A Communication Lower Bound for Convolutional Neural Nets

Yuhui Wang, Grace Dinh

November 2019

1 Introduction

As deep learning becomes more and more popular, convolution neural nets (CNNs) are widely used in many tasks. Thus, it is important to implement CNNs in a effective way. There are two costs to consider: arithmetic and communication. The cost of moving data, either between different levels of memory or between different processors, is much higher than arithmetic options, which means a communication optimal strategy is critical to performance [1].

In this paper, we will analyze the theoretical communication lower bound for CNNs. We will firstly introducing a simplified CNN model and finding a communication lower bound in Section 2, then extend it to a general CNN model in Section 3. In section 4, we derive a sequential I/O lower bound for a single processor system. The improvement over prior lower bounds comes mainly from constant factors, and major components of this paper are Lemma 2.1, Theorem 3.1 and Corollary 4.1.

2 A Simplified CNN Model

We are going to define and analyze a simplified CNN computation , then extend it to a general model later in section 3.

Definition 2.1. A simplified CNN Model is a convolutional operation with one square image, one square filter, one input channel and stride sizes set to one. We can write down the simplified CNN computation as follows:

for $w, h, r, s = 0 : \{N, N, L, L\}$ - 1
    $Out(w, h) += Img(r + w, s + h) \times Filter(r, s)$

where Img and Out are $N \times N$ arrays and $Filter$ is a $L \times L$ array. Boundary conditions are ignored.

Definition 2.2. A useful multiplication for a CNN model is a scalar multiplication $Img(r + w, s + h) \times Filter(r, s)$that eventually contributes to the sum that forms $Out(w, h)$.

For the simplified model, it is easier to find a tight communication lower bound. The proof is inspired by [2], where a matrix multiplication communication lower bound is provided. Now we are ready to present basic lemmas.

Lemma 2.1. Consider the simplified CNN model. A processor that contributes to N_O elements of Out and accesses N_I elements of Img and N_F elements of $Filter$ can perform at most

$$min(N_I N_F, N_O N_F, N_I N_O)$$

useful scalar multiplications.

Proof. At most $N_I N_F$ multiplications can be performed without duplication. Based on the Definition 2.2, at most $N_O N_F$ multiplications are useful. Since the roles of Img and $Filter$ are symmetric in the simplified CNN model, we also conclude that at most $N_I N_O$ multiplications are useful.

We break the CNN computation into phases, where we know the number of communications in each phase. Then a communication lower bound can be found via analyzing the total number of phases.

Definition 2.3 A phase is a computation sequence for a processor, including useful multiplications, receives and sends. We decompose the schedule of the computation on a processor into contiguous phases. Phase l begins when the total number of words sent and received so far by the processor is exactly lT. Therefore in each phase except perhaps the last phase, the communication cost of the processor, either by sending or receiving, is exactly T words.

Lemma 2.2 Consider the simplified CNN model with a processor that has M words of local memory. The number of useful multiplications the processor can perform during one phase is at most

$$\frac{(M+T)^2}{9}$$

Proof. The total number of N_I, N_F and N_O the processor may access during a phase is at most $M+T$, since each element either uses one word of local memory or it is sent from another processor. With Lemma 2.1 we find the maximum number of useful multiplications performed by solving the following optimization problem:

$$\begin{aligned} maximize \quad & min(N_I N_F, N_O N_F, N_I N_O) \\ s.t. \quad & N_I + N_F + N_O \leq M + T \end{aligned}$$

The objective function reaches the maximum value when $N_I = N_F = N_O = \frac{M+T}{3}$, thus

$$min(N_I N_F, N_O N_F, N_I N_O) \leq (M+T)^2/9$$

which concludes the proof.

Next we will prove a tradeoff between memory and communication for the simplified CNN model.

Lemma 2.3 Consider the simplified CNN model with a processor that performs S scalar multiplications. The total number of words that the processor must send or receive is at least

$$\frac{9S}{4M} - M$$

Proof. The number of scalar multiplications is S, and from Lemma 2.2 we have learnt that the maximum number of useful multiplications can be performed during one phase is $(M+T)^2/9$. Therefore, the number of phases is at least

$$\left\lfloor \frac{9S}{(M+T)^2} \right\rfloor$$

Since the processor receives T words during each full phase, the total amount of communication cost is at least

$$\left\lfloor \frac{9S}{(M+T)^2} \right\rfloor T$$

As $\lfloor x \rfloor \geq x - 1$ for any positive x, we also have

$$\left\lfloor \frac{9S}{(M+T)^2} \right\rfloor T \geq \left(\frac{9S}{(M+T)^2} - 1 \right) T$$

Notice that T is a free variable here. In order to get a tight communication lower bound for large problem size, we want to find the positive T that maximize $f(T) = (\frac{9S}{(M+T)^2} - 1)T$. However, the optimal solution of $f(T)$ is complex. For the purpose of simplicity, we maximize $g(T) = \frac{9ST}{(M+T)^2}$ instead. In terms of large problem sizes, the number of phases is far larger than one, thus the optimal solution for $g(T)$ is also nearly optimal for $f(T)$.

Since $g'(T) = \frac{9S(M-T)}{(M+T)^3}$, $g(T)$ is increasing for $T < M$ and decreasing for $T > M$, and so has a maximum at $T = M$.

Therefore the communication lower bound is

$$\frac{9S}{4M} - M$$

which concludes the proof.

Theorem 2.1. Consider the simplified CNN model on a P-processor distributed-memory parallel computer with M words of local memory per processor. At least one of the processors must communicate at least

$$\frac{9N^2L^2}{4PM} - M$$

The total amount of communication is bounded by

$$\frac{9N^2L^2}{4M} - PM$$

Proof. Since we have P processors, at least one processor has to perform at least $\frac{N^2L^2}{P}$ scalar multiplications. The communication of the processor is obtained by using lemma 2.3 with $S = \frac{N^2L^2}{P}$

Next, we prove the overall communication lower bound. Without losing generality, we assume that the ith processor performs S_i scalar multiplications. Since there are P processors and they need to perform N^2L^2 different scalar multiplications in total, the following inequality is satisfied.

$$\sum_{i=1}^{P} S_i \geq N^2L^2$$

The equality holds only when no duplicate multiplications are performed.

Lemma 2.3 shows that the ith processor has a communication lower bound $\frac{9S_i}{4M} - M$. Adding up the communication cost of each processor we derive the communication lower bound for the whole system.

$$\sum_{i=1}^{P}(\frac{9S_i}{4M} - M) \geq \frac{9N^2L^2}{4M} - PM$$

3 A general CNN model

First of all, we need to define a general form of convolution operation.

Definition 3.1. As shown in [1], the convolution operation can be written as seven nested loops computing Out array from $Image$ array and $Filter$ array as follows (boundary conditions are ignored):

for $b, c, k, w, h, r, s = 0 : \{B, C, K, W, H, R, S\}$ - 1
$Out(k, h, w, b) += Img(r + \sigma_W w, s + \sigma_H h, c, b) \times Filter(k, r, s, c)$

where Img, Out and $Filter$ all have four dimensions. B is the number of images, C is the number of input channels, K is the number of filters, W and H are the width and height of an image, R and S are the sizes of one convolution kernel, σ_W and σ_H are strides.

Theorem and lemmas in Section 2 apply to the general CNN model as well. The only difference is that, for the general CNN model, the total amount of required computation is $BCKWHRS$. Same arguments yield the following communication lower bound:

$$\frac{9BCKWHRS}{4M} - PM$$