Energy Efficient Binarized Neural Networks with Adaptive Voltage and Frequency scaling

Jose Nunez-Yanez
University of Bristol
Bristol, BS8 1UB, UK
j.l.nunez-yanez@bristol.ac.uk

Abstract
This research investigates the effects of applying adaptive voltage and frequency scaling (AVFS) to a high-performance and reconfigurable binarized neural network. AVFS is made possible by instrumenting the design netlist with timing detectors that enable the exploitation of the operating margins of the FPGA device in a reliable way. The instrumented binarized neural network is targeted to a state-of-the-art FPGA device with nominal operating voltages of 0.85 Volts and fabricated with a 16nm feature size. The results show that the new network can obtain new performance and energy points that are up to 86% better than nominal at an identical level of classification accuracy. The results also indicate that the built-in neural network robustness allows operation beyond the first point of error while maintaining the classification accuracy largely unaffected within a +1/-1% accuracy deviation.

Author Keywords
FPGA, deep-learning, DVFS, neural network

Introduction
Fully binarized neural networks are a type of convolutional neural networks that reduce the precision of weights and activations from floating point to binary values. They can achieve a high inference accuracy in deep learning applications and are highly suited for FPGA implementation since

floating point matrix multiplications are reduced to binary operations involving XOR gates and bit counts. In this research we study the extension and application of an adaptive voltage scaling framework [6] called Elongate to the FINN binarised neural network [8]. We consider a target platform built around a Xilinx Zynq Ultrascale devices. The results show that Elongate can determine extended operating points of voltage and frequency, enabling higher performance, lower power or trade-offs between performance and power so the amount of computation and energy usage adapts to the workload requirements at run-time. This is particularly relevant to, for example, image classification applications based on machine learning in which the amount of work depends on the amount of frame activity and previously classified objects do not need to be reclassified. The binarized neural network is selected as the case study since its simple control flow and simple logic operations (e.g. such as XOR) do not require DSP blocks that could be problematic to instrument if the critical path end-points are buried inside the DSPs. The built-in error tolerance of the network also enables to operate with an error constraint higher than zero as seen in the experimental analysis. Notice that error rate in this paper refers to allowing or not allowing errors in the instrumented flip-flops and not to errors in the accuracy of the neural network.

Related Work

The hardware acceleration of deep neural networks has been receiving significant attention in recent years with many efforts targeting many-core processors, custom architectures, GPUs and FPGAs accelerators [7]. GPUs offer high peak performance for classical DNN operations such as dense matrix multiplication but recent trends in DNN research that favor sparse networks and compact data representation could benefit from the FPGA strengths. DNNs based on floating-point operands provide overall high classification accuracies but require large compute/memory resources [4]. An example of more compact data representation is introduced in SqueezeNet [3] that uses reduced precision with fixed-point arithmetic and fewer parameters than the full network and it is suitable for deployment on hardware with limited memory. Further reductions in precision are performed in [1] that presents a state of the art implementation of the Alexnet CNN for a vision task using OpenCL. The system uses half-precision floating-point arithmetic (FP16) and is competitive in terms of performance and power with state-of-the-art GPUs achieving 1020 images/s and 23 images/s/watt (similar to a TitanX GPU) with a peak throughput of 1.3TFLOPS. It uses an Arria 10 1150 device at 300 MHz with a power consumption of 45 Watts. The accuracy is top-1 56% and top-5 79% on the Imagenet data set. DSP utilization reaches 97% in the device and the paper identifies external memory bandwidth as one of the main performance limiting factors. Extreme compact data representation has been introduced in Binarized Neural Networks (BNNs) [2] with single-bit neuron values and weights. A complete and efficient framework to implement BNNs on FFGA is FINN [8]. FINN is based on the BNN method developed in [2] providing high performance and low memory cost using XNOR-popcount-threshold data-paths with all the parameters stored in on-chip memory. FINN has a streaming multi-layer pipeline architecture where every layer is composed of a compute engine surrounded by input/output buffers. A FINN engine implements the matrix-vector products of fully-connected layers or the matrix-matrix products of convolution operations. Each engine computes binarized products and then compares against a threshold for binarized activation. It consists of **P** processing elements (PEs), each having **S** SIMD lanes. The first layer of the network receives non-binarised image inputs and hence it requires regular operations while the last layer outputs non-binarised classification results and

does not require thresholding. Although Elongate can be applied to large architectures in this research we use the FINN framework to create the proposed energy proportional and scalable architecture based on voltage and frequency adaptation.

Elongate Framework

The Elongate closed-loop system constantly monitors timing signals originating in the user logic (e.g. BNN) and adapts the clock and voltage as specified by the user. The user has access to a configuration register that defines the allowed activation rate. The activation rate is the number of activations allowed in the detector flip-flops before the clock frequency is reconfigured. An activation rate set to one indicates that a single detector flip-flop activation triggers a clock reconfiguration and corresponds to maximum sensitivity. The design of the detectors ensures that the first activations are detected before errors occur so an activation rate set to one corresponds to an error rate of zero. In general, the requested error rate is zero and this is the default configuration in the BNN system. This means that in this configuration zero errors are introduced in the logic and we call this safe mode of operation NPF (Near Point of Failure). However, the BNN application exhibits strong error tolerance features and in some cases it could be useful to allow the system to perform at a detector error rate higher than zero to obtain even lower power and higher performance if overall classification accuracy remains largely unaffected. An error rate higher than zero is possible if we set the activation rate to a number higher than one. The higher the activation rate the higher the probability that errors will affect the data path logic. We call this mode APF (After Point of Failure) and we explore this possibility in section Accuracy Analysis . The Elongate framework integrates with the Xilinx SDx tools and enables the user to work with C/C++ (OpenCL support will be added in future work) as

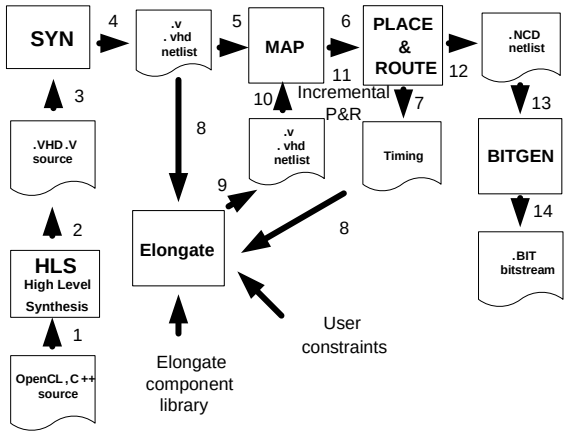

Figure 1: Elongate steps

the design language as done in the BNN application. Notice that using C/C++ and SDx as part of our framework means that it is not possible to track how hardware functions are mapped to the RTL generated by SDx specially for complex designs. We treat this RTL as a blackbox and protect paths independently if they are part of the control or data path logic. The obvious problem is that if the error rate is relaxed to higher than zero then an error in the control logic could be catastrophic resulting in a crash. This indicates that this error tolerance approach is not viable for many designs (i.e. individual design characterization is necessary) but in our BNN case study the dataflow nature of the design and the small control plane enable the system to work reliably. Figure 1 shows in more detail how Elongate integrates with the processing steps taken by the FPGA tools during synthesis and implementation. Elongate processing is perform by perl and tcl scripts. The Elongate component library shown in Figure 1 contains RTL for the detectors and mon-

itoring logic that are inserted in the original design netlist. The numbers in Figure 1 indicate the logical order of the Elongate steps. The incremental P&R in step 10 is used to reuse most of the implementation information and reduce the risk of possible variations in the critical paths. The number of detectors inserted into the netlist is user configurable and normally set to cover as many paths as possible while maintaining overheads within 5%.

Binarised Neural Network Application

Figure 3 shows the topology details of the convolutional FINN BNN as used in this work which has a model size of 187 Kbytes. Figure 2 shows the BNN hardware and the Elongate IP architecture in the Zynq Ultrascale ZC9 device used in the ZCU102 board. This board contains a PMBUS (Power Manager BUS) power control and monitoring system that enables the reading of power and current values using the ARM CPUs. It also enables the ARM CPUs to write new voltage values to the power regulators. Both of these features have been used extensively to measure power and to change the voltage level at which the device operates. The large number of resources available in this device makes it possible to scale the logic from the original design in [8] that targets a Zynq 7045 device. The new BNN processor contains 4 independent compute units with a total of 832 PEs and 1488 SIMDs in the zcu102 Zynq Ultrascale board and a single compute unit, 91 PEs and 176 SIMDs in the zc702 Zynq board. The hardware utilization is shown in Figure 4

In the Zynq Ultrascale configuration nominal classification performance reaches 89500 FPS with a clock frequency of 200 Mhz. The next sections will discuss how this performance can be extended and made energy efficient with the Elongate framework. Energy efficiency is measured by monitoring the PL power in both devices at 37800 FPS/Watt in the zc9 device. Figure 2 shows that a single compute unit (BNN_ZU0) has been instrumented with the Elongate detectors. This means that this compute unit sets the operating point for itself and for the other 3 compute units. The timing analysis data obtained during Elongate integration is used to choose the compute unit with the longest critical paths for instrumentation. Figure 2 shows two Master interfaces (HPM0 and HPM1) going from the PS side to the PL side. The reason is that since HPM0 uses the ELO_CLK it is effectively disabled when a clock gated state with ELO_CLK is initiated. HPM1 does not use ELO_CLK and it is not disabled so when the BNN logic is clock gated and CLK_ELO stops the processor can still communicate with the ELO control logic using the second master HPM1. A single ELO_CLK clock is available for all compute units. In the current configuration it is possible to launch execution using one to four compute units and the SDx software automatically divides the total frame number among the active compute units.

Power Scaling

In this section we focus on the power of the FPGA fabric (i.e. PL) that is supplied by the VCCINT power rail. Other power rails include VCCAUX that powers the clock managers and the IOs among other blocks and the VCCBRAM use with the BlockRAMs. The power drawn from these additional power rails is considerably lower than VCCINT. In addition, the processing side of the device where the ARM processor resides is not included in the calculations. There is a large body of research of power and energy optimization on CPUs via sleep and wake-up states, etc which are outside the scope of this paper. Figure 5 show the measured power in function of the clock frequency and the voltage the BNN operates for the Zynq Ultrascale (ZU) devices. The highest frequency generated by Elongate with a zero-error constraint is 360 MHz for the ZU device. This is sig-

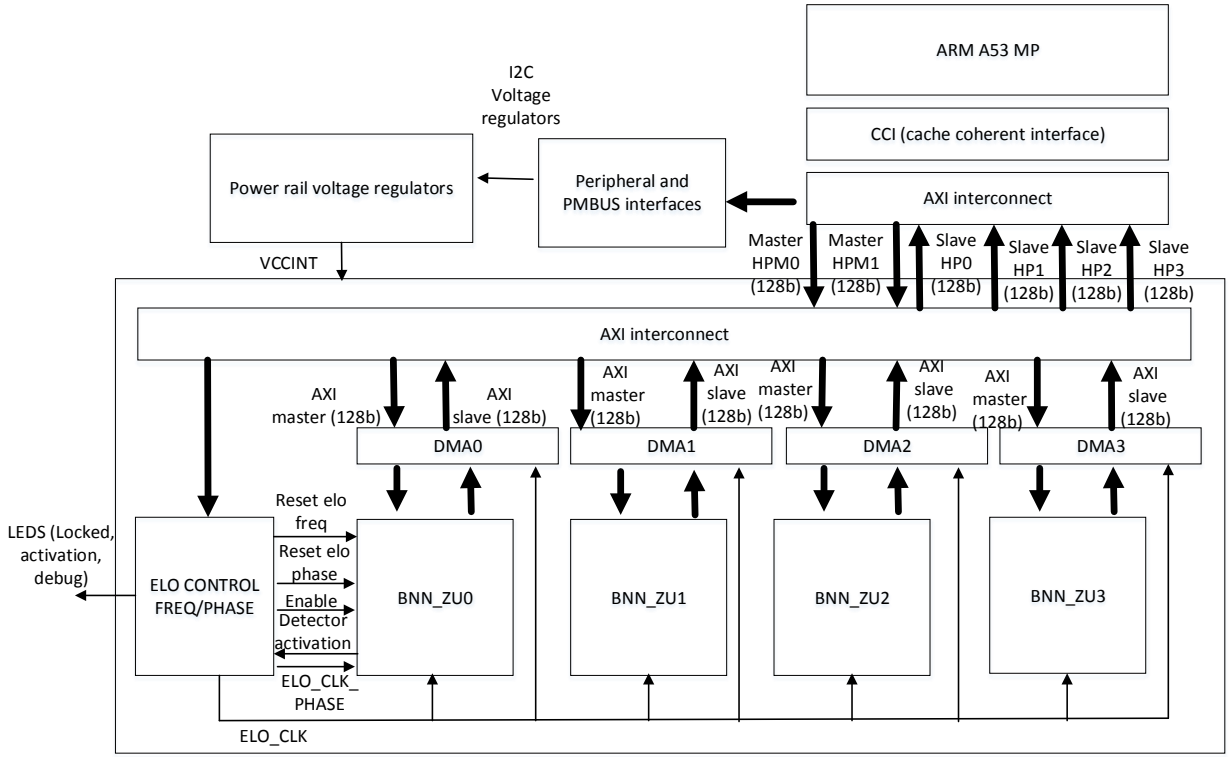

Figure 2: BNN architecture

FINN topology
Input (32x32 RGB image)
3x3-conv-64
3x3-conv-64
pooling
3x3-conv-128
3x3-conv-128
pooling
3x3-conv-256
3x3-conv-256
FC-64
FC-64
FC-64

Figure 3: BNN topology

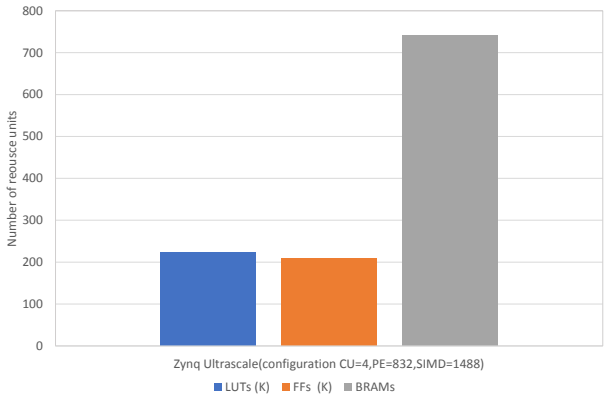

Figure 4: Zynq Ultrascale BNN system hardware resources

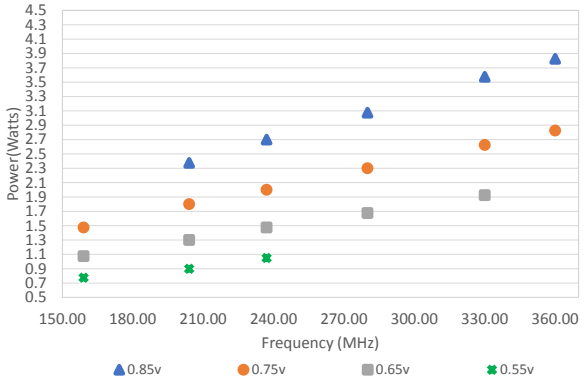

Figure 5: Zynq Ultrascale BNN Power Scaling with CIFAR10

nificantly higher than the nominal 200 MHz for this device. Figure 5 shows that, as expected, power has a linear relation with frequency and that the voltage scaled configurations reduce power significantly since voltage affects both dynamic and static power. The minimum power measured is 0.72 Watts at 160MHz/0.55V for the Ultrascale device. The minimum valid voltage levels for the Zynq Ultrascale device is 0.55v and Elongate logic determines that the maximum frequencies that can be supported at these voltage levels is 160 MHz. These experiments confirm that significant performance and power margins are available in the silicon that can be exploited by Elongate.

Energy and Performance analysis

The multiple frequency and power pairs seen in Figure 5 mean that it is possible to adjust the throughput and power assigned to the task so that computation happens just-in-

time. For example, in a video /image classification problem like the one addressed by the BNN, an initial video sensor could input a large 4K frame and detect regions of interest (ROI) that need to be classified in the neural network. The initial analysis will remove constant backgrounds from further processing in the neural network and will also scale the resulting regions of interest to the frame sizes the neural network handles (32x32 in the case of the FINN BNN). The number of regions in a single frame could vary and range from 0 to thousands depending on the frame activity and the amount of overlapping in ROIs. This means that an energy efficient solution could adapt how much compute throughput is made available to finish just-in-time rather than completing early and then waiting with the corresponding leakage power cost. This is especially relevant in SRAM FPGA technology that needs a full reconfiguration cycle if the device was power gated to reduce leakage during the idle stage. Figure 6 compares the energy and performance obtained with the Elongate configurations with the cases working at nominal voltage and frequency. The figures show that Elongate increases performance up to 86.8% and increases energy efficiency up to 86.3% at the same level of performance. The figures show that the highest performance of BNN in Zynq ultrascale is at 360 MHz achieving 167344 fps. Notice that these high fps are of practical value since although a typical camera might only work at a few hundreds fps the number of 32x32 ROIs in a 4K frame could be much higher (potentially up to 3840 X 2160 or more than 8M fps if we assume 1 pixel displacements) or the streams from several cameras could be processed with a single device.

Accuracy analysis

All the results presented so far have used the neural network at full accuracy so that the activations in the detectors do not affect the functionality of the user logic itself. As

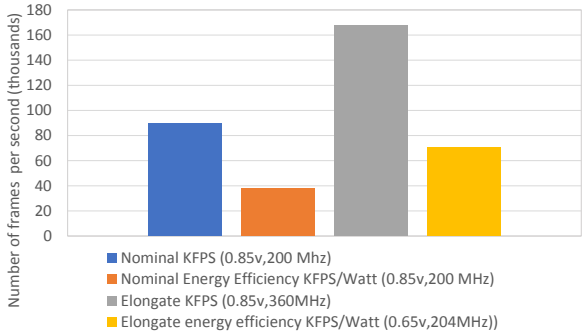

Figure 6: ZYNQ Ultrascale BNN performance on CIFAR10

previously mentioned it is possible to relax this constraint and let errors affect the user logic. Figure 7 shows how the neural network accuracy is affected as the system increases the operating frequency beyond the points found by the detectors. In this experiment, the hardware is processing the first 1000 frames of the CIFAR-10 data set and the obtained error-free classification accuracy is 78.5%. The call-outs indicate the frequency points where the accuracy changes from the error-free accuracy. These points are located at frequencies higher than the frequencies that activate the detectors. As seen in the figures, it is possible to exceed these points by a significant margin and still maintain accuracy within a +1/-1% of the error-free accuracy. For example, for the 0.55v run in Figure 7 there is a maximum frequency of 180 MHz for error-free operation but up to 220 Mhz the accuracy is higher than 77% and only at that point it starts to degrade quickly. This means that a

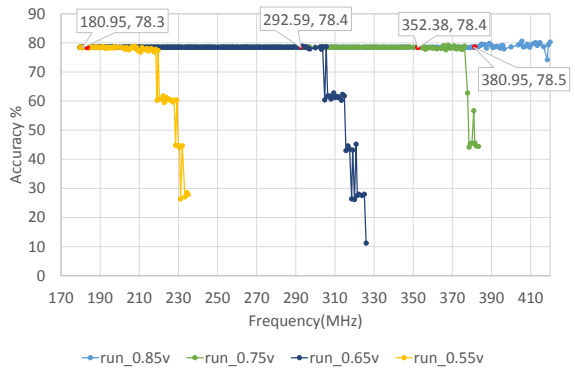

Figure 7: Zynq Ultrascale BNN accuracy on CIFAR10

further increase of 22% in performance is possible at virtually the same level of accuracy. We know that errors are taking place in the logic since the accuracy is not constant but the accuracy does not degrade significantly until we reach a critical point that results in quick degradation. We do not observe a gradual degradation in accuracy as the errors increase. Overall, by exploiting these points located after the first failure it is possible to obtain between 5% to 23% additional performance depending on device and operating point. The conclusion is that the BNN built-in error tolerance could be exploited to increase Elongate performance/energy efficiency higher than the error-free value of 86% if slight variations of classification accuracy are acceptable in the application. However, this additional margin, although present for different voltages, is not constant and the critical point of failure cannot currently be predicted.

Conclusions

In this paper we integrate the Elongate framework with the SDx toolset that enables hardware design based on C/C++. The new framework has then been applied to a fully binarised neural network which is well suited to FPGA devices thanks to the simple binary data paths and low memory complexity. The new Elongated BNN shows higher than 80% improved performance and energy efficiency. As a comparison point the IBM TrueNorth chip measured in [8] with the same CIFAR-10 benchmark achieves 1.2 KFPS and has a power dissipation of 6.11 KFPS/Watt against 167 KFPS and 36 KFPS/Watt in this work (4.6 Watts measured total PL power at maximum performance with 0.85v,360 MHz). Also, the authors in [5] report a peak performance of 40.7 TOPs in their work and compare it with 11.681 TOPs estimated for [8]. Our solution is based on [8] but it uses 4 compute units and clocks 1.8 faster thanks to Elongate. This result in an estimated value of 84.1 TOPs which, we believe, is the highest performance reported to date for a convolutional network accelerator. Finally, the BNN application shows interesting error tolerance features that enable the exploitation of after-point-of-failure states if certain variability of the system accuracy is acceptable. As future work we plan to apply Elongate framework to future versions of the FINN network that will increase precision to more than one bit to represent weights and feature maps while also extending the FINN BNN to deeper topologies such as RESNET able to handle data sets more complex than CIFAR-10 such as Imagenet. A technology demonstrator has been made available at https://github.com/eejlny/Elongate-BNN-demonstrator for the Zynq Ultrascale device.

Acknowledgements

This work was partially supported by Xilinx and UK EPSRC with the ENPOWER (EP/L00321X/1) and the ENEAC

(EP/N002539/1) projects.

REFERENCES

1. Utku Aydonat, Shane O'Connell, Davor Capalija, Andrew C. Ling, and Gordon R. Chiu. 2017. An OpenCL™Deep Learning Accelerator on Arria 10. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '17)*. ACM, New York, NY, USA, 55–64. DOI: http://dx.doi.org/10.1145/3020078.3021738

2. Matthieu Courbariaux and Yoshua Bengio. 2016. BinaryNet: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1. *CoRR* abs/1602.02830 (2016). http://arxiv.org/abs/1602.02830

3. Forrest N. Iandola, Matthew W. Moskewicz, Khalid Ashraf, Song Han, William J. Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. *CoRR* abs/1602.07360 (2016). http://arxiv.org/abs/1602.07360

4. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (Nov 1998), 2278–2324. DOI:http://dx.doi.org/10.1109/5.726791

5. D. J. M. Moss, E. Nurvitadhi, J. Sim, A. Mishra, D. Marr, S. Subhaschandra, and P. H. W. Leong. 2017. High performance binary neural networks on the Xeon+FPGA platform. In *2017 27th International Conference on Field Programmable Logic and Applications (FPL)*. 1–4. DOI: http://dx.doi.org/10.23919/FPL.2017.8056823

6. Jose Nunez-Yanez. 2017. Adaptive voltage scaling in a heterogeneous FPGA device with memory and logic in-situ detectors. *Microprocessors and Microsystems* 51, Supplement C (2017), 227 – 238. DOI: http://dx.doi.org/https://doi.org/10.1016/j.micpro.2017.04.021

7. Eriko Nurvitadhi, Ganesh Venkatesh, Jaewoong Sim, Debbie Marr, Randy Huang, Jason Ong Gee Hock, Yeong Tat Liew, Krishnan Srivatsan, Duncan Moss, Suchit Subhaschandra, and Guy Boudoukh. 2017. Can FPGAs Beat GPUs in Accelerating Next-Generation Deep Neural Networks?. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '17)*. ACM, New York, NY, USA, 5–14. DOI: http://dx.doi.org/10.1145/3020078.3021740

8. Yaman Umuroglu, Nicholas J. Fraser, Giulio Gambardella, Michaela Blott, Philip Leong, Magnus Jahre, and Kees Vissers. 2017. FINN: A Framework for Fast, Scalable Binarized Neural Network Inference. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '17)*. ACM, New York, NY, USA, 65–74. DOI: http://dx.doi.org/10.1145/3020078.3021744