがある．

FLOPS... 浮動小数点の四則演算可能回数 [回/s]
B/F... メモリバンド幅 [Bytes/s][*1]／ FLOPS[回/s]

FLOPS はムーアの法則[*2]に伴い指数関数的に増加する．「富岳」においては 442PetaFLOPS[*3]，つまり 1s 間に約 44 京回の基本演算が可能である．次に，B/F の説明のためにコードの一例を示す．

```
double a[N], b[N], c[N];
...
for ( int32_t i = 0; i < N; i++ ) {
    a[i] = b[i] + c[i];
}
```

Listing 1　double 型変数の 1 回の四則演算

このコードにおける 1 回の for ループについて，演算回数は右辺における加算の 1 回であるが，その間，double の変数を 2 個読み，1 個分の書き込みを行うという，計 24Bytes の読み書きを要する．この場合，このコードを最高速度で実行するには B/F=24 であればよい．しかし,「富岳」のマイクロプロセッサである Fujitsu A64FX であってもわずか B/F=0.37[*4]程度である．一般的な CPU においても B/F=0.5 程度であり，メモリバンド幅の低さが性能を理論値まで上げる上での障壁となっていることがわかる．これを改善するためにキャッシュメモリがあり，A64FX の場合，L1 キャッシュに格納した場合，B/F は 4 程度まで増える．このために，キャッシュの利用効率改善を目的とした問題の分割や計算順序の入れ替えといった検討が大切になる．

12.2 高性能計算（HPC）とは何か？

プログラム中には並列化可能な部分と不可能な部分がある．

p 割の部分が並列化可能で，その性能向上率を s とすると，プログラム全体の性能向上率は

$$\frac{1}{(1-p)+\left(\frac{p}{s}\right)}^{*5} \tag{1}$$

となる

並列にはモデル並列とデータ並列の 2 つがある．モデル並列は，ニューラルネットワークならば，巨大なニューラルネットをいくつかに分割し解くような，1 つの問題（=モデル）を分割し，並列に解く方法であるが，これにはモデル間の通信を要し，通信待ち等のボトルネックの要因が多い．一方，データ並列は同一のニューラルネットに異なるデータセットを与えトレーニングするような並列処理を指し，それぞれは独立した処理なので，モデル間の通信を要さない．

12.2.2 並列計算機の種類

並列計算をする計算機は，共有メモリ型と分散メモリ型に大別される．共有メモリ型は，一般的な CPU のような，複数の演算器（=マルチコア）で同一のメモリを参照する「共有メモリ型」と，ネットワークで接続された複数のコンピュータのような，それぞれ異なるメモリ空間を使用する「分散メモリ型」に大別できる．前者は演算器間でメモリのコピー等による通信を要さないこと，後者は計算機の台数に制約が無いことが利点であり，現在のスパコンはこのハイブリットとなっている．共有メモ

[*1] 単位時間あたりに伝送可能なデータ量
[*2] 集積回路上のトランジスタ数は 2 年で 2 倍となるという定説．
[*3] Peta=10^15
[*4] メモリバンド幅が 1.0TBytes/s，演算性能が 2.7TFLOPS
[*5] これをアムダールの法則という．

リ型なら OpenMP，分散メモリ型なら MPI（Message Passing Interface）と，それぞれ並列化をサポートするライブラリが存在する．

12.2.3 **並列計算の効率**

12.3 **神経回路シミュレーションの並列化**