Is Approximation Universally Defensive Against Adversarial Attacks in Deep Neural Networks?

Ayesha Siddique, Khaza Anuarul Hoque
Department of Electrical Engineering and Computer Science
University of Missouri, Columbia, MO, USA
ayesha.siddique@mail.missouri.edu, hoquek@missouri.edu

***Abstract*—Approximate computing is known for its effectiveness in improvising the energy efficiency of deep neural network (DNN) accelerators at the cost of slight accuracy loss. Very recently, the inexact nature of approximate components, such as approximate multipliers have also been reported successful in defending adversarial attacks on DNNs models. Since the approximation errors traverse through the DNN layers as masked or unmasked, this raises a key research question—*can approximate computing always offer a defense against adversarial attacks in DNNs, i.e., are they universally defensive?* Towards this, we present an extensive adversarial robustness analysis of different approximate DNN accelerators (AxDNNs) using the state-of-the-art approximate multipliers. In particular, we evaluate the impact of ten adversarial attacks on different AxDNNs using the MNIST and CIFAR-10 datasets. Our results demonstrate that adversarial attacks on AxDNNs can cause 53% accuracy loss whereas the same attack may lead to almost no accuracy loss (as low as 0.06%) in the accurate DNN. Thus, approximate computing cannot be referred to as a universal defense strategy against adversarial attacks.**

***Index Terms*—Adversarial Attacks, Adversarial Robustness, Approximate Computing, Deep Neural Networks**

I. Introduction

Approximate computing in deep neural networks (DNNs) has recently gained prominence in exploring the accuracy and energy trade-offs for big-data automation [1]. Approximate deep neural networks (AxDNN) accelerators employ inexact full adders [2], truncated carry chains [3], etc., which induce approximation errors in them. Unfortunately, DNNs are susceptible to adversarial attacks [4] and AxDNNs are no exception [5]. This limits their deployment in the safety-critical applications since the adversary may use partial information about the model to craft adversarial examples and exploit transferability property of DNNs [6] [7] to attack AxDNNs. Recent works in defending adversarial attacks are targeted mostly for accurate DNNs [8] [4] and thus, the robustness and defense of AxDNNs is vastly under-explored [9].

Very recently, Guesmi et al. presented approximate computing as an effective structural defense strategy against adversarial attacks by incorporating an array multiplier, with approximate mirror adders instead of exact full adders, in the AxDNN inference phase [5]. Even though this solution opens up a new dimension of research, such defensive behavior of approximate computing cannot be generalized with one AxDNN. This is due to the fact that approximation errors traverse through the AxDNN layers as masked and un-masked and hence, may render their structural defense inconsistent in an adversarial environment. Hence, there is a pent-up need to explore the adversarial robustness of AxDNNs extensively which also includes investigating the impact of adversarial attacks with different perturbation budgets under different attack scenarios. Additionally, it is also required to explore if quantization is supportive towards adversarial defense in AxDNNs since quantization can also improve the robustness in accurate DNNs [10]. Towards this, the research questions that need to be investigated are as follows:

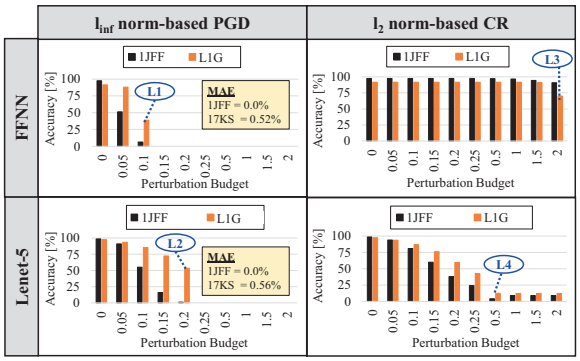

Figure 1: Impact of adversarial attacks on accurate and approximate versions of FFNN and Lenet-5. The accurate and approximate DNNs contain accurate 1JFF and approximate L1G multipliers, respectively, from Evoapprox8b [11] library.

(Q1) Does approximate computing in AxDNNs provide universal defense against adversarial attacks? How does the adversarial robustness of AxDNNs vary with the change in the perturbation budget?
(Q2) Are adversarial attacks transferable from accurate DNNs to AxDNNs irrespective of their difference in exactness and model structure?
(Q3) How does approximate computing react to quantization in AxDNNs under adversarial attacks? Are they supportive or antagonistic to each other?

A. Motivational Case Study and Key Observations

Prior to extensively analyzing the adversarial robustness of AxDNNs, we first present a motivational case study to demonstrate their both defensive and perturbing nature towards

adversarial attacks. We trained a 5-layered convolutional neural network, i.e., Lenet-5, and feed-forward neural network (FFNN) on the MNIST [12] dataset. We replaced their accurate multipliers with approximate counterparts, using Evoapprox8b [11] library, to build two different AxDNNs. We compared the classification accuracy of each resulting AxDNN with its exact counterpart under the l_∞ norm-based projected gradient descent (PGD) and l_2 norm-based contrast reduction (CR) attacks. As shown in Fig. 1, we observe that the accuracy of both AxDNNs is higher than the accurate DNNs in the case of former attack (see label L1 and L2). However, the same approximate FFNN exhibits an opposite behavior in the case of later attack i.e., its accuracy decreases with an increase in the strength of the attack (see label L3). Moreover, a clear drop in accuracy of more than 75% is observed around perturbation budget (ϵ) of 0.5 in the case of approximate Lenet-5 (AxL5) with the later attack (see Label L4). After this value, the accuracy of AxL5 decreases sharply. Such conflicting observations motivated us to extensively analyze the adversarial robustness of AxDNNs.

B. Novel Contributions

This paper makes the following novel contributions:

1) An extensive adversarial approximation analysis to expose the perturbing nature of approximation noise in different adversarial settings with varying perturbation budgets. **[Section IV.B]**
2) A transferability analysis to determine whether the adversarial attacks are transferable from accurate DNNs to AxDNNs irrespective of their difference in exactness and model structure. **[Section IV.C]**
3) An adversarial quantization analysis to determine whether quantization and approximate computing are supportive or antagonistic to each other. **[Section IV.D]**

Since the multipliers consume more energy as compared to other arithmetic units (e.g., adders) [13]; therefore, we employ the state-of-the-art approximate multipliers [11] in AxDNNs. In particular, we explore the impact of 10 different adversarial attacks on approximate Lenet-5 (AxL5) and Alexnet (AxAlx). We use the MNIST [12] and CIFAR-10 [14] datasets for the adversarial robustness analysis. Our results demonstrate that an adversarial attack on AxDNNs may lead to 53% accuracy loss. Conversely, the same attack may lead to almost no accuracy loss (as low as 0.06%) in accurate DNNs. This behavior contradicts the observations in [5]. Our analysis reveals that *AxDNNs are not universally defensive towards the adversarial attacks.* Furthermore, the adversarial attacks are transferable from accurate DNNs to AxDNNs irrespective of their difference in exactness and model structure. *We also observe that approximate computing acts antagonistically to quantization.*

The remainder of this paper is structured as follows: Section II and Section III present a threat model and methodology for analyzing the adversarial robustness of AxDNNs. Section IV presents the results for adversarial robustness analysis of AxDNNs in comparison with accurate DNNs. Finally, Section V concludes the paper.

II. Threat Model

In this section, a threat model is presented for exploring the adversarial robustness of AxDNNs.

A. Adversary's Knowledge

We assume that the adversary uses an accurate classifier model for generating the adversarial examples. The adversary has either (i) partial knowledge about the AxDNN i.e., the model structure is known but inexactness is not known, or (ii) no knowledge about the AxDNN i.e., both model structure and inexactness are not known (see Fig. 2). Since the adversary lacks the information about the inexactness of AxDNNs only in the former case; therefore, it is considered as a special case of transferability. This attack scenario is similar to the black-box attacks in [5].

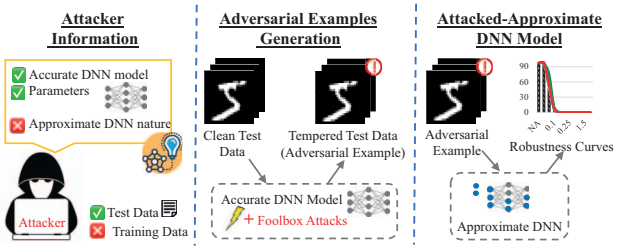

Figure 2: Attack scenario with both model structure and inexactness not known to the adversary

B. Attack Generation

In this paper, the adversary is assumed to be exploratory. The adversary can evade the AxDNN by tampering with the test images, during the inference phase, without influencing the training data. The adversary seeks to craft adversarial examples by finding the perturbation that maximizes the loss of a model on a given sample while keeping the perturbation magnitude lower than a given budget [16]. Table I enlists the gradient and decision-based attacks and the distance metrics used in this paper. The distance metrics such as, l_0, l_2, and l_∞ norms help in approximating the human perception of visual difference. The l_0 norm counts the number of pixels with different values at corresponding positions in the original and perturbed images. The l_2 norm measures the Euclidean distance between two

Table I: Adversarial Attacks [15], Types, and Distance Metrics

Attack Name	Attack Type	Distance Measure
Fast Gradient Method (FGM)	gradient	l_2, l_∞ norm
Basic Iterative Method (BIM)	gradient	l_2, l_∞ norm
Projected Gradient Descent (PGD)	gradient	l_2, l_∞ norm
Contrast Reduction Attack (CR)	decision	l_2 norm
Repeated Additive Gaussian (RAG)	decision	l_2 norm
Repeated Additive Uniform Noise (RAU)	decision	l_2 , l_∞ norm

images. The l_∞ norm measures the maximum difference for all pixels at corresponding positions in two images.

III. Evaluation Methodology

Fig. 3 shows the overview of our methodology for AxDNN's robustness evaluation. It consists of four main steps: accurate DNN training, adversarial examples generation, attacks on AxDNN inference, and percentage robustness. Algorithm 1 delineates these steps. Line 1 and 2 train the accurate DNN with accurate multipliers and check whether the accuracy of the trained model is above the user-defined threshold. In this paper, we consider baseline accuracy as a threshold value. The adversarial robustness analysis of accurate DNN and AxDNNs starts from Line 6. First, the accurate multiplier and different adversarial attacks, with multiple perturbation budgets (ranging from 0 to *p*, where *p* is a set of integers) are used for generating the adversarial examples. Higher is the perturbation budget, the higher is the strength of the adversarial attack. Then, the quantized accurate DNN and AxDNNs with accurate and approximate multipliers, respectively, are evaluated against the adversarial examples. Line 8 verifies if the adversary succeeded in misclassification i.e., forcing the output to an arbitrary false label. If the goal of the adversary is achieved then, the counter of successful attack generation is incremented. Lastly, the robustness is evaluated in Line 15, for every perturbation budget, as the percentage rate of attacks for which the adversary fails to generate an effective adversarial example that fools the victim accurate DNN or AxDNN.

Algorithm 1: Robustness Evaluation

Inputs : Type of multipliers: mults ={ACC, JV3, ...};
Type of adversarial attack: attack = BIM or PGD, etc.
Perturbation budget: eps = [0, p];
Labelled test set: $\mathcal{D} = (X^t, L^t)$;
Quantization level: Qlevel;
Accuracy threshold: Ath
Outputs: Percentage Robustness: Rlevels

```
1:  model = ExactDNNtrain (mults(1))
    // Train DNN with accurate multiplier
2:  if Accuracy(model) ≥ Ath then
3:      for j = 1 : length(eps) do
4:          adv = 0;
5:          for k = 1 : size(D) do
6:              (X_k^{t*}, L_k^{t*}) = AdvExGen (model, mults(1), eps(j), attack, X_k^t)
                // Adv. examples generation with accurate multiplier
7:              Qmdl = FixedPointQuantization (model, Qlevel)
                // Apply fixed point quantization on inference model
8:              (X_k^q, L_k^q) = AdvAttackOnQuanModel (Qmdl, mults eps(j),
                attack, X_k^{t*}, L_k^{t*})
                // Adv. attacks on accurate DNN and AxDNNs
9:              if L_k^q ≠ L_k^t then
10:                 adv++;
11:             else
12:                 NOP;
13:             end if
14:         end for
15:         Rlevels (eps(j)) = (1 - Adv/ size(D)) * 100;
16:     end for
17: end if
```

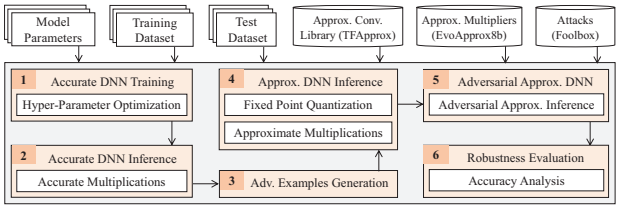

Figure 3: Methodology for analyzing the adversarial robustness of approximate deep neural networks (AxDNNs)

IV. Results and Discussions

This section discusses the experimental setup and the impact of adversarial attacks and quantization on AxDNNs under different adversarial settings (adversarial approximation and adversarial quantization analysis) and transferability of adversarial attacks from accurate DNNs to AxDNNs.

A. Experimental Setup

In this paper, accurate Lenet-5 (AccL5) and Alexnet (AccAlx) architectures are used with their baseline accuracy as 98% and 81%, respectively. The LeNet-5 architecture is comprised of two sets of convolutional and average pooling layers, followed by a flattening convolutional layer, two fully-connected layers, and finally a softmax classifier. The Alexnet architecture contains five convolutional layers, three average pooling layers, and two fully connected layers. For approximate counterparts of these accurate DNNs, the accurate multipliers in the convolutional layers are replaced with approximate unsigned multipliers using Evoapprox8b [11] library. The approximate multipliers are employed in AxL5 and AxAlx according to their error resilience towards the MNIST [12] and CIFAR-10 [14] classification, respectively. For example, the approximate multipliers having accuracy less than 90% in AxL5 and 75% in AxAlx are discarded. The adversarial examples are generated using the Foolbox library [15].

B. Adversarial Approximation Analysis

To investigate the adversarial robustness of AxDNNs, this section discusses the impact of both gradient and decision-based attacks with reference to the first attack scenario in Section II-A.

1) Approximate DNNs under Gradient-Based Attacks: In AxDNNs, both approximation noise and adversarial robustness can be quantified in terms of the mean average error (MAE) of the approximate multipliers. The lower the MAE is, the higher is the actual inference accuracy of AxDNNs (in the absence of adversarial attacks) and hence, higher is their adversarial robustness. For example, Fig. 4 shows that M8-based (MAE = 1.54%) AxL5 exhibits 6% inference accuracy lower than M7-based (MAE = 1.12%) AxL5 in absence of any adversarial attack. Hence, it undergoes 11% more accuracy loss under l_∞ norm-based BIM attack with ϵ = 0.2. It is also observed that two AxDNNs, having the same inference accuracy, behave in a similar fashion under adversarial attacks. For instance, M2

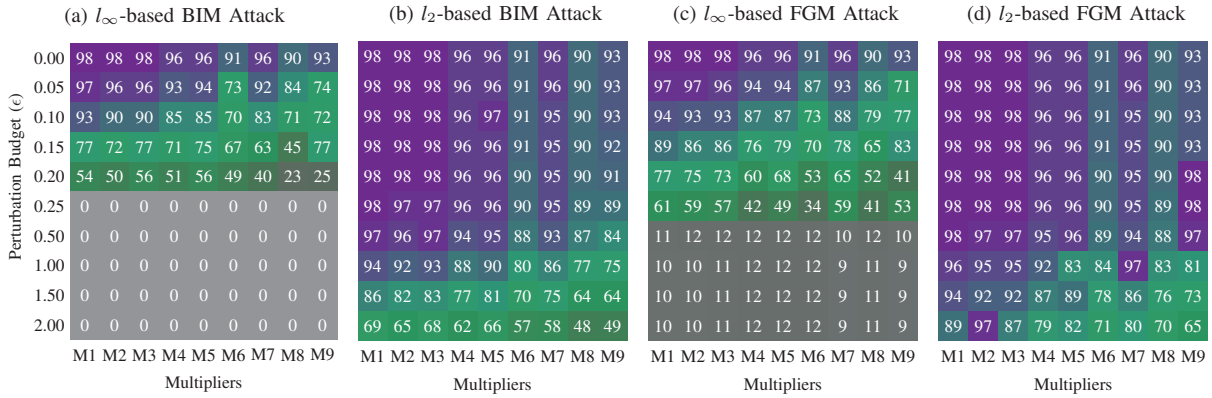

Figure 4: Adversarial robustness of accurate and approximate LeNet-5 under BIM and FGM attacks with the MNIST [12] dataset. The labels M1 to M9 refer to the 1JFF (Accurate), 96D, 12N4, 17KS, 1AGV, FTA, JQQ, L40 and JV3 multipliers in EvoApprox8b [11] library.

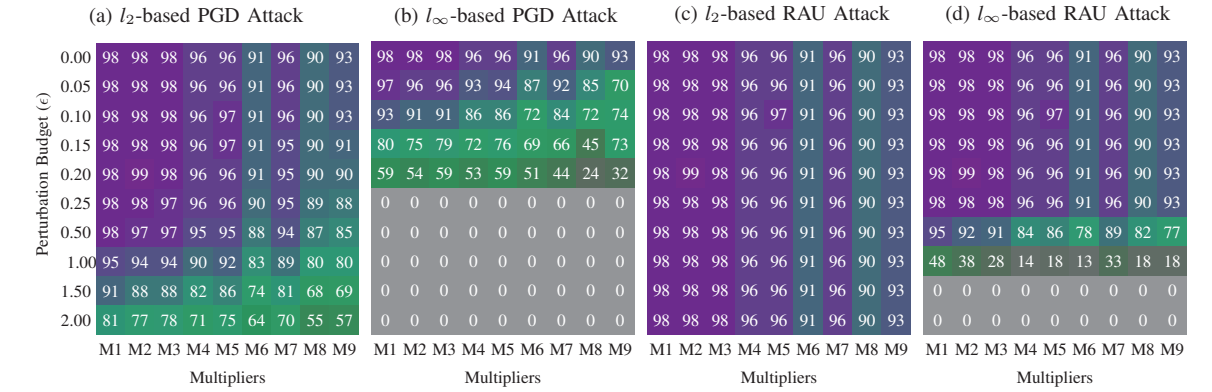

Figure 5: Adversarial robustness of accurate and approximate LeNet-5 under PGD and RAU attacks with the MNIST [12] dataset. The labels M1 to M9 refer to the 1JFF (Accurate), 96D, 12N4, 17KS, 1AGV, FTA, JQQ, L40 and JV3 multipliers in EvoApprox8b [11] library.

and M3-based AxL5 perform close to AccL5 under adversarial attacks due to their same inference accuracy.

The adversarial robustness analysis in Fig. 4 reveals that the accuracy and hence, adversarial robustness of AxDNNs decreases with an increase in the strength of the adversarial attacks, even if the adversary is unaware of the inexact inference engine. For example, l_∞ norm-based BIM attack, with ϵ = 0.2, on AccL5 leads to 44% accuracy loss. However, the same strength of the attack results in 67% accuracy loss in M8-based AxL5 (see Fig. 4a). The same trend is observed with other gradient-based attacks.

It is also observed that AxDNNs, similar to accurate DNNs, exhibit more adversarial robustness under l_2 norm-based attacks as compared to their l_∞-norm based counterparts. This trend is noticeable with all adversarial attacks. For example, l_∞ norm-based FGM attack, with ϵ = 0.25, leads to 37% and 49% accuracy loss in AccL5 and M8-based AxL5, respectively (see Fig. 4c). On the other hand, l_2 norm-based FGM attack, with ϵ = 0.25, causes almost no accuracy loss in both AccL5 and M9-based AxL5 initially (see Fig. 4d). Later, the accuracy of M9-based AxL5 increases with higher values of ϵ e.g., around 0.2 but then, drops again to 65% with ϵ = 2. This small deviating defensive behavior is exceptional and very often observed in AxDNNs *due to data-dependent discontinuity of their approximation-induced errors* [5]. Such discontinuity can be referred to masking and non-masking of erroneous approximation bits which can traverse through AxDNN layers.

Interestingly, Fig. 4d shows a 28% accuracy loss in M9-based AxL5 but only 9% accuracy loss in AccL5 is observed under l_2 norm-based FGM attack with ϵ = 2. Likewise, Fig. 5a shows a 36% accuracy loss in M9-based AxL5 but 17% accuracy loss only in AccL5 is observed under the l_2 norm-based PGD attack with ϵ = 2. Such a trend is also observed with small perturbation budgets. For example, Fig. 5b illustrate that l_∞ norm-based PGD attack with even ϵ = 0.05 leads to 23% accuracy loss in M9-based AxL5 but only 1% accuracy loss in AccL5. This identifies the non-defensive nature of AxDNNs under adversarial attacks. This observation *contradicts the defensive approximation* in [5], where approximate computing was rendered defensive with such adversarial attacks. Furthermore,

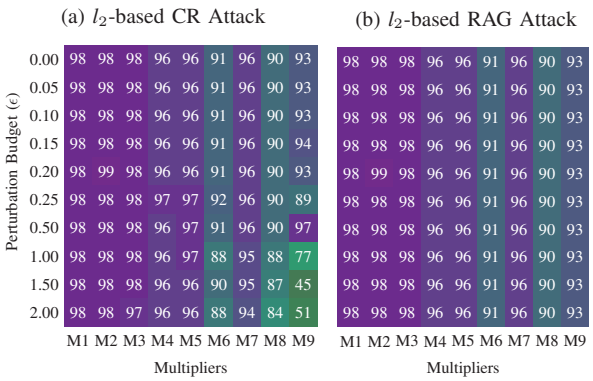

Figure 6: Adversarial robustness of accurate and approximate LeNet-5 under CR and RAG attacks with the MNIST [12] dataset. The labels M1 to M9 refer to the 1JFF (Accurate), 96D, 12N4, 17KS, 1AGV, FTA, JQQ, L40 and JV3 multipliers in EvoApprox8b [11] library.

the BIM (see Fig. 4a and Fig. 4b) and PGD attacks (see Fig. 5a and Fig. 5b) seem to have more impact on the adversarial robustness of both AccL5 and AxL5 in comparison to the FGM attacks. In the case of AxAlx, the CIFAR-10 [14] classification under gradient-based attacks shows that their adversarial robustness is very close to AccAlx. Therefore, these results are excluded from the paper.

2) Approximate DNNs under Decision-based Attacks: The decision-based attacks affect the adversarial robustness of both AxDNNs and accurate DNNs. For example, Fig. 5d shows that l_∞ norm-based RAU attack, with $\epsilon = 1$, leads to 50% accuracy loss in AccL5. However, the same attack leads to 75% accuracy loss in M8-based AxL5. The same trend is observed in the case of other decision-based attacks. It is also noticed that l_2 norm-based decision-based attacks are comparatively less perturbing. However, their impact on the accuracy of AxDNNs is higher with an increase in the approximation noise in AxDNNs. Interestingly, l_2-based CR attack in Fig. 6a undergoes almost no accuracy loss (as low as 0.06%) in AccL5 in spite of high perturbation budget i.e., $\epsilon = 1.5$. The same attack has 53% accuracy loss in M8-based AxL5. Small deviations in these trends are observed with decision-based attacks as well but they do not cause significant changes in the accuracy. Moreover, a similar adversarial robustness trend is observed with CIFAR-10 [14] classification in the case of decision-based attacks on Alexnet as shown in Fig. 7. AxAlx performs close to AccAlx but the impact of decision-based attacks is more noticeable in l_∞ norm-based RAU attack (see Fig. 7d).

C. Transferability Analysis

Adversarial examples are known for their transferability [17], which means that they are transferable from one model to another model. This also refers to the fact that it is possible to attack models to which the attacker does not have access [7]. Therefore, for further analyzing the adversarial robustness of AxDNNs, we craft the adversarial examples using accurate DNN models (second attack scenario discussed in Section II-A) and evaluate their impact on AxDNNs with different model structures. Our results show that the adversarial attacks are more transferable if the adversary is not aware of both the inexactness and type of DNN model used in the inference engine. For example, Table II shows that l_∞ norm-based BIM attack (ϵ = 0.05), strongest attack in previous section, is more transferable from AccL5 to AxAlx when compared to AxL5, and AccAlx to AxL5 when compared to AxAlx. The same trend is observed with both MNIST [12] and CIFAR-10 [14] datasets.

Table II: Transferability Analysis with l_∞ norm-based BIM attack (ϵ = 0.05). X/Y represents accuracy before/after attack

DNN Models	MNIST [12]		CIFAR-10 [14]	
	AxL5	AxAlx	AxL5	AxAlx
AccL5	98/97	67/43	54/9	53/4
AxAlx	98/9	67/11	54/20	53/10

D. Adversarial Quantization and Approximation

Fig. 4 - Fig. 7 present AxDNNs which employ approximate computing along-with quantization. From their comparison with 8-bit quantized accurate DNNs in Fig. 8, we observe that quantization improves the adversarial robustness [10]. However, approximate computing does not support this behavior. The classification accuracy of AxDNNs decreases with an increase in the strength of the adversarial attacks in spite of employing quantization. For example, under l_∞ norm-based PGD attack (ϵ = 0.2), the quantization increases the accuracy of non-quantized AccL5 by 58% in Fig. 8 (see label L5). Conversely, approximate computing decreases the accuracy of quantized AccL5 by 35% in M8-based AxL5 under the same attack in Fig. 5b. The self-error-inducing nature of approximate computing degrades the performance of quantized DNN models and hence, leads to successful adversarial attacks. Thus, approximate computing acts antagonistically to quantization under the adversarial attacks.

Summary. Most state-of-the-art AxDNNs employ approximate multipliers for reducing their energy consumption [2] [13] [18]. These approximate multipliers either have approximate partial products generation or addition. However, approximation error in both cases depends on the *specific* input bit combinations [19]. Recent work of Gusemi et al. [5] exploited such approximation behavior in AxDNN for the defense against the fixed strength of the adversarial attacks. However, such defensive nature of AxDNNs is not always observable as shown by our experimental results in Section IV. In a real-world scenario, the adversary can vary the perturbation budget which may lead to successful misclassification. Interestingly, AxDNNs are not only vulnerable to higher perturbation budgets but also to very small perturbations budgets in some cases, such as ϵ =0.05 as shown in our experimental results. Note, an attack with such a small perturbations budget can be stealthy enough to bypass the attack detection techniques and remain imperceptible to the human eye.

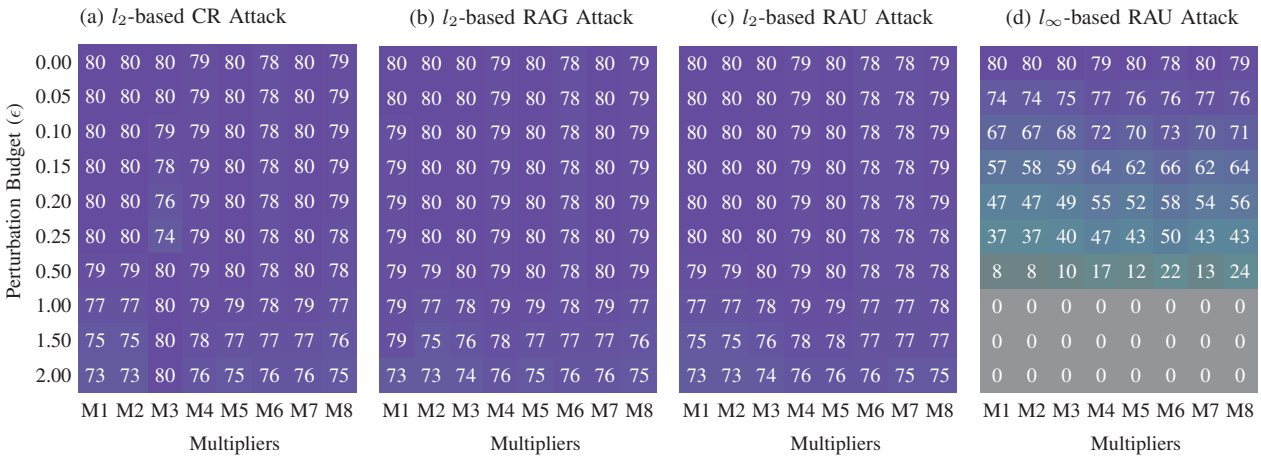

Figure 7: Adversarial robustness of accurate and approximate Alexnet under CR, RAG and RAU attacks with the CIFAR-10 [14] dataset. The labels M1 to M9 refer to the 1JFF (Accurate), 2P7, KEM, 150Q, 14VP, QJD, 1446 and GS2 multipliers in EvoApprox8b [11] library.

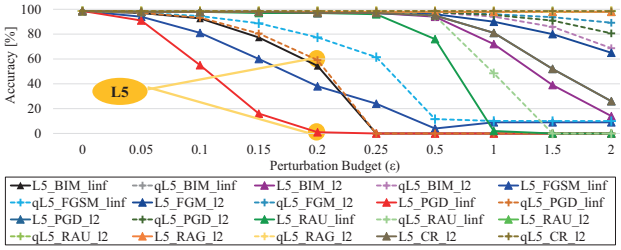

Figure 8: Adversarial robustness of quantized (qL5) and non-quantized accurate Lenet-5 (L5) using the MNIST [12]

V. Conclusion

In this paper, we explored the adversarial robustness of AxDNNs, using the state-of-the-art unsigned approximate multipliers, with the MNIST and CIFAR10 datasets. We empirically show that an adversarial attack on AxDNN leads to 53% accuracy loss. Conversely, the same attack leads to almost no accuracy loss (as low as 0.06%) in accurate DNNs. We observe that approximate computing can reduce the adversarial robustness in spite of quantization in AxDNNs and partial knowledge of the adversary about the model structure. In summary, this work answers three research questions in Section I as follows:

(A1) Though AxDNNs often exhibit mere defensive behavior; this trend is not universal (or consistent). Their adversarial robustness decreases with an increase in the perturbation budget and occasionally, surpasses the accurate DNNs. They are *not universally defensive* in nature towards the adversarial attacks.
(A2) The adversarial attacks are *transferable from accurate DNNs to AxDNNs* even if the adversary has partial knowledge about their inexactness and model structure.
(A3) The quantization and approximate computing act *antagonistic* to each other in an adversarial environment.

References

[1] A. Siddique et al., "Exploring fault-energy trade-offs in approximate DNN hardware accelerators," in ***ISQED***. IEEE, 2021, pp. 343–348.
[2] M. Riaz et al., "CAxCNN: Towards the use of canonic sign digit based approximation for hardware-friendly convolutional neural networks," *IEEE Access*, vol. 8, pp. 127 014–127 021, 2020.
[3] M. A. Hanif et al., "CANN: Curable approximations for high-performance deep neural network accelerators," in ***DAC***. IEEE, 2019, pp. 1–6.
[4] Q. Xu et al., "Security of neural networks from hardware perspective: A survey and beyond," in ***ASP-DAC***. IEEE, 2021, pp. 449–454.
[5] A. Guesmi et al., "Defensive approximation: securing CNNs using approximate computing," in ***ASPLOS***, 2021, pp. 990–1003.
[6] N. Papernot et al., "Practical black-box attacks against machine learning," in ***ASIACCS***, 2017, pp. 506–519.
[7] A. Demontis et al., "Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks," in ***USENIX***, 2019, pp. 321–338.
[8] X. Wang et al., "DNNGuard: An elastic heterogeneous dnn accelerator architecture against adversarial attacks," in ***ASPLOS***, 2020, pp. 19–34.
[9] M. Capra et al., "An updated survey of efficient hardware architectures for accelerating deep convolutional neural networks," *Future Internet*, vol. 12, no. 7, p. 113, 2020.
[10] F. Khalid et al., "QuSecNets: Quantization-based defense mechanism for securing deep neural network against adversarial attacks," in ***IOLTS***. IEEE, 2019, pp. 182–187.
[11] V. Mrazek et al., "Evoapprox8b: Library of approximate adders and multipliers for circuit design and benchmarking of approximation methods," in ***DATE***. IEEE, 2017, pp. 258–261.
[12] Y. LeCun et al., "Mnist handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.
[13] A. Marchisio et al., "Red-cane: A systematic methodology for resilience analysis and design of capsule networks under approximations," in ***DATE***. IEEE, 2020, pp. 1205–1210.
[14] A. Krizhevsky et al., "Learning multiple layers of features from tiny images," 2009.
[15] J. Rauber et al., "Foolbox: A python toolbox to benchmark the robustness of machine learning models," *arXiv preprint arXiv:1707.04131*, 2017.
[16] F. Khalid et al., "TrISec: training data-unaware imperceptible security attacks on deep neural networks," in ***IOLTS***. IEEE, 2019, pp. 188–193.
[17] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," *arXiv preprint arXiv:1611.02770*, 2016.
[18] O. Spantidi et al., "Positive/negative approximate multipliers for DNN accelerators," *arXiv preprint arXiv:2107.09366*, 2021.
[19] S. Mazahir et al., "Probabilistic error analysis of approximate adders and multipliers," in *Approximate Circuits*. Springer, 2019, pp. 99–120.