OFFICE OF DEAN ACADEMICS

Ph.D. No.11176
December 04, 2023

Subject: **Ph.D. thesis submitted by Saurabh Tewari (2015CSZ8046)**

Dear Prof. Paul and Prof. Kumar,

Reports from all the examiners on the aforesaid Ph.D. thesis have now been received. Both examiners have asked for minor revisions in the report and desired corrections. A detailed response should be submitted by the candidate addressing the examiners' queries at the earliest possible. The same be forwarded by the supervisor(s) after ensuring that the examiners' concerns are addressed.

With regards,

(Dean, Academics) 04/12/23

Prof. Kolin Paul and Prof. Anshul Kumar, CSE

External review of Ph.D. thesis of Mr. Saurabh Tewari

Summary overview

The thesis aims to provide novel energy-efficient solutions for NN accelerators by reducing off-chip memory accesses. Since every memory access reduction technique cannot be applied to all types of neural networks, the author presents different techniques for 3 types of networks: Self Organizing Maps (SOMs), Convolutional Neural networks (CNNs), and Recurrent Neural Networks (RNNs).

CNNs- The author calculates the impact of architectural parameters on the overall off-chip memory access of CNN accelerators and uses that information to determine the optimal tile dimensions to reduce the accesses and the total energy of the CNN accelerators. They further implemented the hardware design for memory-intensive CNN layers on FPGA to measure the off-chip memory accesses, latency, run-time, and design power.

RNNs - The author developed a framework that considers - data shape, tile dimensions, accelerator architecture parameters, and data resolution and computes the off-chip memory accesses of 3D data partitioned into small tiles and proposed a data reuse framework independent of the accelerator's on-chip memory size, making it suitable for LSTM accelerators with small on-chip memory. The proposed approach splits the computations and combines them in a way that significantly reduces the off-chip memory accesses of large matrices. They implemented the hardware design for LSTMs on FPGA for the proposed data-reuse approaches.

SOMs - The author applied quantization techniques on Self Organizing Maps (SOMs) to analyze the impact on NN accuracy and the benefits of improving energy efficiency. They used a custom semi-systolic array design for different bit-width implementations to analyze the accuracy versus energy trade-off for SOMs.

Chapter-wise Summary

Chapter 2: Analyzing Off-chip Memory Accesses

This chapter proposes a new framework for analyzing off-chip memory usage based on the assumption that data movement energy primarily depends on the number of bytes accessed from the memory.

The author proposes an analytical framework that integrates models of NN layers to compute a layer's off-chip memory accesses, data access energy, and the number of

compute cycles for mapping a layer on an NN accelerator. The framework also considers the bus width and data alignment to precisely compute the off-chip memory accesses.

The algorithm computes the total number of bytes accessed from off-chip memory by accumulating the number of bytes accessed for each tile considering the following factors:

⟨Tc, Tr, Tn⟩: Tile dimensions

⟨W, H, N ⟩: Input data shape

δ: Number of overlapped elements between consecutive tiles

The experimental results on popular CNNs, AlexNet, and VGG16, show that the difference between estimated memory access by this framework and measured off-chip memory accesses is less than 4%.

Chapter 3: Optimizing the Performance of CNN Accelerators

This chapter talks about improving the off-chip memory access for Convolutional networks using data reuse. CNN accelerators apply loop tiling to partition the layer data into small tiles that fit into on-chip memory. The chapter claims previous approaches ignored the architectural parameters and address alignment and assume all tiles of the same dimensions have the same off-chip memory accesses due to which the tile dimensions determined by their approaches are suboptimal.

However, the author proposed a new bus width aware (BWA) approach which 1. finds the optimal tile dimensions and 2. suggests the best data reuse scheme by calculating the number of bytes accessed from off-chip memory for each problem. The implementation is configurable for layer shapes, tile dimensions, on-chip memory sizes, bus width, and data resolution.

The data reuse schemes discussed in this thesis are: 1. Input Reuse oriented Scheduling, 2. Output reuse-oriented scheduling and 3. Weight reuse-oriented scheduling.

The section is concluded with the result that their algorithm reduces the off-chip memory accesses of the Convolutional Layers of VGG16 by 16%, 29%, and of AlexNet by 9%, 16% on 64 and 128 bits data bus, respectively, for 8 bits data width, compared to the previous approaches.

Chapter 4: Optimizing Performance of RNN/LSTM Accelerators

LSTM computations involve several large matrix-vector multiplications performed for many time steps. The sizes of these matrices can be significant in several MBs and often exceed the size of the accelerator's on-chip memory. Hence, they are partitioned into blocks and accessed from off-chip memory repeatedly by the accelerator, which results in a large volume of off-chip memory accesses and energy consumption.

The author presents a Split And Combine Computations (SACC) approach that splits the computations of a time step and computes the partial sums of two consecutive time steps.

In the proposed approach, only half of the matrix input R, a weight matrix updated during training is accessed at each time step, reducing the R matrix accesses by half. While previously various methods used quantization and pruning techniques to compress the models' size to fit in the on-chip memory, however, the author claims that their approach is orthogonal to the quantization techniques and can be integrated with different quantization techniques to reduce memory access further.

For a 50% on-chip buffer to matrix size ratio, the SACC approach reduces 32% energy for a 64 KB on-chip buffer and 30% for a 128 KB on-chip buffer size compared to conventional approaches.

Chapter 5: Performance Improvement of SOM by Using Low Bit-Width Resolution

The author explores quantization techniques to improve energy efficiency in the Self Organizing Maps (SOM) network. SOM uses a type of unsupervised learning called the competitive ANN learning model which is used for dimensionality reduction and clustering. The objective of training a SOM network is to embed genomic features of a specific bacteria into a SOM. Each SOM network is trained to recognize only one bacteria. The author further deployed SOM in an FPGA design with Vivado v.2016.4 and calculated area and power numbers for different weight resolutions from the reports generated by the Vivado tool.

The section concludes with an observation that 16-bit fixed-point representation for trained weights provides a good balance between resource utilization, energy efficiency, and accuracy.

Strong Points

The strongest part of the thesis was chapter 4, in which the author proposed an approach to increase the reuse of a large data structure R, so as to improve the performance and reduce the power-consumption of computation. The idea is to split the matrix R diagonally, and then compute the upper and lower parts of the S matrix separately and then combine them together. Once the computation of lower and upper parts of R are separated, they can use the upper part of R for computing half of S of two steps. They further optimized this by blocking the computation of S into smaller blocks. This approach is quite neat. A paper on this topic was published by the authors in DATE 2021 – which is quite commendable. They implemented the hardware design for LSTMs on FPGA for the proposed data-reuse approaches. For a 50% on-chip buffer to matrix size ratio, the SACC approach reduces 32% energy for a 64 KB on-chip buffer and 30% for a 128 KB on-chip buffer size compared to conventional approaches.

The second chapter creates a mathematical model that estimate a layer's off-chip memory accesses, data access energy, and the number of compute cycles for mapping a layer on an NN accelerator. They combine this to make models for the whole NN. They validate this model against some popular models and find an error rate of less than 4%. This is very powerful, since now to design an efficient accelerator, you do not need to synthesize each chip and evaluate it – but you can just evaluate it using these cost models – which can be done at a much higher level, and faster during the design space exploration time.

The third chapter uses the models from the previous chapter to optimize the off-chip memory access of CNN.

The fifth chapter optimizes the power consumption of a SOM (Self-Organizing Map) by quantization. This is the first work that explores the effectiveness of quantization in reducing the power consumption of SOM. They conclude that 16-bit representation provides a trade-off between power and accuracy.

Opportunities for improvement

- Page 3, first paragraph: "Multilayer NNs with more than three layers are referred to as deep neural networks (DNNs)" This definition does not seem to be true, in practice, neural networks with three or more Hidden layers are often considered deep neural networks.

- Page 16, Figure 2.3: The figure is not understandable due to the unavailability of proper axis labels and legends. is supposed to depict the importance of having good tile dimensions and how bad dimensions can affect memory access however it does not clearly depict this idea. Maybe the explanation can be improved.

- Page 33, first paragraph: "In the IRO scheduling scheme, all the operations involving a given ifm tile are scheduled consecutively to access each ifm tile only once from the off-chip memory. There are Ho Wo number of ifm tiles in spatial dimension and Ni number of ifm tiles in Tro Tco Tni depth dimension" - The ORO scheduling scheme is confused with IRO.

- Page 37, first para: "It took less than 60 minutes to determine the optimal solution for VGG16 on Intel Core i7-6700 CPU (@3.40GHz×8" while the thesis focuses on optimizing the offchip memory access during INFERENCE stage on the edge devices, is it optimal to spend ~1 hour deciding optimal tile dimensions for a network?

- Chapter 3 - More elaboration is required on the model/tool that computes B of the CLs using the BWA approach to find the optimal tile dimensions. How is the solution space explored?

- Page 60, first para: "When the on-chip buffer to R matrix size is close to 48%, TSI-WR reduces the memory access by≈25%". The TWI paper claimed memory consumption reduction of 28.4 - 57.3% compared to the conventional methods however the current thesis shows that the TWI approach only reduced 25% of memory access. Why is there a difference in the values?

- The study fails to specify how the accuracy and energy efficiency change when e.g., the neurons are changed <100 or >100. Will 16-bit fixed-point representation still provide a good balance between resource utilization, energy efficiency, and accuracy?

- A representation, or visualization of SOM networks when trained on the data could help readers understand the context and working of SOMs better. Also, more information about the dataset used for training the networks would be useful.

- The training of SOM could have further benefitted from a co-training approach as mentioned in http://www.ijicic.org/ijicic-140626.pdf or initializing weights using PCA instead of random values as PCA can classify gene sequences into groups of known biological categories when relatively small amounts of sequence data were analyzed in advance.

- The thesis would also benefit by a discussion on the scalability and generality of the proposed techniques for different types of NNs and architectures. It would be interesting to discuss how the techniques can be adapted and applied to more complex and modern NN models, such as Transformer-based models and Graph Neural Networks. Can similar performance be expected when applying the SACC approach to these architectures?

Answers to Specific Questions

- **Highlight the points in the thesis, which, in your opinion constitute a significant original contribution to domain knowledge.**

  The approach presented in Chapter 4 – to split the matrix R diagonally, and then internally block it, and pipeline the computation of two steps to increase the reuse of R is neat and effective. This clearly demonstrates a significant and original contribution to the domain knowledge.

- **Identify the aspects of the candidate's work, which demonstrate his/her capacity to carry out independent research.**

The approach presented in Chapter 4 – to split the matrix R diagonally, and then internally block it, and pipeline the computation of two steps to increase the reuse of R is neat and effective. This clearly demonstrates the candidate's capacity to carry out independent research.

- **Point out specific observations made by the candidate which in your opinion, need revision or clarification.**

  Provided in the last section of my comments.

- **Comment on the standard of presentation of the thesis.**

  No comments. I did not attend the thesis presentation.

Thesis Summary

Chapter-1: Gives a generic introduction to the topic and gives overview of the thesis content.

Chapter 2: Presents a framework to calculate off-chip memory accesses for DNNs. The results are presented for different shapes and architectural constraints. Optimal solutions are identified by comparing different data partitioning and scheduling schemes.

Chapter-3:
The chapter presents a method to opitmise memory accesses for CNNs. While tiling, two issues arise: aligned data may need padding of bits because of large bus size compared to data width; and unaligned data requires lot of extra bytes of transfer for a single address. They propose two functions to compute the number of accesses required for 3D data. Comparison with input resuse, output resuse and weight reuse is given.

Chapter-4:
This chapter discusses the performance improvement for RNN and LSTM accelerators.
The idea is: at time step t, St upper and ht-1 are input from previous time step and RL is used to calculate St lower and St+1 lower. St lower is combined with St upper from previous time step to form St. St+1 lower is sent to the next time step to combine with St+1 upper which is calculated in the next time step using R upper. Thus only half the matrix is accessed at each time step, reducing the number of accesses. It divides the R matrix into blocks of size B * B. First qt is calculated and depending on the timestep is even or odd the lower or upper diagonal blocks of R are used to calculate S for t and t + 1 accordingly. It uses extra 4N + 4B on chip memory to store the 4 partial sum along with 4 * B2 memory required in other proposed architecture.

Chapter-5:
The chapter addresses the need for recognition of pathogenic bacteria which is necessary for proper prescription of medication on battery powered devices. The proposal uses Self Organizing Map (SOM) that uses competitive Artificial Neural Network unsupervised learning model, to recognise an unknown bacteria and presents an SiLago and an FPGA implementation of their work. A SOM is trained on one bacteria type, so multiple SOMs are trained for multiple bacteria and during inference/testing, the unknown bacteria is tested against all the SOM networks and the best one is selected. SOM is implemented in MATLAB to calculate the accuracy loss while using fixed point format across different bit widths.

Chapter-6:
Presents conclusion and future research directions

Detailed comments/Questions:

Chapter-2

Section 2.2 how did you compute the stride based tiles? Does delta represent the stride?
In case of overlapped strided accesses, have you accounted for data already fetched, or you compute the fetch time separately for each tile?
Your computation completes the calculation of complete tile (all frames included) at a time?
What wil happen if you calculate for first frame of all and then the second? This way you will have more overlapped accesses and data reuse?

You can include more details of your Xilinx implementation for measurement of accesses.
Elaborate the caption and legend in Fig 2.8
add a summary to chapter-2
What dataset has been used for VGG16, AlexNet etc.?
How does your proposal scale for larger and deeper networks?
What is the percentage margin between estimate and measured values?
Does this margin remain consistent across different network and datasets?

Chapter-3

Discusses the performance improvement for CNNs by doing tiled access to off-chip memory.
Related work only cites two papers. Are there no other works in this topic? How is your tile dimension calculation different from them? Does it give comparatively better results?
Fig 3.7 your proposal gives better performance as but width increases. For smaller bus-width SS and BWA are similar. Can you elaborate the reason?
Fig 3.8 please use some other colour code as the current ones are not distinguishable.
Here BWA performs better than tile size based approach. By tile based are you indicating to your algo in chapter-3 or SS?
How does SS[28] compare in the context of the resuse schemes?
Fig 3.10 the difference in improvement of BWA over SS for different bus widths is less in FC layer compared to what you obtained in te CL layer (fig. 3.7). Elaborate possible reasons.

Chapter-4

Fig 4.9 your policy reuses the values of R by partially loading it and using it over all activation values. This results in less offchip latency for reading inputs. But you would need more space to store the partial output values.
What is the impact on the output buffer size and latency to fetch+add the partial values at the end?
Why do you choose to divide R into a diagonal matrix and not a simple row-wise split? In other words, you could perform computations on first 2 rows then next next row(s)? What is the advantage of diagonal matrix?

Chapter-5

SiLago is designed to be reconfigurable and easy to program. The description is very brief and difficult to follow. BioSOM could have been elaborated more.
What is the existing work in SOM networks?
Are there any benchmark models/datasets for SOM?
Where did you get the training data?
How did you decide the model?

Overall, the thesis has made good contributions. The experiments and the results are satisfactory. The work is also published at reputed venues which demonstrates the relevance of the topic.

The work is sufficient for the award of the degree.