ニューラルネットワークアクセラレータにおける コア間通信削減のためのタスク配置の検討

進藤 智司*，津邑 公暁（名古屋工業大学）

Task Mapping for Reducing Communication Cost on Neural Network Accelerators

Satoshi Shindo, Tomoaki Tsumura (Nagoya Institute of Technology)

1. はじめに

画像認識や音声認識等において，ニューラルネットワークを用いた機械学習が注目されているが，認識率を向上させるために，ネットワーク規模を拡大させることが主流であり，シミュレーションに必要な計算時間が増大している．そこで，計算時間の短縮と消費電力の削減を目指し，**ニューラルネットワークアクセラレータ（NNA）**[1]が盛んに研究されている．我々は高性能な NNA の開発を目指しているが，ソフトウェア実行の際には考慮する必要のなかった点が性能に影響を及ぼす可能性がある．そこで本稿では，複数コアへのタスク配置に着目し，コア間通信量最小化のための適切な配置について検討する．

2. ニューラルネットワークアクセラレータ

我々が開発中の NNA は Shared Memory と複数のコアから構成され，各コアが Shared Memory に対してデータを読み書きすることで，全てのコアでデータを共有することができる．各コアは，前層から与えられる入力を格納する Input Buffer，シナプス重みを格納する Weight Buffer，これらのバッファに接続した，ニューロンの出力計算用の演算ユニットから構成される．このアクセラレータでは，コアが Shared Memory からデータを読み出す際，一旦 Input Buffer に格納した後，演算ユニットに読み出される．このとき，Input Buffer 中に読み出したいデータがあれば，そのデータを再利用することで Shared Memory からの読み出しを省略することができる．

なお，ニューラルネットワークは層間にデータ依存があるが，層内にはないため，同層に含まれるニューロンは複数コアで並列に出力計算することが可能である．ニューロンの出力計算には前層のニューロンの出力が入力として必要となるため，全てのコアで一層ずつ順にニューロンの出力を計算していくことになるが，この際，ニューロンの出力を全コアで共有する必要がある．

3. 複数コアへのタスク配置

ニューロンの出力を全コアで共有する際に，Shared Memory を介したコア間通信が発生する．そこで本稿ではコア間通信量を最小化することを目指し，各コアの入力の受け取り方に着目して，2 つのタスク配置の方針を考える．1 つ目はニューロンをグループ化し，各グループに属するニューロンの出力計算を個別のコアに割り当てる方針である．この方針では，前層から与えられる全ての入力を受け取るための通信が必要となるが，各ニューロンの出力計算をそれぞれのコアで完結して行うことができる．2 つ目は前層から与えられる入力をグループ化し，それぞれのグループに属する入力に対応した計算を個別のコアに割り当てる方針である．この方針では，各コアの計算結果を統合するための通信および処理が別途必要となるが，Input Buffer 中のデータの再利用回数を増やすことができる．

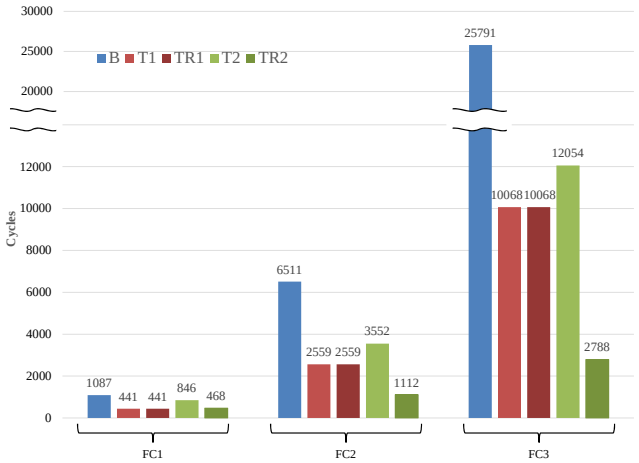

Fig.1　Evaluation Results

4. 評価

前章で述べた 2 つのタスク配置方法を，現在我々が開発中の NNA 上に実装しシミュレーションにより評価した．評価では，入出力数がそれぞれ 512（FC1），1280（FC2），2560（FC3）である全結合層に含まれるニューロンの出力を計算する際の実行サイクル数を測定した．評価結果を Fig.1 に示す．結果は，1 コアを使用するモデル (B)，16 コアを使用し 1 つ目のタスク配置を適用するモデル (T1)，2 つ目のタスク配置を適用するモデル (T2)，それぞれのタスク配置を適用し Input Buffer を再利用するモデル (TR1)(TR2) を示している．評価の結果，FC1 では (T1) の実行サイクル数が (T2) および (TR2) と比べて少なくなっている一方，FC2 および FC3 では (TR2) が最も少ない実行サイクル数となっている．これにより，タスク配置が実行サイクル数に影響を与えることと，ネットワークの規模によって有効なタスク配置が異なることを確認した．

文　献

(1) Chen, et.al.: Dadiannao: A machine-learning supercomputer, *Proc. 47th Annual Int'l Symp. on Microarchitecture*, pp.609–622 (2014)