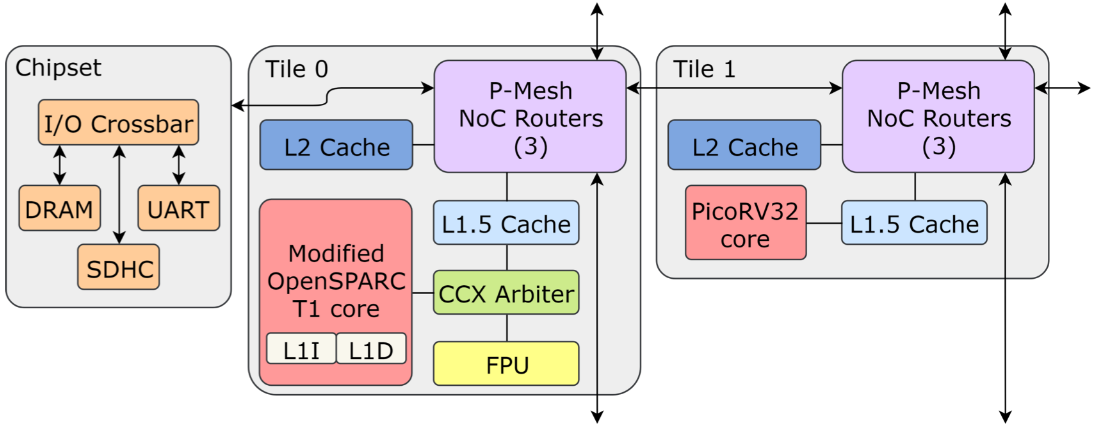
JuxtaPiton: Enabling Heterogeneous-ISA Research with RISC-V and SPARC FPGA Soft-cores

PRINCETON UNIVERSITY

Katie Lim, Jonathan Balkind, David Wentzlaff

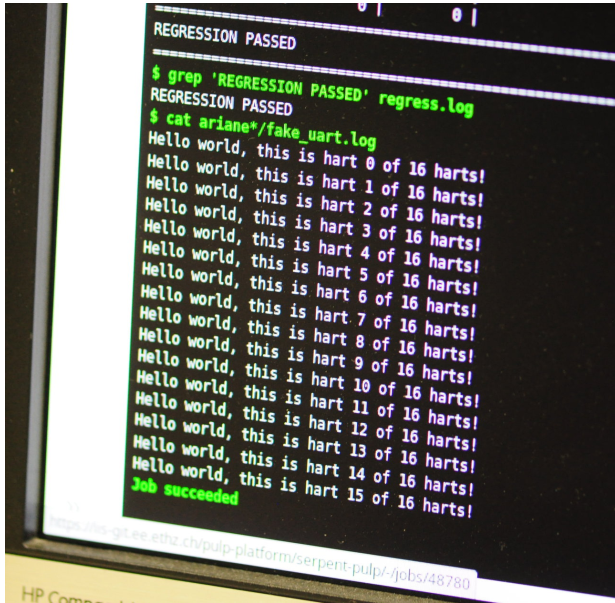
PRINCETON
School of Engineering and Applied Science

- **JuxtaPiton** is the world's first open-source, general-purpose, heterogeneous-ISA processor
- It integrates the PicoRV32 FPGA soft-core into OpenPiton, the manycore research framework



RISC-V OpenPiton

- RISC-V and SPARC cores work together, sharing memory
- SPARC core runs Linux, RISC-V binaries sent to PicoRV32 core
- OpenPiton+Ariane now connects another Linux-capable core to OpenPiton, enabling heterogeneous-ISA OS research
- Seven different cores connected to OpenPiton so far!



```
REGRESSION PASSED
$ grep 'REGRESSION PASSED' regress.log
REGRESSION PASSED
$ cat ariane*/fake_uart.log
Hello world, this is hart 0 of 16 harts!
Hello world, this is hart 1 of 16 harts!
Hello world, this is hart 2 of 16 harts!
Hello world, this is hart 3 of 16 harts!
Hello world, this is hart 4 of 16 harts!
Hello world, this is hart 5 of 16 harts!
Hello world, this is hart 6 of 16 harts!
Hello world, this is hart 7 of 16 harts!
Hello world, this is hart 8 of 16 harts!
Hello world, this is hart 9 of 16 harts!
Hello world, this is hart 10 of 16 harts!
Hello world, this is hart 11 of 16 harts!
Hello world, this is hart 12 of 16 harts!
Hello world, this is hart 13 of 16 harts!
Hello world, this is hart 14 of 16 harts!
Hello world, this is hart 15 of 16 harts!
Job succeeded
```

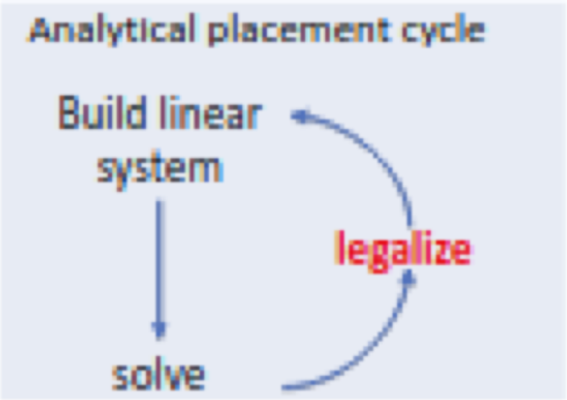
https://iis-git.ee.ethz.ch/pulp-platform/serpent-pulp/-/jobs/48780
HP Comp

https://github.com/PrincetonUniversity/openpiton OpenPiton + Ariane

MODA-PSO: Towards Fast Hard Block Legalization for Analytical FPGA Placement

Yun Zhou, Dries Vercruyce, and Dirk Stroobandt, Ghent University

Motivation

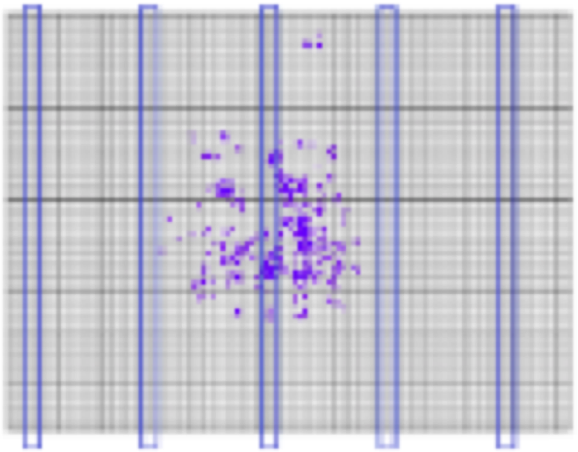


Remove block overlap in optimized placement

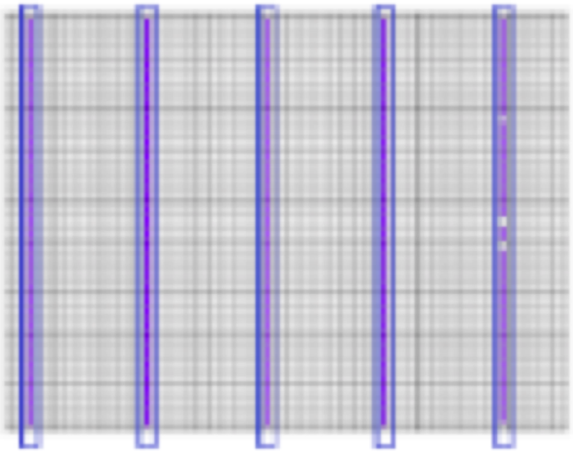
- Fast
- High quality

Currently: simulated annealing-based method

Optimized



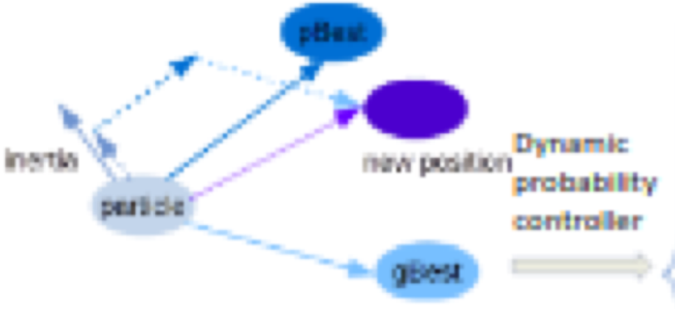
Legalized

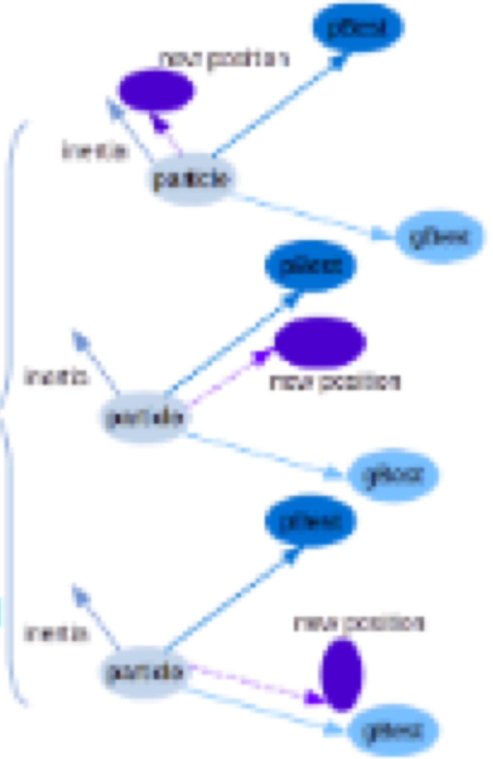


MODA-PSO

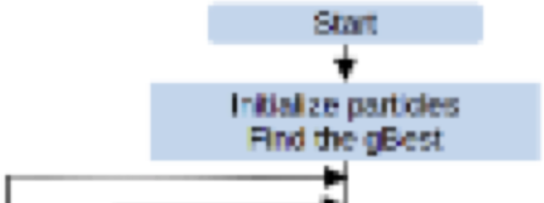
MODA-PSO: **MO**dified **D**iscrete **A**daptive **P**article **S**warm **O**ptimization

- Modified learning rules





- Discrete representation based on the swap operator
- Adaptive learning parameters based on the swarm evolutionary status



A PYNQ-compliant online platform for Zynq-based DNN developers

Chen Chen, Jun Xia, Yang Wen Ming, Li Kang, and Zhilei Chai

E-mail: cchen922143@gmail.com, zlchai@jiangnan.edu.cn

Jiangnan University, Wuxi, Jiangsu Prov. ,China

Abstract: The Zynq SoC from Xilinx is able to support software/hardware co-designing in one single chip, making it possible to take advantage of software flexibility and hardware acceleration at the same time. PYNQ project from Xilinx is trying to take advantage of high performance and low power consumption of Zynq while improve its programmability. In order to improve the ecosystem of PYNQ and help more embedded AI applications use the Zynq-based high-efficiency computational engine, this paper proposes a PYNQ-compliant online platform (OpenHEC-PYNQ) that integrates all necessary factors for the Zynq-based DNN developer.

Problem of DNN development process on Zynq

- Many resources and tools are necessary such as datasets, deep learning framework, fixed-point quantitative tool, the Vivado toolkit and FPGA boards etc.
- Although the heterogeneous SoC coupled with HLS (High Level Synthesis) improves description abstraction of system designing and frees users from non-trivial FPGA driver issues, it is more suitable for system-level programmers.

Our work

- Proposing A PYNQ-compliant online platform that integrates all necessary factors including Zynq devices for the Zynq-based DNN developer

✓ one-stop service for HDL/HLS designers

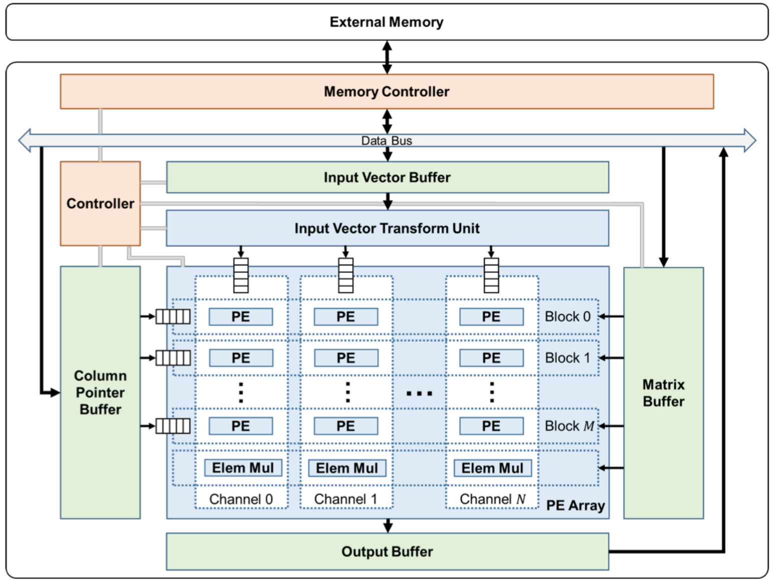
✓ access all resources and finish jobs through Docker-based FPGA cloud platform

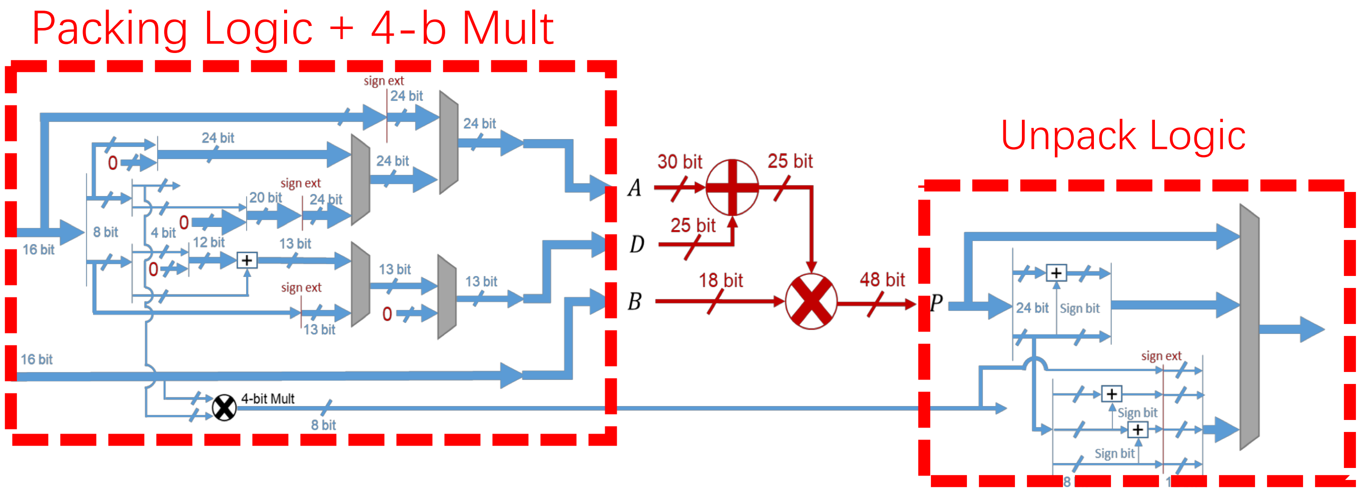
- Taking YOLOv2, one popular Object Detection algorithm, as an example to demonstrate the convenience of the online platform and contribute an open source project for PYNQ.

A Fine-Grained Sparse Accelerator for Multi-Precision DNN

Shulin Zeng[1,3], Yunjun Lin[2], Shuang Liang[1], Junlong Kang[3], Dongliang Xie[3],
Yi Shan[3], Song Han[2], Yu Wang[1], Huazhong Yang[1]
[1]Dept. of E.E., Tsinghua University, Beijing, China
[2]Massachusetts Institute of Technology, MA USA, [3]Xilinx, Beijing China
Email: zengsl18@mails.tsinghua.edu.cn Email: yu-wang@mail.tsinghua.edu.cn

- A Fine-Grained Sparse Accelerator based on EIE[1]
- Introducing Input Vector Transform Unit and PE Array to further support CNN
- DSP-based multi-precision multiplier design





[1] Han, Song, et al. "EIE: efficient inference engine on compressed deep neural network." 2016 ISCA

Building FPGA State Machines from Sequential Code

Carl-Johannes Johnsen <cjjohnsen@nbi.ku.dk>
Kenneth Skovhede <skovhede@nbi.ku.dk>

We introduce a barrier-like construct to the synchronous message exchange (SME) model. This is used for transforming a SME process into a state machine by partitioning the sequential body into states.

This allows for easy sequencing of processes, while retaining the sequential structure of the code. Preliminary results shows increased clock rate and reduced resource consumption.

Design and Implementation of a Deterministic FPGA Router on a CPU+FPGA Acceleration Platform

Dario Korolija and Mirjana Stojilović, École Polytechnique Fédérale de Lausanne (EPFL)

- Instead of trying to accelerate FPGA routing purely in software, we venture into sharing the workload between an FPGA and a CPU

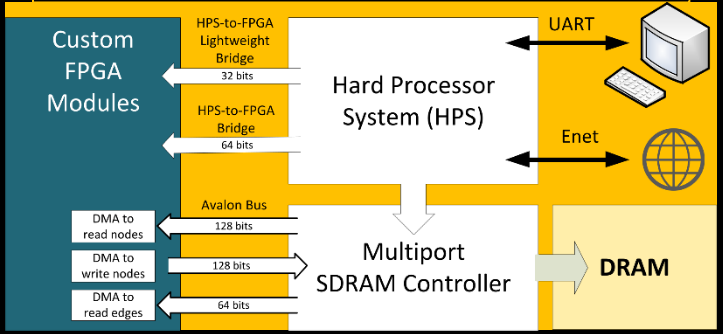


Figure: FPGA+CPU system diagram

Accelerating FPGA routing is hard

Our strategies:

1. Less irregular memory accesses
   - Group wires into sets
   - Reduce redundant expansions
2. Parallel and pipelined computation
3. Efficient HW priority queue

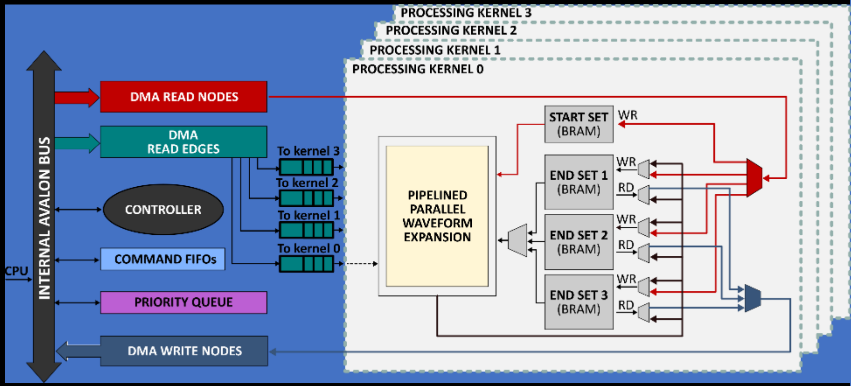


Figure: FPGA waveform expansion accelerator

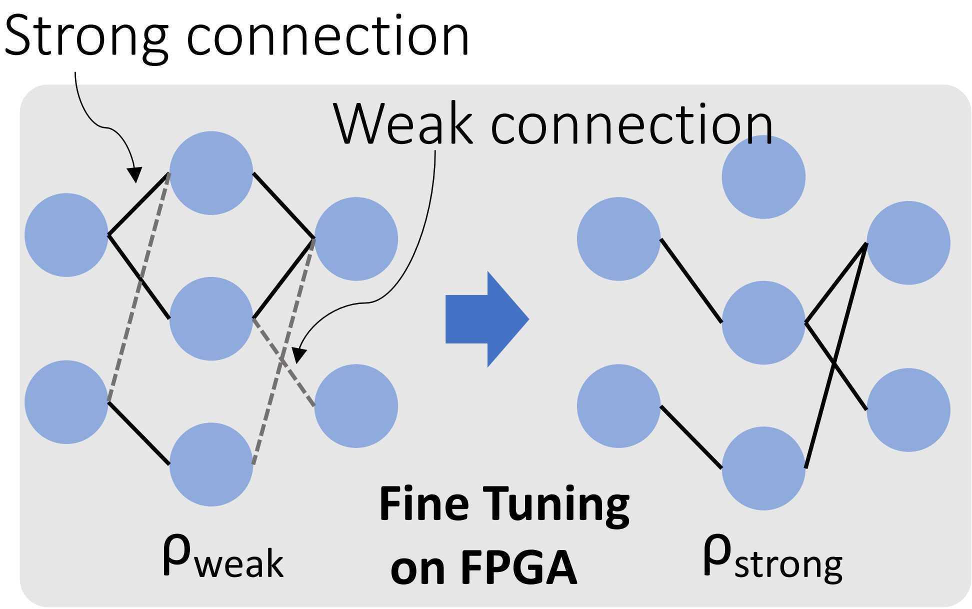
Tested on L-1 and L-4 FPGA architectures and VTR circuits.
Compared to VTR 8.0 on Intel Core i5.

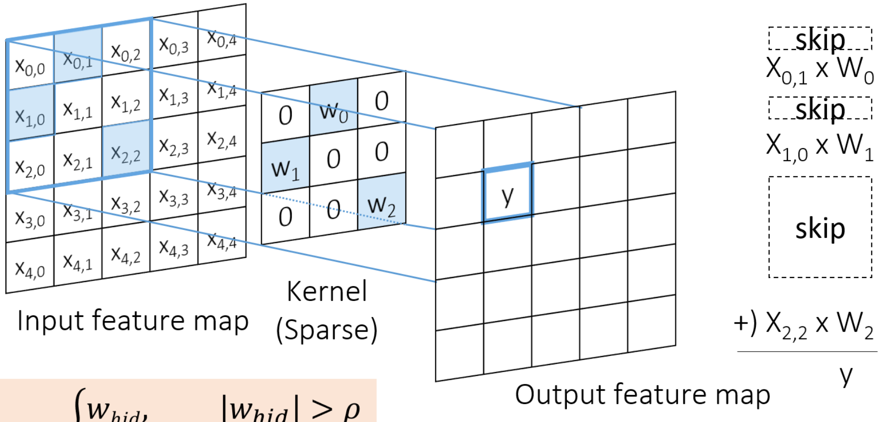
- Our FPGA+CPU router outperforms CPU-only router
- Yet, our mid-end acceleration platform (Intel DE1-SoC) does not outperform a full-blown desktop CPU hardware
  - A performance model predicts good results if a more powerful platform, e.g., HARP FPGA+CPU, would be used (future work…)

An FPGA-based Fine Tuning Accelerator for a Sparse CNN

Hiroki Nakahara, Akira Jinguji, Masayuki Shimoda, Shimpei Sato (Tokyo Institute of Technology, Japan)

- **Use pre-trained model (sparse weight) by ImageNet**
  - **80-85% of weights can be ignored during fine-tuning**
- **Sparseness convolution accelerator on Xilinx VCU1025**
- **4.0 times faster than NVIDIA GTX1080 Ti**




$x_{0,0}$ $x_{0,1}$ $x_{0,2}$ $x_{0,3}$ $x_{0,4}$
$x_{1,0}$ $x_{1,1}$ $x_{1,2}$ $x_{1,3}$ $x_{1,4}$
$x_{2,0}$ $x_{2,1}$ $x_{2,2}$ $x_{2,3}$ $x_{2,4}$
$x_{3,0}$ $x_{3,1}$ $x_{3,2}$ $x_{3,3}$ $x_{3,4}$
$x_{4,0}$ $x_{4,1}$ $x_{4,2}$ $x_{4,3}$ $x_{4,4}$
Input feature map
0 w_0 0
w_1 0 0
0 0 w_2
Kernel (Sparse)
y
Output feature map
skip
$X_{0,1}$ x W_0
skip
$X_{1,0}$ x W_1
skip
+) $X_{2,2}$ x W_2
y

$$w = \begin{cases} w_{hid}, & |w_{hid}| > \rho \\ 0, & Otherwise \end{cases}$$

ρ: Threshold

Dataflow Systolic Array Implementations of Matrix Decomposition Using High-Level Synthesis

Jie Liu, Jason Cong

- Matrix decomposition is a fundamental topic in numerical algebra. Many specific systolic array structures of matrix decomposition algorithms have been proposed to maintain high performance as the problem size scales up.

- In this work, the authors broadly explore different mappings of Cholesky, LU and QR decomposition algorithms to systolic arrays. Evaluation of the optimized designs implemented using Xilinx HLS tools shows up to 50.13x and 4.58x better throughput compared with the corresponding methods in Xilinx HLS linear algebra library and the LAPACK library on CPUs.

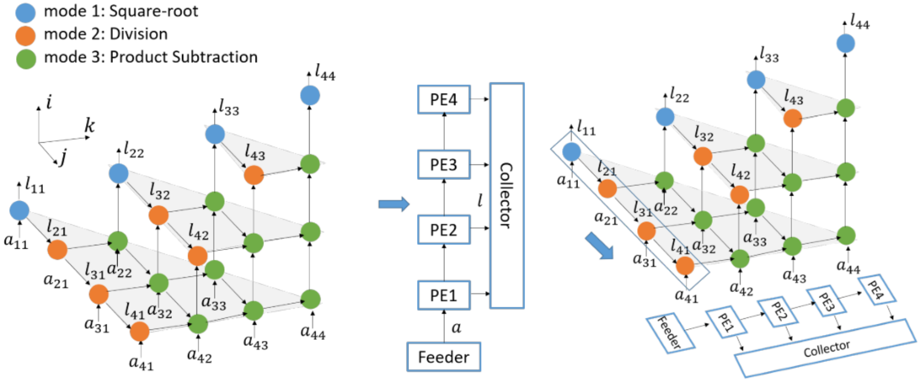


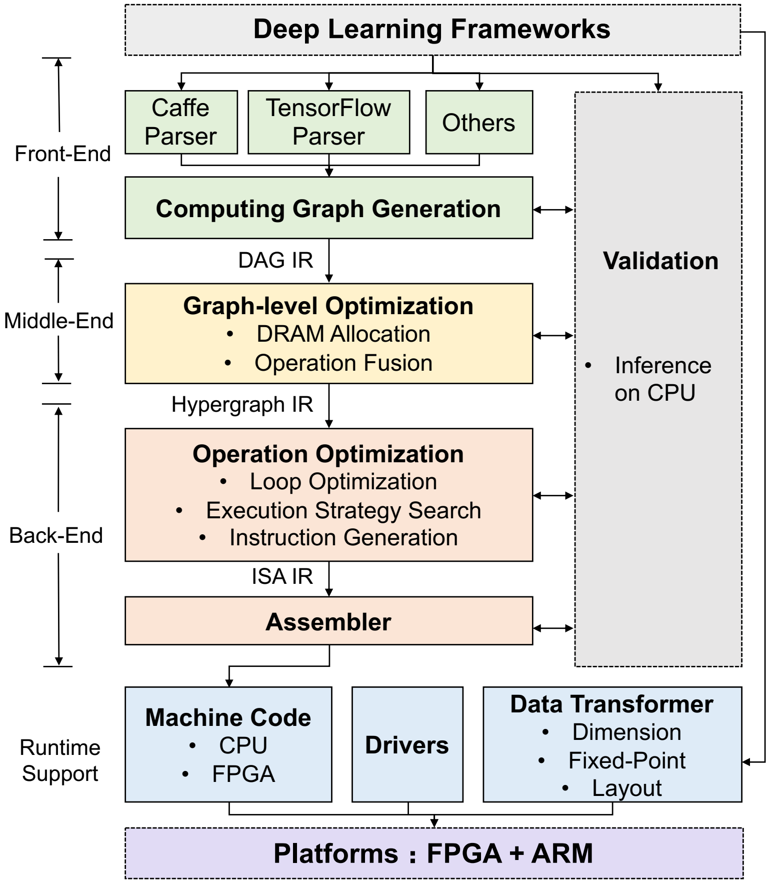
FIGURE 1. Projection of Cholesky decomposition from dependency graph to 1D systolic array

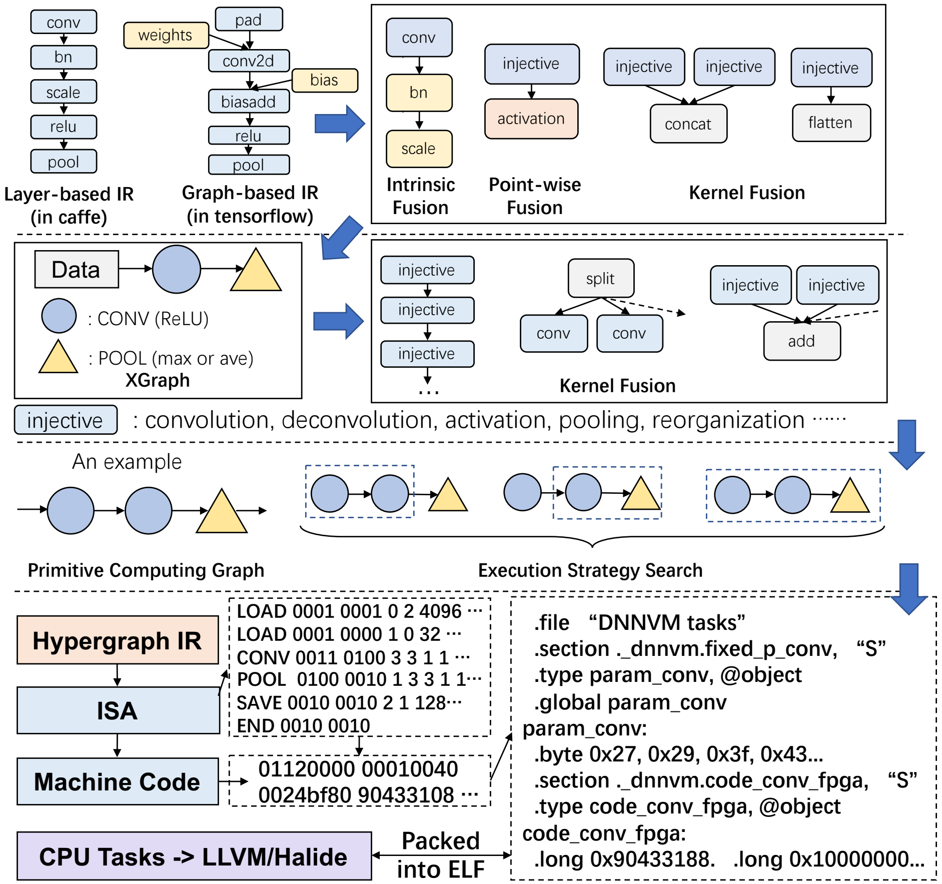
CHOLESKY		SA_1D	SA_2D	HLS LIB	LAPACK-1 thread
Size 8	Latency(us)	1.728	1.572	11.516	1.260
	Throughput (MMatrices/s)	1.524	2.016	0.087	0.794
Size 16	Latency(us)	4.220	3.460	29.740	3.060
	Throughput (MMatrices/s)	0.415	1.497	0.034	0.327
Size 32	Latency(us)	11.172	17.708	128.588	4.407
	Throughput (MMatrices/s)	0.116	0.401	0.008	0.227

TABLE 1. Performance Comparison of Cholesky Decomposition

DNNVM : End-to-End Compiler Leveraging Operation Fusion on FPGA-based CNN Accelerators

Yu Xing(1), Shuang Liang(2), Lingzhi Sui(1), Zhen Zhang(1), Jiantao Qiu(2), Xijie Jia(1), Xin Liu(1), Yushun Wang(1),
Yi Shan(1), Yu Wang(2)
Xilinx(1) Tsinghua University(2)





```
LOAD 0001 0001 0 2 4096 ···
LOAD 0001 0000 1 0 32 ···
CONV 0011 0100 3 3 1 1 ···
POOL  0100 0010 1 3 3 1 1···
SAVE 0010 0010 2 1 128···
END 0010 0010
```

```
  01120000 00010040
  0024bf80 90433108 ···
```

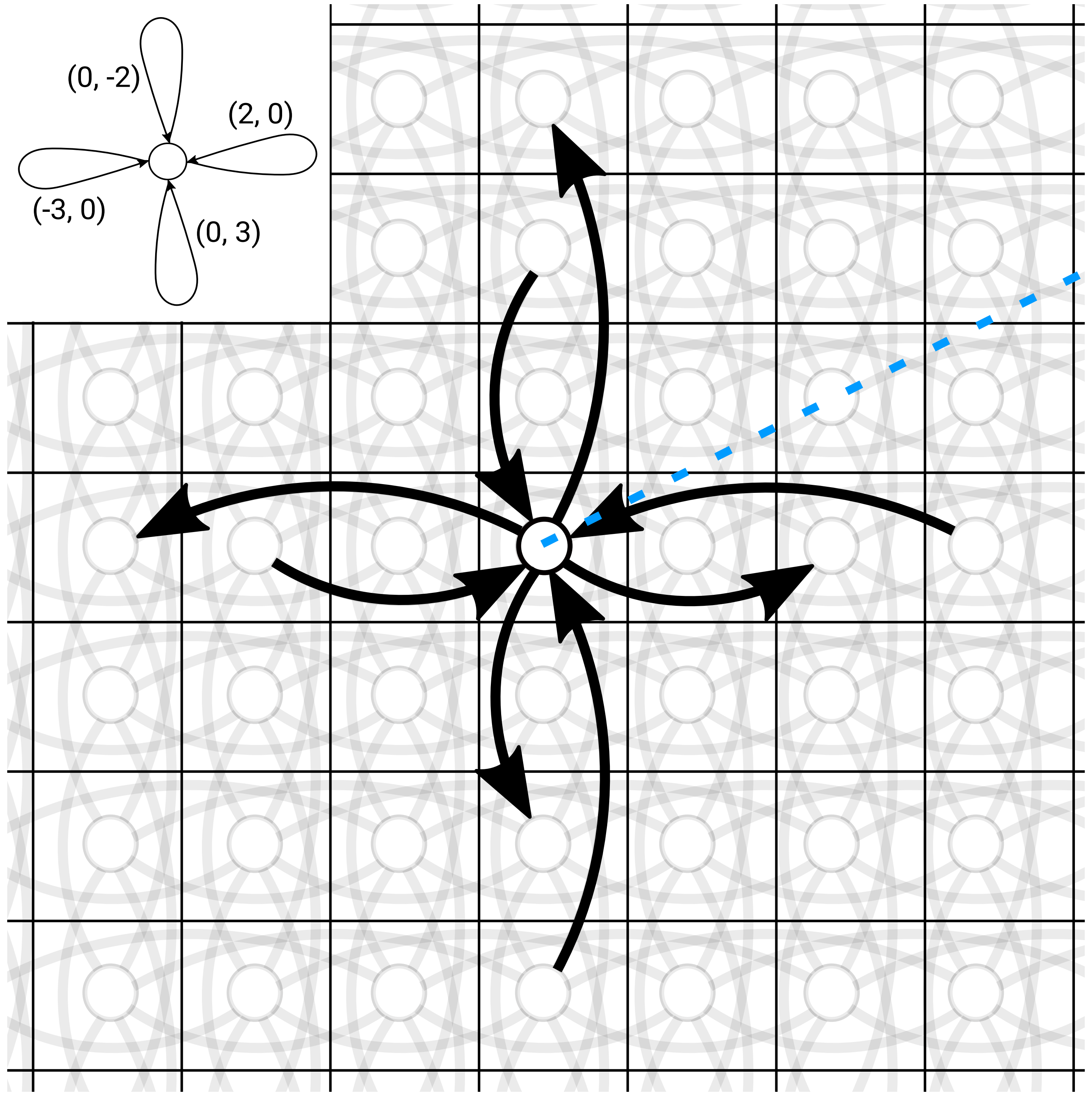
```
  .file   "DNNVM tasks"
  .section ._dnnvm.fixed_p_conv,   "S"
  .type param_conv, @object
  .global param_conv
param_conv:
  .byte 0x27, 0x29, 0x3f, 0x43...
  .section ._dnnvm.code_conv_fpga,   "S"
  .type code_conv_fpga, @object
code_conv_fpga:
  .long 0x90433188.   .long 0x10000000...
```

- A full-stack compiler infrastructure to generate optimized instructions for our FPGA-based CNN accelerator.
- Up to 1.26x throughout improvement by leveraging operation fusion on our benchmarks.
- State-of-the-art performance for VGG and ResNet50 on ZU9.

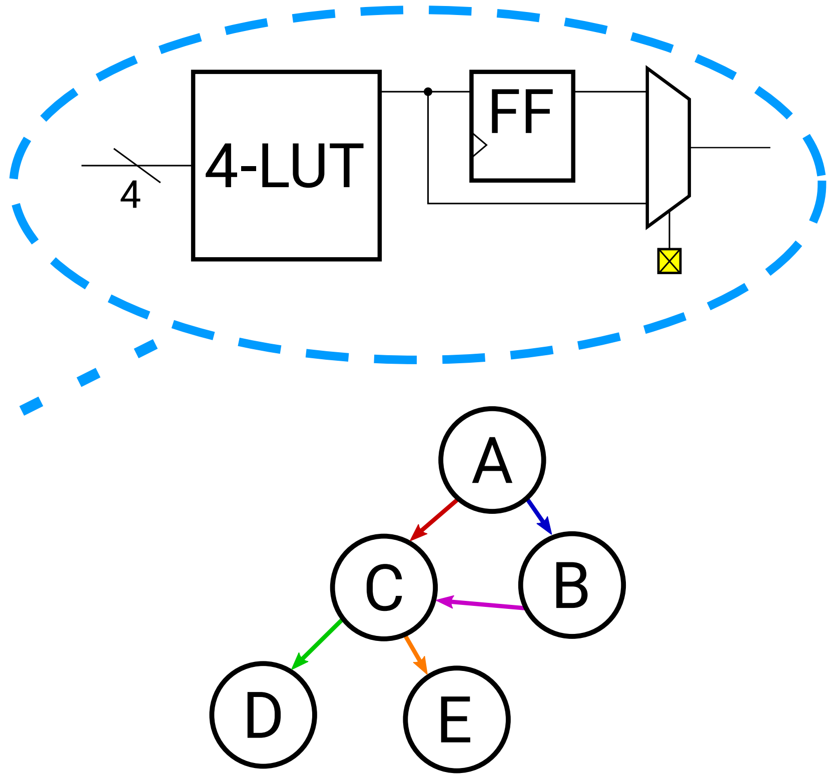
On Feasibility of FPGAs Without Dedicated Programmable Interconnect Structure

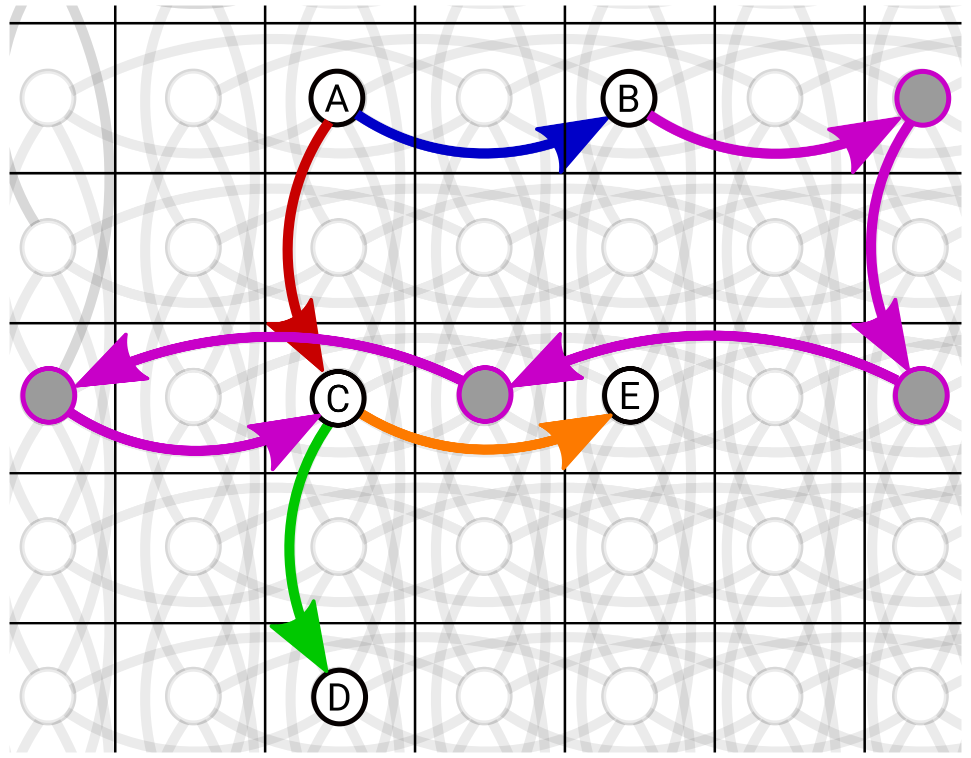
Anastasiia Kucherenko, Stefan Nikolić, and Paolo Ienne

Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland



Witness architecture





Example embedding

How much routing flexibility do we really need?

Can we do without a dedicated routing structure altogether?

Yes, we can!

(though at a heavy price for now)

We first prove that the architecture on the left can implement any circuit, by giving a placement and routing algorithm.

We then extend the algorithm to a broad class of other architectures.

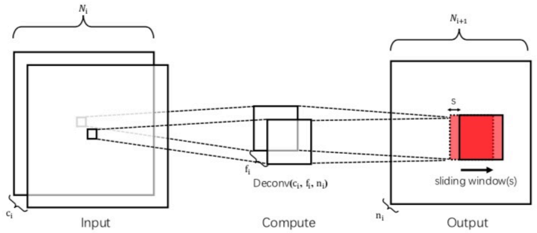
Fast Confidence Detection: One Hot Way to Detect Adversarial Attacks via Sensor Pattern Noise Fingerprinting

Yazhu Lan[1], Kent Nixon[1], Qingli Guo[4], Guohe Zhang[2], Yuanchao Xu[3], Hai Li[1], Yiran Chen[1]
Duke University(1), Xian Jiaotong University(2), Capital Normal University(3), University of Chinese Academy of Sciences(4)
email: yazhu.lan@duke.edu

- We propose an innovative method (**FCDM**) for fast confidence detection of adversarial attacks based on integrity of sensor pattern noise embedded in input examples.
- Our proposed method is capable of providing a confidence distribution model of most of popular adversarial attacks.
- Our presented method can provide early attack warning with even the attack types based on different properties of the confidence distribution models.
- We realize our proposed method on an FPGA platform and achieve a high efficiency of 29.740 IPS/W with a power consumption of only 0.7626W.

FTConv: FPGA Acceleration for Transposed Convolution Layers in Deep Neural Networks

- Accelerator for Transposed Convolution
- Why FTConv: High computation complexity & Few studies focus on transposed convolution



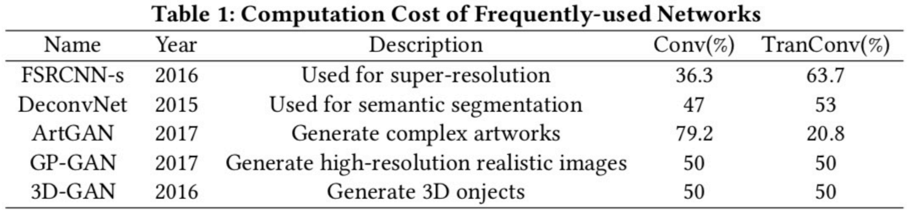
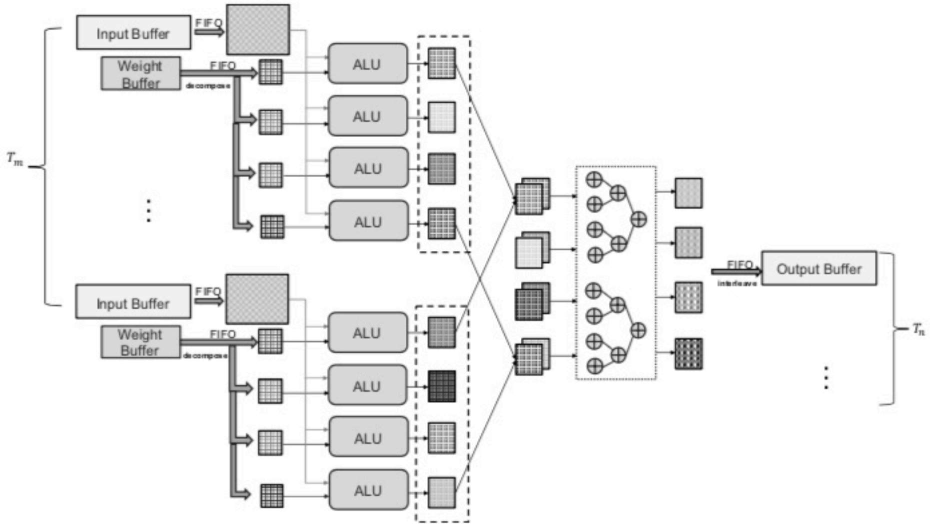
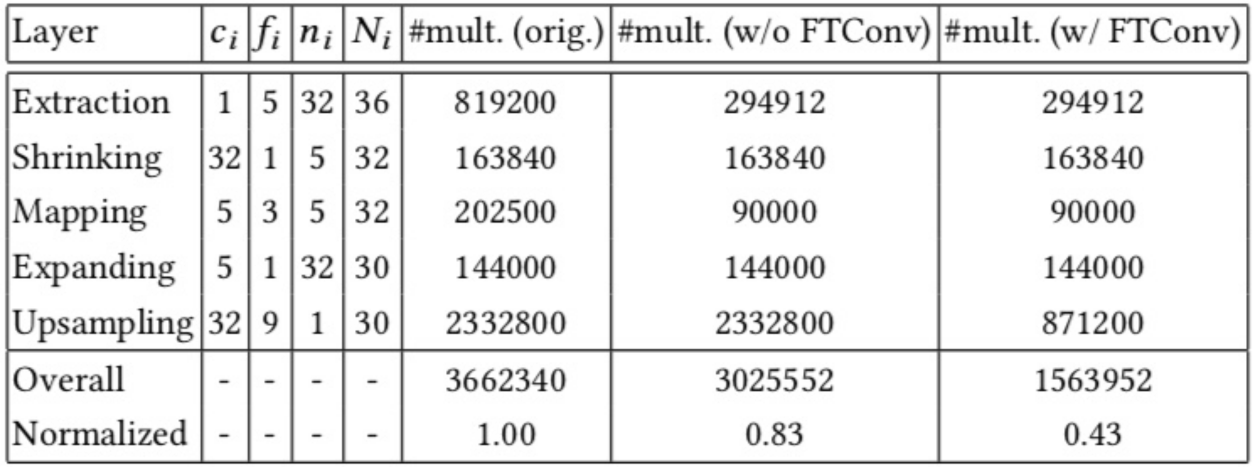


Table 1: Computation Cost of Frequently-used Networks

Name	Year	Description	Conv(%)	TranConv(%)
FSRCNN-s	2016	Used for super-resolution	36.3	63.7
DeconvNet	2015	Used for semantic segmentation	47	53
ArtGAN	2017	Generate complex artworks	79.2	20.8
GP-GAN	2017	Generate high-resolution realistic images	50	50
3D-GAN	2016	Generate 3D onjects	50	50

- Architecture Overview and Theoretical Results:





Layer	c_i	f_i	n_i	N_i	#mult. (orig.)	#mult. (w/o FTConv)	#mult. (w/ FTConv)
Extraction	1	5	32	36	819200	294912	294912
Shrinking	32	1	5	32	163840	163840	163840
Mapping	5	3	5	32	202500	90000	90000
Expanding	5	1	32	30	144000	144000	144000
Upsampling	32	9	1	30	2332800	2332800	871200
Overall	-	-	-	-	3662340	3025552	1563952
Normalized	-	-	-	-	1.00	0.83	0.43

Zhucheng Tang, Guojie Luo, Ming Jiang; Peking University　**Contact: zhucheng.tang@pku.edu.cn**

PVT-Aware Sensing and Voltage Scaling for Energy Efficient FPGAs

Konstantinos Maragos[1], George Lentaris[1], Dimitrios Soudris[1], Vasilis F. Pavlidis[2]
National Technical University of Athens, School of ECE, Greece[1]
The University of Manchester, School of Computer Science, UK[2]

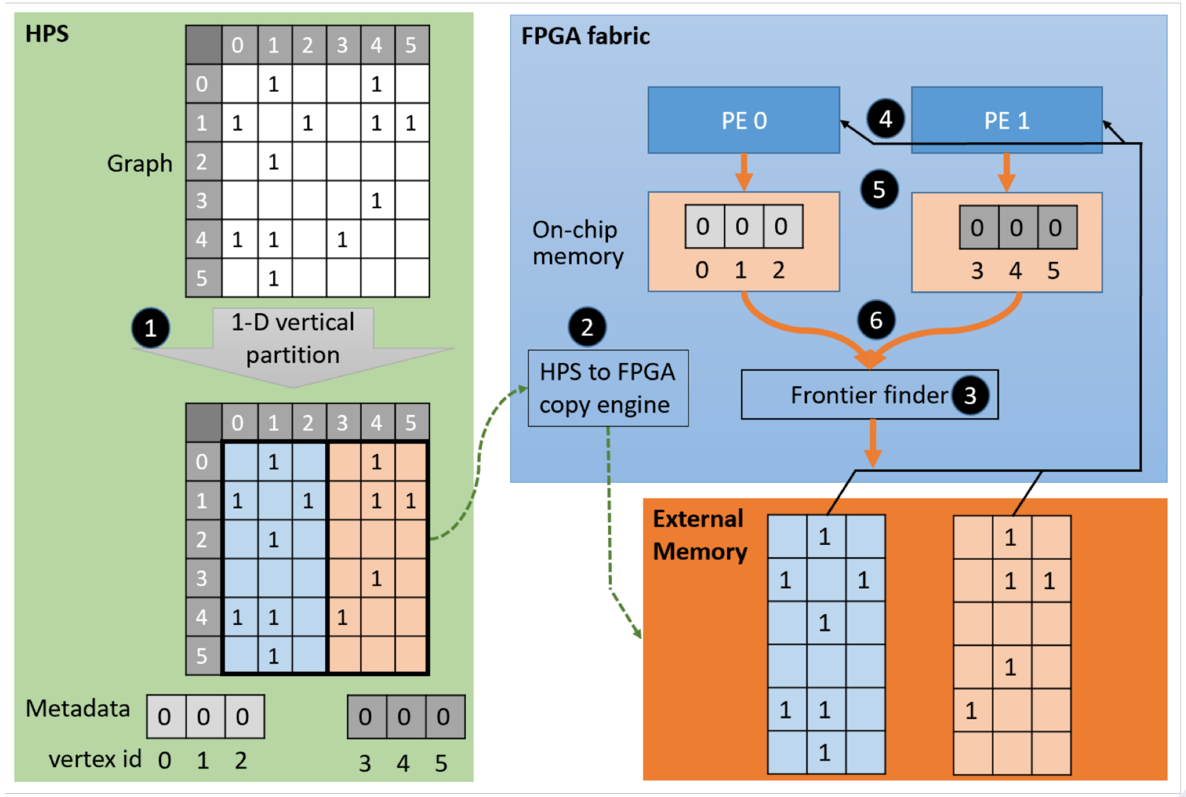
- Propose a method to narrow the operation guard-bands and improve energy efficiency of present-day FPGAs
- Deployment of a sensor network across the fabric of the FPGA to evaluate process, voltage and temperature (PVT) variations
- Reduction of power via voltage scaling in tandem with online monitoring of PVT variations
- Fast integration of the proposed method as ready-to-use IP
- Substantial powers savings in realistic benchmarks , e.g., 27%, while maintaining their nominal timing performance and functional integrity

HARDWARE-SOFTWARE CODESIGNED BFS FOR FPGAS

Zach Sherer, Eric Finnerty, Yan Luo, Hang Liu

University of Massachusetts Lowell ACANETS Lab

- Breadth-first search (BFS) accelerator designed for SoC FPGAs
- Uses vertical graph partitioning techniques to separate the graph into independently traversable sections
  - Uses multiple independent processing elements to parallelize the traversal
  - Moves away from a "fully-connected" implementation
- Applications in low-power data center as well as autonomous vehicles
- Achieves 2.3x speedup over state-of-the-art "fully-connected" designs



Graph

	0	1	2	3	4	5
0		1			1	
1	1		1		1	1
2		1				
3					1	
4	1	1		1		
5		1				

Metadata	0	0	0	0	0	0
vertex id	0	1	2	3	4	5

- PAI-FCNN: FPGA Based CNN Inference System Lansong Diao, Zhao Jiang, Hao Liang, Chang'an Ye, Kai Chen, Li Ding, Shunli Dou, Meng Sun, Lixue Xia, Jiansong Zhang, Wei Lin, Alibaba Group Contact: muduan.zjs@alibaba-inc.com
- SwitchAgg: A Further Step Towards In-Network Computation Fan Yang, ZhanWang, Xiaoxiao Ma, Guojun Yuan, Xuejun An, Institute of Computing Technology, Chinese Academy of Sciences Contact: yangfan@ncic.ac.cn
- Embracing Systolic: Super Systolization of Large-Scale Circulant Matrix-vector Multiplication on FPGA with Subquadratic Space Complexity Jiafeng Xie, Villanova University Chiou-Yng Lee, Lunghwa University of Science & Technology Contact: jiafeng.xie@villanova.edu
- Speedy: An Accelerator for Sparse Convolutional Neural Networks on FPGAs Liqiang Lu1, Yun Liang1 Ruirui Huang2, Wei Lin2, Xiaoyuan Cui2, Jiansong Zhang2 1 Peking University 2 Alibaba group Contact: liqianglu@pku.edu.cn
- A Hybrid Data-Consistent Framework for Link-Aware AccessManagement in Emerging CPU-FPGA Platforms Liang Feng, Jieru Zhao, Tingyuan Liang, HKUST Sharad Sinha, IIT Goa Wei Zhang, HKUST Contact: lfengad@connect.ust.hk
- Compressed CNN Training with FPGA-based Accelerator Kaiyuan Guo, Shuang Liang, Jincheng Yu, Xuefei Ning, Wenshuo Li, Yu Wang, Huazhong Yang, Tsinghua University Contact: gky15@mails.tsinghua.edu.cn
- Optimizing Order-Associative Kernel Computation with Joint Memory Banking and Data Reuse Juan Escobedo, University of Central Florida Mingjie Lin, University of Central Florida Contact: johne1312@knights.ucf.edu