Dynamic Heterogeneous Voltage Regulation for Systolic Array-Based DNN Accelerators

Jianhao Chen*, Joseph Riad*, Edgar Sánchez-Sinencio†, and Peng Li‡
*†Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA
‡Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA, USA
Email: *{chenjh, joseph.riad}@tamu.edu, †sanchez@ece.tamu.edu, ‡lip@ucsb.edu

***Abstract*—With the growing performance and wide application of deep neural networks (DNNs), recent years have seen enormous efforts on DNN accelerator hardware design for platforms from mobile devices to data centers. The systolic array has been a popular architectural choice for many proposed DNN accelerators with hundreds to thousands of processing elements (PEs) for parallel computing. Systolic array-based DNN acceslerators for datacenter applications have high power consumption and non-uniform workload distribution, which makes power delivery network (PDN) design challenging. Server-class multicore processors have benefited from distributed on-chip voltage regulation and heterogeneous voltage regulation (HVR) for improving energy efficiency while guaranteeing power delivery integrity. This paper presents the first work on HVR-based PDN architecture and control for systolic array-based DNN accelerators. We propose to employ a PDN architecture comprising heterogeneous on-chip and off-chip voltage regulators and multiple power domains. By analyzing patterns of typical DNN workloads via a modeling framework, we propose a DNN workload-aware dynamic PDN control policy to maximize system energy efficiency while ensuring power integrity. We demonstrate significant energy efficiency improvements brought by the proposed PDN architecture, dynamic control, and power gating, which lead to a more than five-fold reduction of leakage energy and PDN energy overhead for systolic array DNN accelerators.**

I. Introduction

Neural networks have been adopted in many disciplines including image and pattern classification. Recent advancement in machine learning (ML) has made convolutional neural networks (CNNs) one of the most successful and widely adopted models for image classification and video analysis. Since the introduction of the AlexNet deep neural network (DNN) model [1] and its successful training on graphic processing units (GPUs), many large-scale DNNs have been proposed for which high-performance general-purpose GPUs (GPGPUs) are routinely deployed to accelerate both inference and training. While having produced promising results in many application domains, modern DNNs have grown to a very large size and can routinely have a deep architecture with hundreds of layers, resulting in a huge number of MAC operations during inference and training. For example, as the first DNN architecture that exceeds human ability in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [2], the ResNet DNN model has up to 152 layers and requires 11.3 billion FLOPs to classify an image [3]. High computational and energy cost required by inference and training hinders the deployment of large DNNs.

The development of dedicated hardware DNN accelerators offers a viable solution to addressing the computational and energy dissipation challenges that come with deep learning. Among the proposed deep learning hardware accelerator architectures, systolic array [4] has been adopted in many industrial designs such as Google's Tensor Processing Unit (TPU) [5], NVidia's Tensor Cores, and ARM's ML Processor due its low complexity, high compute intensity, and high data distribution bandwidth. This hardware architecture comprises multiple interconnected processing elements (PEs) by which the multiply-accumulate (MAC) operations in CNNs can be accelerated in a parallel fashion. The optimization of systolic array hardware accelerator architectures has been targeted in prior work. For instance, Scale-Sim [6] is a systolic array CNN accelerator simulator which provides cycle-accurate memory access traces and estimated bandwidth requirements and can be used for design space exploration. [7] proposes a framework called ThUnderVolt which enables aggressive voltage underscaling of systolic arrays to save power.

This work primarily concerns of high energy dissipation of systolic array-based DNN accelerators in datacenters and cloud computing. In these settings, DNN accelerators may be commissioned to support computationally-intensive machine learning and data analytics applications. A large amount of power must be properly delivered to these DNN accelerators via a power delivery network (PDN) across the circuit board and the accelerator chip, posing major challenges to the PDN design. PDN design in general presents a major challenge to the development of a wide variety of processor systems including server-class multi-core processors and large-scale systems-on-a-chips (SoCs) which burn large amounts of power. While delivering high power, an improperly designed PDNs may incur large losses, hence compromising the overall system's energy efficiency, and fail to sufficiently suppress power supply noise, jeopardizing power delivery integrity and causing timing errors.

Joint optimization of efficiency and integrity of power delivery is a complex matter; it is compounded by limited pin connections to off-chip voltage regulation modules, additional power loss caused by power distribution and voltage regulators, and large spatiotemporally non-uniform DC/transient load currents, which can inject severe noise to the power/ground rails of the on-chip devices.

Due to the increased proximity between voltage regulators and on-chip current loads, integration of multiple distributed on-chip voltage regulators has been shown to be promising for addressing the efficiency and integrity challenges of power delivery as demonstrated in Intel's Haswell [8] and IBM's POWER8 [9] processors. Going beyond distributed on-chip voltage regulation, heterogeneous voltage regulation (HVR) offers rich adaptability and rapid response to fast changing current loads by utilizing voltage regulators of distinct and complimentary characteristics in terms of form-factor, response time, and conversion/regulation efficiency [10]. Multiple voltage conversion/regulation stages comprising efficient off-chip and on-chip switching converters and area-efficient on-chip linear regulators enable highly adaptive power delivery. Such HVR systems can be adapted by a workload-aware control policy [10] which extends the commonly used dynamic power management techniques [11]–[14].

This paper presents the first work on HVR based PDN architecture and control for systolic array DNN accelerators. We propose to employ a PDN architecture comprising heterogeneous on-chip and off-chip voltage regulators and multiple power domains, specifically optimized for the targeted systolic array-based accelerators. By analyzing patterns of typical DNN workloads via a modeling framework, we propose a workload-aware dynamic PDN control policy to maximize system energy efficiency while ensuring power delivery integrity. We demonstrate significant energy efficiency improvements brought by the proposed PDN architecture, dynamic PDN control, and power gating leading to a more than 5-fold reduction of leakage energy and PDN energy overhead for systolic array DNN accelerators.

II. Systolic Array (SA) based Accelerator Modeling

A. Overview

We focus on the systolic array architecture as shown in Fig. 1, which has been used in many DNN accelerator designs [5], [15], [16]. It comprises an array of processing elements (PEs) capable of performing MAC operations. Systolic arrays enjoy low complexity and high compute intensity and can be readily leveraged to parallelize matrix-matrix multiplications, which are the dominant computation in a DNN. The data are fed from the edges of the array and propagate to the target PEs via unidirectional data links. Based on a chosen dataflow design, the PEs may store incoming input/weight data or intermediate results, i.e. partial sums (Psums), in their local scratchpad memories (e.g. register files) to exploit data reuse, leading to reduced bandwidth requirement and energy dissipation.

B. Dataflow modeling

The dataflow of a systolic array-based DNN accelerator determines how the MAC operations in a neural network layer are mapped to PEs of the systolic array, how input/weight data are fetched and reused, and how output data, e.g. Psums, are stored. We consider two commonly explored types of dataflows: output stationary (OS) and weight stationary (WS), where the output or filter weight data are kept stationary in the array to maximize their reuse, respectively [15].

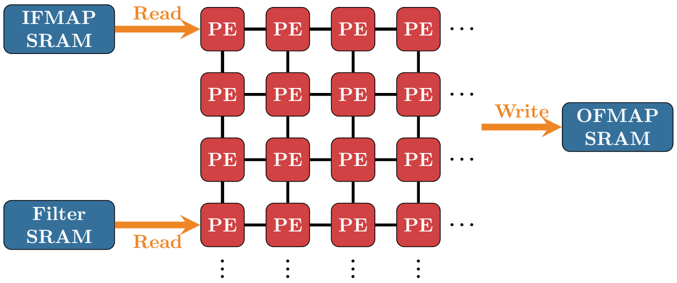

Fig. 1. Architecture of systolic array-based DNN accelerators.

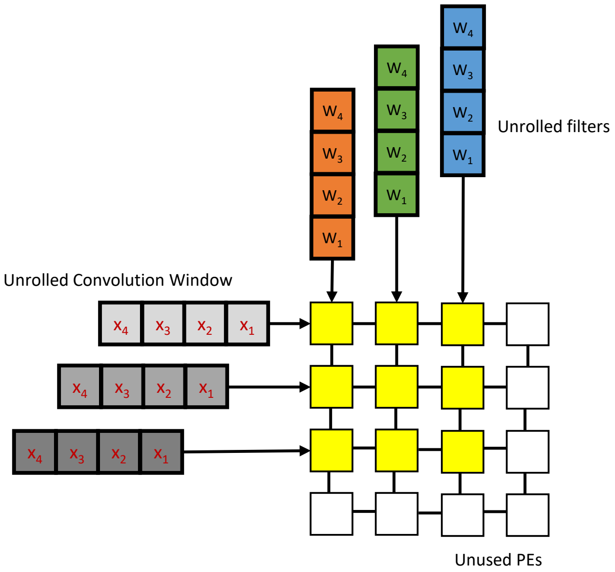

Fig. 2. A systolic array with output stationary (OS) dataflow.

1) Output stationary: The OS dataflow is shown in Fig. 2. To enable filter weight and input feature map (IFMAP) data reuse, the computation of a pixel in the output feature map (OFMAP) is assigned to a PE according to the location of PE in the array and the position of the pixel in the OFMAP. PEs in the same row compute the outputs for pixels at the same positions but in different channels so that they share the same convolution window in the IFMAP. The PEs in the same column compute the outputs for pixels in the same channel of the OFMAP but at different positions so that they share the same filter data.

Because the data can only be fed from the edges of the array and propagate across the array via unidirectional links, there is one cycle delay between the arrivals of adjacent IFMAP data and filter data stream. The activation of PEs starts from the top left corner and ripples down to the bottom right corner of the array.

2) Weight Stationary: In the WS dataflow, each PE keeps the weight data associated with one pixel location in a filter stationary as shown in Fig. 3. The PEs in the same row are assigned with the pixels at the same position and in the same channel of different filters. The PEs in the same column are assigned with the pixels in the same filter but at different positions. Therefore, the PEs in the same row share the same data stream in the IFMAP. After finishing all the multiplications associated with the assigned pixels and generating all Psums, the PEs are assigned with other pixels in a filter if the processing of the current layer is not completed. As shown in the figure, the weight data are fed to the assigned PEs from the top of the array first. Then, the data streams of the IFMAP are fed from the left side of the array from which the PEs start to compute.

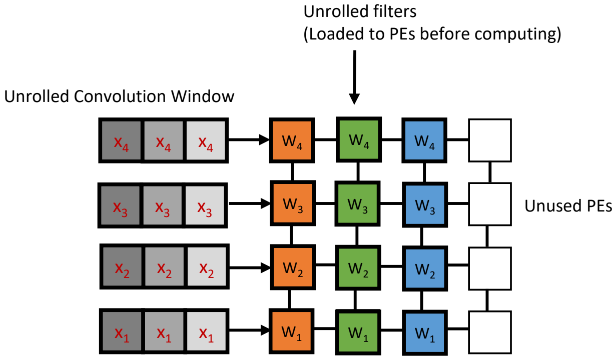

Fig. 3. A systolic array with weight stationary (WS) dataflow.

3) PE Utilization: As discussed previously, both dataflows allow for data reuse based on assigning certain MAC operations to specific PEs in the array, which may limit the utilization of the PEs. Fig. 2 and Fig. 3 show that it is possible for certain PEs to be unused. In the OS dataflow, each column corresponds to a filter and each row corresponds to the same position in different channels of the output feature map. Therefore, the maximal number of utilized columns is bound by the number of filters and the maximal number of active rows is bound by the dimension of the output feature map. However, in the WS dataflow, each row corresponds to pixels at the same position and the same channel in different filters. Hence, the maximal number of active rows is bound by the number of pixels in a filter. What is common to both dataflows is that each column corresponds to a filter, and therefore, the maximal number of active columns is always bound by the number of filters in a given layer.

C. Power modeling

At the architecture level, the energy consumption of CPUs and GPUs can be commonly evaluated using an architectural simulator and a set of power models. For example, architecture simulators GEM5 [17] and GPGPU-Sim [18] provide statistics of hardware utilization, which may be fed into architecture-level power models such as McPAT [19], Watch [20], GPUWattch [21] to generate power dissipation data. To characterize the power/energy dissipation of the targeted dedicated systolic array DNN accelerators, we follow a similar set of steps to first generate architecture-level utilization traces and then use architectural power models to estimate power/energy dissipation. The overall work flow is based on integrating and adapting the SCALE-Sim simulator [6] as shown in Fig. 4.

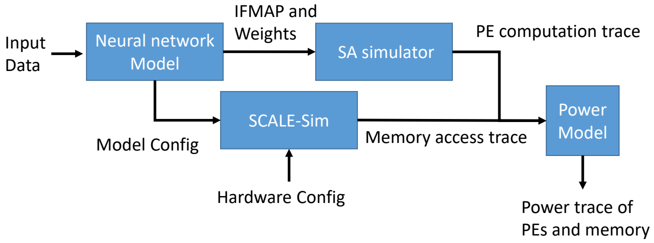

Fig. 4. The power modeling flow for SA hardware accelerators.

SCALE-Sim [6] is an architecture-level simulator for CNN accelerators with its main focus on design space exploration and estimation of data bandwidth requirements. It produces memory access trace based on the configuration of the accelerator hardware under simulation and the CNN model to be accelerated. SCALE-Sim only models the convolutional layers and fully connected layers which may contribute to more than 80% of the total computations in a CNN [6], [22], [23]. One limitation of SCALE-Sim, however, is that it does not provide PE utilization traces. We developed an architecture-level systolic array simulator to generate the operation traces for PEs.

The power modeling work flow starts from the inference of the targeted CNN model to extract the input feature maps and filter data of all its layers. The computation trace of each PE in the systolic array is generated by running our SA simulator based on the CNN's IFMAP and filter data and the assumed dataflow, i.e. WS or OS. The PE computation traces from the SA simulator are combined with the memory access traces from SCALE-Sim and fed into several power models for power trace generation.

CACTI [24] is an architecture-level power, area and timing modeling framework for SRAM based structures, which is used in this work for power estimation of on-chip memory. We adopt approaches similar to ones in [22] and [23] to estimate the PE energy dissipation by:

$$E_{comp} = N_{comp} \times e_{comp},$$

where E_{comp} is the total energy spent on computation, N_{comp} is the number of MAC operations, and e_{comp} is the power consumption per MAC operation. As in [15], we assume that each PE can skip trivial computation with zero inputs that leads to a zero output to save power. Apart from dynamic power, leakage power contributes a large portion of the total chip power at advanced technology nodes. 40% of the total chip power consumption is considered leakage power based on the estimation in [25]. We adopt the published power data from

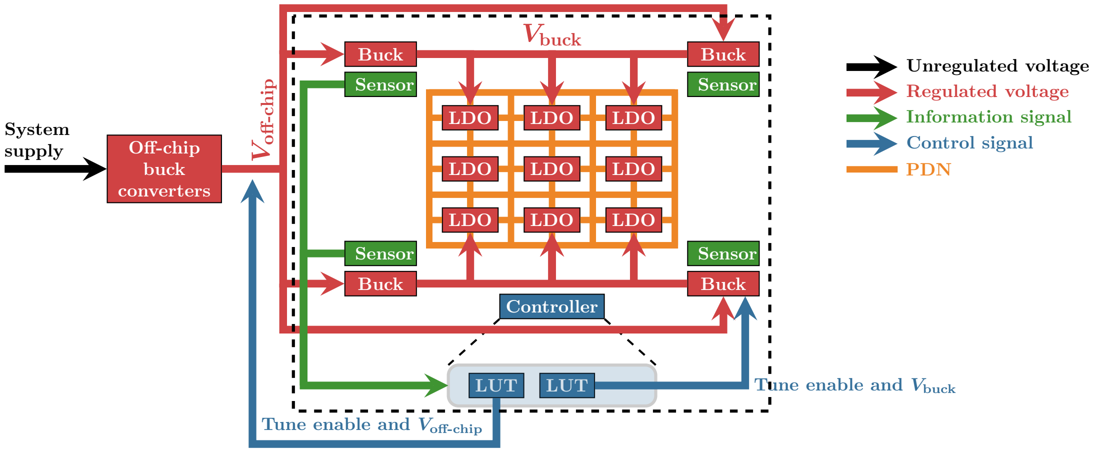

Fig. 5. The three-stage hybrid voltage regulation architecture.

[15] to estimate the power dissipation of one MAC operation and the chip leakage power.

III. Hybrid Voltage Regulation

Fig. 5 shows a conceptual block diagram of a general three-stage hybrid voltage regulation (HVR) architecture [10]. Our goal is to adapt it for robust and efficient power delivery for SA DNN accelerators. The HVR system consists of a cluster of off-chip buck converters and on-chip voltage regulators and the on-chip PDN (power grids). Power-efficient off-chip buck converters are to downconvert the system supply voltage to a level suitable for chip-level processing. The on-chip PDN is powered by multiple area-efficient low-dropout voltage regulators (LDOs) distributed across the chip, acting as the third stage of voltage regulation for ensuring minimum power supply ripple for the on-chip circuits. To improve the overall system efficiency, a second stage of voltage conversion based on a cluster of on-chip buck converters is added, allowing more tunablity of the PDN and helping reduce the power loss of LDOs by decreasing the voltage drop across the LDOs' power transistors. The power efficiency characteristics of various voltage regulators are characterized offline and stored in on-chip lookup tables (LUTs). Activity counter based power sensors are deployed on-chip to estimate the load current to the PDN.

Fig. 6 shows the control flow of the three-stage HVR architecture. The estimated load current is relayed to an on-chip controller that uses pre-characterized efficiency LUTs of the voltage regulators to optimally set the numbers of active off-chip and on-chip buck converters, N_{off} and N_{on} respectively, and their output voltages to maximize the energy efficiency of the entire PDN while meeting a specified power integrity constraint. Control commands are issued based on two control cycle periods: $T_{off} = 100us$ and $T_{on} = 1us$. In each T_{off} control cycle, the optimal values N_{off} and off-chip buck converter output voltage are updated based on the current load estimated from the previous control cycle. At the inner loop of the control and during each shorter T_{on} control cycle, the optimal values for N_{on} and the output voltage of the on-chip buck converters (V_{buck}), which is also the input voltage to the on-chip LDOs, are set.

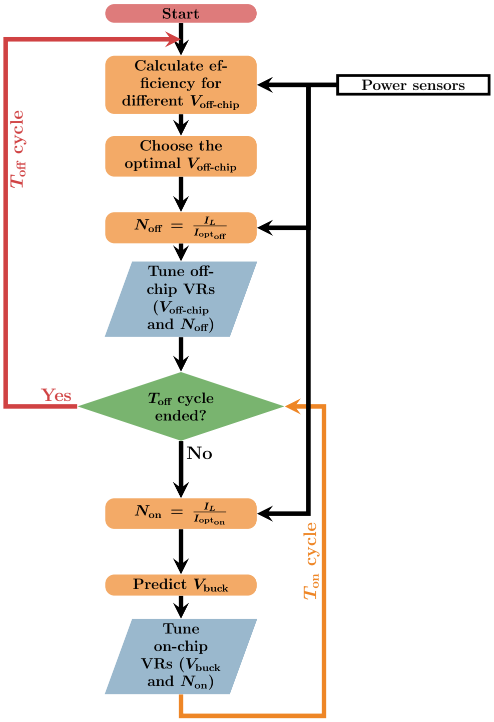
Start
Calculate efficiency for different $V_{\text{off-chip}}$
Power sensors
Choose the optimal $V_{\text{off-chip}}$
$N_{\text{off}} = \frac{I_L}{I_{\text{opt}_{\text{off}}}}$
Tune off-chip VRs ($V_{\text{off-chip}}$ and N_{off})
T_{off} cycle ended?
Yes
No
T_{off} cycle
$N_{\text{on}} = \frac{I_L}{I_{\text{opt}_{\text{on}}}}$
Predict V_{buck}
Tune on-chip VRs (V_{buck} and N_{on})
T_{on} cycle

Fig. 6. Control flow for the 3-stage HVR architecture

IV. PDN Design for SA-based DNN Accelerators

We discuss how to adapt the general HVR architecture by introducing DNN workload aware power gating to significantly reduce leakage energy of the targeted accelerators, and modify dynamic control policy to address systolic-array specific power integrity challenges.

A. DNN workload aware power gating

As discussed in Section II-B3, the PE utilization of a systolic array-based DNN accelerator depends on the dataflow and the sizes/numbers of input feature maps, output feature maps, and filters. Hence, the PE utilization can vary widely from layer to layer of a CNN. Shutting down the unused PEs in the array can save a significant amount of leakage power of the unused PEs. As a commonly adopted low-power design technique, however, power gating may come with large area and power overheads due to reasons such as inclusion of gating transistors and gating control, particularly at a fine-grained level.

Fortunately, due to the regularity and predictability of DNN workloads on a systolic array, coarse-grained power gating can be realized to achieve large leakage saving with low overhead. The key issue is to partition a given systolic array into multiple power domains in a way cognizant of the intended dataflows. As described in Section II-B3, in both WS and OS dataflow we modeled in this work, the maximal numbers of active PE columns in the array are bound by the number of filters in a convolutional layer. The number of filters in each layer of commonly adopted CNN models is a power of 2. For example, the convolution layers in the ResNet-50 model have 2^k filters, where k ranges from 6 to 11. It becomes natural to divide the array into a small number of power domains with each containing the same number of 2^m columns, where $m < k$, as shown in Fig. 7. Whenever the number of filters in a layer is less than the width of the systolic array, it may become possible to turn off several power domains to save leakage power. As we will show in Section V-C, having no more than 8 power domains is sufficient to gain substantial leakage power saving for practical deep CNN models.

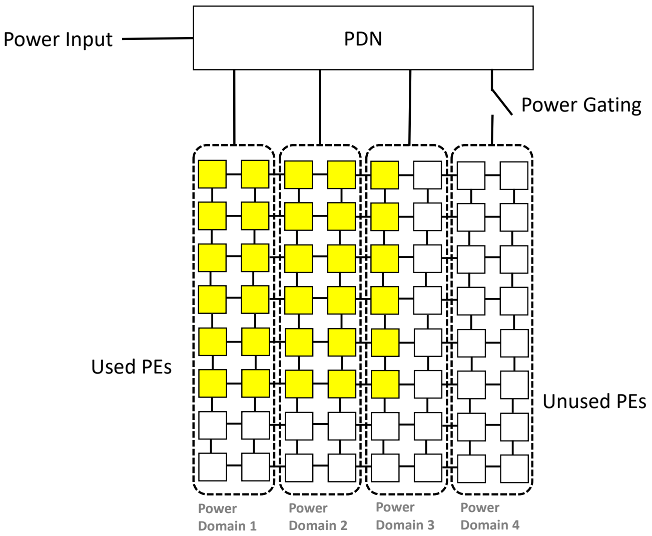

Fig. 7. Partition the systolic array into multiple power domains.

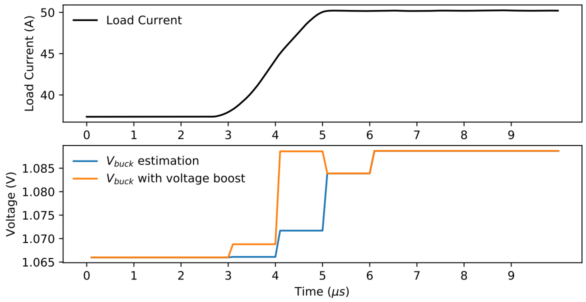

Fig. 8. Proposed LDO input-voltage boost mechanism during the startup phase.

B. Dynamic control for power integrity assurance

The current load estimation accuracy is a key to the success of the HVR dynamic control policy described in Section III. It has been demonstrated in [10] that HVR can lead to large energy efficiency gains when equipped with an enhanced current load estimation accuracy provided by additional on-chip power supply noise sensors and an integrated machine learning (ML) predictor while ensuring power integrity.

In this work, we avoid the additional overheads by dropping the ML predictor and power noise sensors and leverage the predictability of DNN workloads to address a unique power integrity challenge introduced by the SA based WS or OS dataflows. In each of these dataflows, the PEs in the systolic array are sequentially activated either from the top-left corner or the left edge of the SA during the startup phase of the processing of a convolutional layer. As a result, large current load surge may occur within one clock cycle as more PEs are activited. Without accurate ML based load prediction, the dynamic control policy may simply underestimate the required input voltage to the on-chip LDOs required for ensuring power integrity based on the current load estimated from the previous control cycle.

To tackle the above problem with low overhead, we make use the fact that the distribution of the DNN workload over the systolic array can be well predicted in advance by interacting with the system control logic that initializes the processing of each neural network layer during the startup phase. As such, during each startup phase, the on-chip LDO input voltage (V_{buck}) is boosted based on a safe estimation of the amount of current surge.

An example of the proposed LDO input voltage boost is shown in Fig.8. The blue curve is the optimal V_{buck} calculated by the dynamic control policy which acts on the estimated workload current in the previous control cycle. The orange curve is the boosted V_{buck} . The SA accelerator enters the startup phase of processing a new convolutional layer at the $3\mu s$ time. The blue curve lags the actual current load during the startup phase. The input voltage boost mechanism increases the V_{buck} from time $3\mu s$ to $5\mu s$ to prevent supply voltage violations. The startup phase ends at the $5\mu s$ time from which the dynamic control policy can set the V_{buck} optimally

TABLE I
NUMBERS AND SIZES OF IFMAPS AND FILTERS IN BENCHMARKED DNN MODELS

Models	Input size	Max # filters per layer	Min # filters per layer	Max filter size	Min filter size
ResNet	224*224*3	2048	64	4608	64
MobileNet	224*224*3	1024	32	9216	27
SSD-ResNet	1200*1200*3	512	16	4608	64
SSD-MobileNet	300*300*3	546	12	9216	27

without power supply noise violation. While boosting the LDO input voltage during the startup phases degrades PDN system efficiency, startup phases are only a small fraction of the overall processing time. Hence, the resulting efficiency degradation is negligible.

V. EXPERIMENTS AND RESULTS

A. Experimental setups

We utilize the proposed modeling framework for systolic array (SA) accelerators to assess the performance of the proposed PDN architecture and control policy. We consider a 65nm 100MHz SA CNN accelerator with an array size of 256 x 256, which is identical to that of the Google's TPU [5]. The power and energy characterization of the SA accelerator is based on the data extracted from [15], [23]. 40% of the total chip power is assumed to be due to leakage [25]. The output stationary (OS) and weight stationary (WS) dataflows are evaluated. The sizes and numbers of IFMAP and filters are listed in Tab. I. These models vary widely in the size of convolutional computation and are selected to well represent both light and heavy CNN workload conditions for comprehensive performance evaluation under a variety of workloads and hardware utilization.

The accelerator is powered by two PDN architectures with or without dynamic control. The two-stage PDN architecture contains clusters of on-chip buck converters and off-chip buck converters while the 3-stage architecture further incorporates the third stage of distributed on-chip LDOs. Without using dynamic control, i.e. static control, intermediate output voltages in the conversion chain and the number of active voltage regulators are fixed while these control variables can be adjusted dynamically with the proposed dynamic control policy. Furthermore, our proposed dynamic control policy and power gating approach are combined to demonstrate even greater energy saving. The power supplies for the on-chip memory are static in all scenarios.

For a fair comparison, all PDN architectures and control settings are configured to meet the same minimum on-chip power supply voltage target of $0.95\,\mathrm{V}$ beyond which power integrity is assumed to be ensured. For the two-stage PDN architecture and the three-stage static PDN architecture, the output voltages of on-chip buck converters are properly set, respectively, to ensure power integrity. We adopt the on-chip LDO input voltage boost mechanism described in Section IV-B to eliminate the voltage violations during the startup phase of the accelerator for three-stage PDNs adjusted by the dynamic control.

B. Dynamic control policy with single power domain

First, we compare the energy efficiency of the two-stage and three-stage PDN architectures with a single power domain with or without the proposed dynamic control. The energy dissipation breakdowns of the PDN for an OS and WS systolic array are reported in Fig. 9 and Fig. 10, respectively for four benchmarking deep learning models. For each particular benchmark DNN, all evaluated schemes supply the same amount of energy to the SA accelerator while they have different energy overheads due to the power distribution and voltage regulation in the PDN. The PDN energy overheads are broken down to those of the LDO regulators, the on-chip buck converters, the package, and the off-chip buck converters, denoted by `ldo`, `on-buck pkg` and `off-buck`, respectively. All energy components are normalized w.r.t the total energy consumed for running the corresponding benchmark in each case.

In Fig. 9 and Fig. 10, the three-stage PDNs with dynamic and static control policies are denoted by `3-dynamic` and `3-static`, respectively. The two-stage PDNs with dynamic and static control policy are denoted by `2-dynamic` and `2-static`, respectively. For both two-stage and three-stage PDNs, the application of the proposed dynamic control policy improves the energy efficiency because of the dynamic adaptation w.r.t the workload. Among all schemes, the three-stage PDN with dynamic control leads to the lowest energy overhead for all benchmarks and across both OS and WS dataflows, and can reduce the energy overhead by up to 10.21% compared with the two-stage static PDNs, indicating the effectiveness of the proposed techniques.

C. Multiple power domains with power gating

Next, we evaluate the additional energy savings brought by using multiple power domains and enabling power gating based on the four DNN models over the OS and WS dataflows. The results for the OS and WS systolic array are shown in Fig. 11 and Fig. 12, respectively. `3-dynamic` denotes single-power-domain without power gating case while `3-pg-2`, `3-pg-4` and `3-pg-8` denote power gating with 2, 4 and 8 power domains, respectively. The energy dissipations are normalized w.r.t the maximum energy delivered to the SA accelerator (processor) (excluding the overhead of the PDN) among all PDN schemes for each benchmark. This maximum value corresponds to the energy dissipated by the accelerator without adopting multiple power domains and power gating. In each case, the energies consumed by the accelerator and

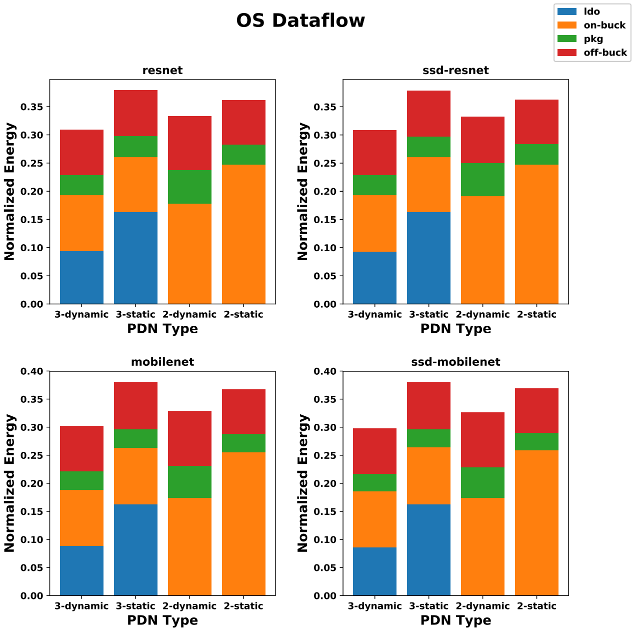

Fig. 9. Normalized energy consumption of different PDNs with a single power domain for an OS systolic array.

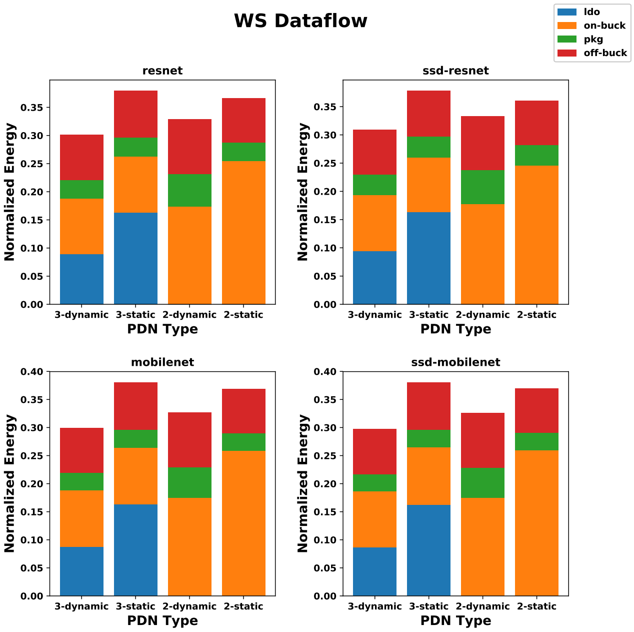

Fig. 10. Normalized energy consumption of different PDNs with a single power domain for a WS systolic array.

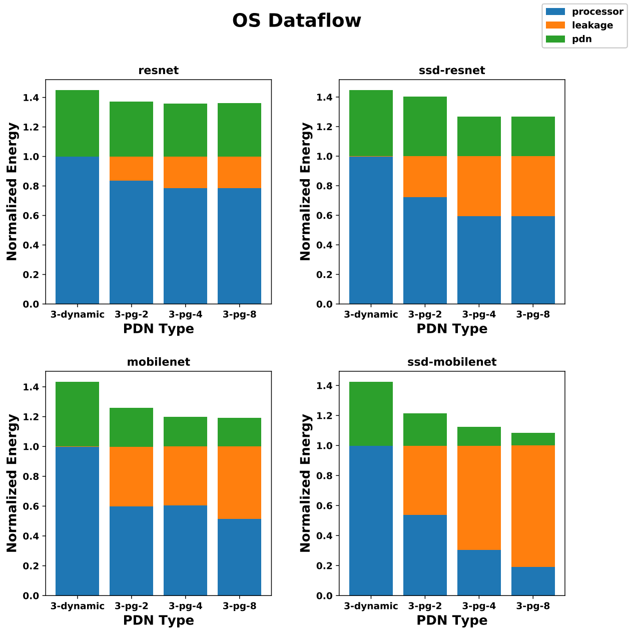

Fig. 11. Energy consumption normalized to maximum processor energy for the OS dataflow. Yellow bars show large **leakage saving** due to power gating.

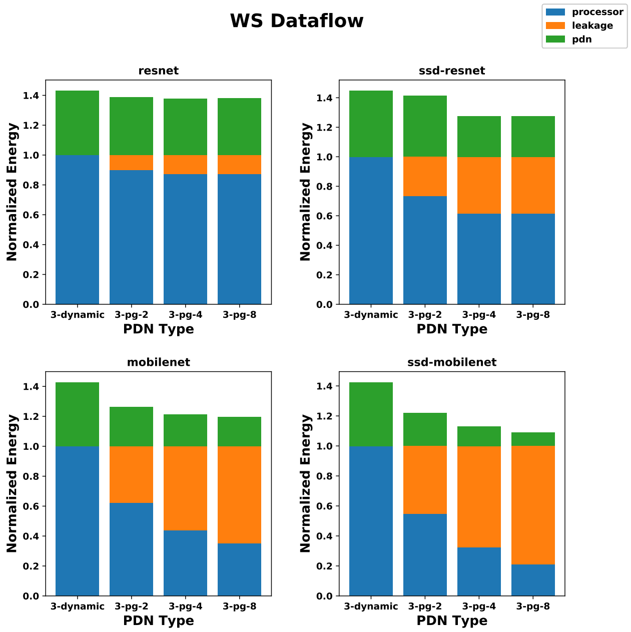

Fig. 12. Energy consumption normalized to maximum processor energy for the WS dataflow. Yellow bars show large **leakage saving** due to power gating.

PDN, and leakage saving due to power gating are reported as `processor`, `pdn`, and `leakage` in the figures.

It is evident from Fig. 11 and Fig. 12 that the proposed power gating technique can significantly improve energy efficiency via leakage saving in addition to the reduction of energy loss in the PDN because layers with low hardware utilization exist in all four models. More specifically, it reduces PDN energy overhead by up to 5.1X and leakage energy by up to 5.3X. These savings increase substantially as more power domains are used.

Among all four benchmarking DNN models, the proposed power gating technique saves the largest amount of energy for the SSD-MobileNet model because it is the most lightweight model and has many convolution layers with less than 64 filters. Compared with those of MobileNet and SSD-MobileNet, the energy saving for ResNet and SSD-ResNet is smaller since the average utilization of both models is higher. The results on these two models also suggest that finer-grained power gating with more than 4 power domains has less additional benefit for heavy-weight models with a larger average number of filters per layer. Single shot detection (SSD) applications benefit more from power gating than image classification applications since they have a higher percentage of layers with fewer filters.

VI. Conclusion

We have presented the first work on HVR based PDN architecture and control for systolic array-based DNN accelerators. PDN architectures comprising heterogeneous on-chip and off-chip voltage regulators and multiple power domains lead to improved energy efficiency of the targeted SA accelerators. By using our simulation and modeling framework, we have further demonstrated that the proposed workload-aware dynamic PDN control policy improves energy efficiency while ensuring power delivery integrity. Combining the proposed HVR PDN architectures, dynamic control, and power gating lead to significant energy saving particularly for deep learning models with a low average number of filters per layer.

VII. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grants No. 2000851 and No. 1741173.

References

[1] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[2] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[4] H. T. Kung, "Why systolic architectures?" *Computer*, vol. 15, pp. 37–46, 1982.

[5]

[6] A. Samajdar, Y. Zhu, P. N. Whatmough, M. Mattina, and T. Krishna, "Scale-sim: Systolic CNN accelerator," *CoRR*, vol. abs/1811.02883, 2018. [Online]. Available: http://arxiv.org/abs/1811.02883

[7] J. Zhang, K. Rangineni, Z. Ghodsi, and S. Garg, "Thundervolt: Enabling aggressive voltage underscaling and timing error resilience for energy efficient deep neural network accelerators," *CoRR*, vol. abs/1802.03806, 2018. [Online]. Available: http://arxiv.org/abs/1802.03806

[8] E. A. Burton *et al.*, "FIVR- Fully Integrated Voltage Regulators on 4th Generation Intel® Core SoCs," in *APEC*. IEEE, 2014, pp. 432–439.

[9] V. Zyuban *et al.*, "IBM POWER8 Circuit Design and Energy Optimization," *IBM Journal of Research and Development*, vol. 59, no. 1, pp. 9–1, 2015.

[10] X. Zhan, J. Chen, E. Sánchez-Sinencio, and P. Li, "Power Management for Multicore Processors via Heterogeneous Voltage Regulation and Machine Learning Enabled Adaptation," *TVLSI*, pp. 1–14, 2019.

[11] D. Pathak, H. Homayoun, and I. Savidis, "Smart grid on chip: Work load-balanced on-chip power delivery," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, no. 9, pp. 2538–2551, 2017.

[12] H. Li, J. Xu, Z. Wang, P. Yang, R. K. V. Maeda, and Z. Tian, "Adaptive power delivery system management for many-core processors with on/off-chip voltage regulators," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2017*, 2017, pp. 1265–1268.

[13] W. Lee, Y. Wang, and M. Pedram, "Optimizing a reconfigurable power distribution network in a multicore platform," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp. 1110–1123, 2015.

[14] A. A. Sinkar, H. R. Ghasemi, M. J. Schulte, U. R. Karpuzcu, and N. S. Kim, "Low-cost per-core voltage domain support for power-constrained high-performance processors," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 4, pp. 747–758, 2014.

[15] Y. Chen, J. Emer, and V. Sze, "Eyeriss: A spatial architecture for energy-efficient dataflow for convolutional neural networks," in *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)*, 2016, pp. 367–379.

[16] Xilinx. Xilinx Machine Learning Suite. (2020). [Online]. Available: https://www.xilinx.com/products/acceleration-solutions/xilinx-machine-learning-suite.html

[17] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti *et al.*, "The gem5 simulator," *ACM SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1–7, 2011.

[18] A. Bakhoda, G. L. Yuan, W. W. L. Fung, H. Wong, and T. M. Aamodt, "Analyzing cuda workloads using a detailed gpu simulator," in *2009 IEEE International Symposium on Performance Analysis of Systems and Software*, 2009, pp. 163–174.

[19] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "Mcpat: an integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*. ACM, 2009, pp. 469–480.

[20] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: a framework for architectural-level power analysis and optimizations," in *Proceedings of 27th International Symposium on Computer Architecture (IEEE Cat. No.RS00201)*, 2000, pp. 83–94.

[21] J. Leng, T. Hetherington, A. ElTantawy, S. Gilani, N. S. Kim, T. M. Aamodt, and V. J. Reddi, "Gpuwattch: Enabling energy optimizations in gpgpus," *SIGARCH Comput. Archit. News*, vol. 41, no. 3, p. 487–498, Jun. 2013. [Online]. Available: https://doi.org/10.1145/2508148.2485964

[22] H. Yang, Y. Zhu, and J. Liu, "End-to-end learning of energy-constrained deep neural networks," *CoRR*, vol. abs/1806.04321, 2018. [Online]. Available: http://arxiv.org/abs/1806.04321

[23] T. Yang, Y. Chen, and V. Sze, "Designing energy-efficient convolutional neural networks using energy-aware pruning," *CoRR*, vol. abs/1611.05128, 2016. [Online]. Available: http://arxiv.org/abs/1611.05128

[24] S. Li, K. Chen, J. H. Ahn, J. B. Brockman, and N. P. Jouppi, "Cacti-p: Architecture-level modeling for sram-based structures with advanced leakage reduction techniques," in *2011 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2011, pp. 694–701.

[25] D. Zoni, A. Canidio, W. Fornaciari, P. Englezakis, C. Nicopoulos, and Y. Sazeides, "Blackout: Enabling fine-grained power gating of buffers in network-on-chip routers," *Journal of Parallel and Distributed Computing*, vol. 104, 01 2017.