Introducing the 1\textsuperscript{st} ACM ReQuEST Workshop/Tournament on Reproducible Software/Hardware Co-design of Pareto-Efficient Deep Learning

\url{cKnowledge.org/request-cfp-asplos2018.html} \url{cKnowledge.org/request}

c-located with ACM ASPLOS 2018 (Williamsburg, VA, USA)
March 24th, 2018
(proposal submitted in June 2017)

Overview

Artificial Intelligence (AI), Machine Learning (ML) and other emerging workloads demand efficient computer systems from the cloud to the edge. Systems designers, however, face numerous challenges from tackling the ever-growing space of design and optimization choices (including algorithms, models, software frameworks, libraries, hardware platforms, optimization techniques) to balancing off multiple objectives (including accuracy, speed, throughput, power, size, price). Furthermore, the lack of a common experimental framework and methodology makes it even more challenging to keep up with and build upon the latest research advances.

The ACM Reproducible Quality-Efficient Systems Tournaments initiative (ReQuEST) invites a multi-disciplinary community (workloads/software/hardware) to decompose the complex multi-objective benchmarking, co-design and optimization process into customizable workflows with reusable components (see the introduction to ReQuEST). We leverage the open Collective Knowledge workflow framework (CK) and the rigorous ACM artifact evaluation methodology (AE) to allow the community collaboratively explore quality vs. efficiency trade-offs for rapidly evolving workloads across diverse systems.

The 1\textsuperscript{st} ReQuEST tournament served as a proof-of-concept of our approach. We invited the community to submit complete implementations (code, data, scripts, etc.) for the popular ImageNet object classification challenge. For several weeks, four volunteers collaborated with the authors to convert their artifacts into a common CK format and evaluate the converted artifacts on the original or similar platforms. The evaluation metrics included accuracy on the ImageNet validation set (50,000 images), latency (seconds per image), throughput (images per second), platform price (dollars) and peak power consumption (Watts).

Since collapsing all metrics into one to select a single winner often results in over-engineered solutions, we have opted instead to select multiple implementations from a Pareto-frontier, based on their uniqueness or simply to obtain a reference implementation. The authors of selected solutions were given an opportunity to share their insights at the associated ReQuEST workshop co-located with the 23rd ACM ASPLOS conference.
The ReQuEST-ASPLOS’18 proceedings, available in the ACM Digital Library, include five papers with Artifact Appendices and a set of ACM reproducibility badges. The papers are accompanied by snapshots of Collective Knowledge workflows covering a very diverse model/software/hardware stack:

- **Models:** MobileNets, ResNet-18, ResNet-50, Inception-v3, VGG16, AlexNet, SSD.
- **Data types:** 8-bit integer, 16-bit floating-point (half), 32-bit floating-point (float).
- **AI frameworks and libraries:** MXNet, TensorFlow, Caffe, Keras, Arm Compute Library, cuDNN, TVM, NNVM.
- **Platforms:** Xilinx Pynq-Z1 FPGA, Arm Cortex CPUs and Arm Mali GPGPUs (Linaro HiKey960 and T-Firefly RK3399), a farm of Raspberry Pi devices, NVIDIA Jetson TX1 and TX2, and Intel Xeon servers in Amazon Web Services, Google Cloud and Microsoft Azure.

The validated ReQuEST-ASPLOS’18 results available on the ReQuEST scoreboard also exhibit amazing diversity:

- **Latency:** 4 .. 500 milliseconds per image
- **Throughput:** 2 .. 465 images per second
- **Top 1 accuracy:** 41 .. 75 percent
- **Top 5 accuracy:** 65 .. 93 percent
- **Model size (pre-trained weights):** 2 .. 130 megabytes
- **Peak power consumption:** 2.5 .. 180 Watts
- **Device frequency:** 100 .. 2600 megahertz
- **Device cost:** 40 .. 1200 dollars
- **Cloud usage cost:** 2.6E-6 .. 9.5E-6 dollars per inference

Most importantly, the community can now access all the above CK workflows under permissive licenses and continue collaborating on them via dedicated GitHub projects. First, the workflows can be automatically adapted to new platforms and environments by either detecting already installed dependencies (e.g. libraries) or rebuilding dependencies via an integrated package manager supporting Linux, Windows, MacOS and Android. Second, the workflows can be customized by swapping in new models, data sets, frameworks, libraries, and so on. Third, the workflows can be extended to expose new design and optimization choices (e.g. quantization), as well as evaluation metrics (e.g. power or memory consumption). Finally, the workflows can be used for collaborative autotuning (“crowd-tuning”) to explore huge optimization spaces using devices such as Android phones and tablets, with best solutions being made available to the community on the online CK scoreboard.
Our overwhelmingly positive experience has also allowed us to critically assess several potential issues with scaling up this approach and suggest how to overcome them:

- Fair competitive benchmarking between different platforms, frameworks and models is hard work. It requires carefully considering model equivalence (e.g. performing the same mix of operations), input equivalence (e.g. preprocessing the inputs in the same way), output equivalence (e.g. validating the outputs for each input, not just calculating the usual aggregate accuracy score), etc. Formalizing the benchmarking requirements and encapsulating them in shared CK components (e.g. using a framework-independent model representation such as ONNX) and workflows (e.g. for input conversion and output validation), should help standardize and automate the benchmarking process.

- Thorough artifact evaluation can take several person-weeks. Each submitted workflow needs to be studied in detail in its original form and then converted into a common format. However, the more reusable CK components (such as workflows, modules/plugins, packages) are shared by the community, the easier the conversion becomes. For example, we have successfully reused several previously shared components for models, frameworks and libraries, as well as the universal CK workflow for program benchmarking and autotuning. We propose to introduce a new ACM reproducibility badge for such unified "plug&play" components. This could eventually lead to creating a "marketplace" for Pareto-efficient implementations (code and data) shared as portable, customizable and reusable CK components.

- Artifact evaluation may require access to expensive computational resources (e.g. cloud instances with 72-core servers), proprietary tools (e.g. Intel compilers), and auxiliary hardware (e.g. power meters). Raising the profile of AE by widely recognizing its benefits and impact should help us obtain access, licensees and sponsorship from the industry and funding agencies.

- Full experimental evaluation can take many weeks (for example, when validating accuracy on 50,000 images on a 100 MHz FPGA board). The AE committee can collaborate with the authors to determine a minimally useful scope for evaluation which would still provide insights to the community. The community can eventually crowdsource full evaluation. In other words, AE can be "staged" with a quick check that the artifacts are "functional" before the camera-ready deadline followed by full evaluation using the ReQuEST methodology. In fact, ReQuEST can grow into a non-profit service to conferences and journals. Sponsorship should help attract experienced full-time evaluators, as well as part-time volunteers, to work on unifying and evaluating artifacts and workflows.

Our future plans include:

- collaborating with the community, our Advisory Board and ACM to address the above issues;
- using the ReQuEST experience to assist AE at the upcoming SysML'19 conference;
- replacing non-representative benchmarks with realistic workloads;
- creating realistic training sets based on mispredictions shared by the community;
- improving the benchmarking and co-design methodology, and contributing to emerging benchmarking initiatives such as MLPerf;
- collaborating with other competitions such as LPIRC, DAWNBench and SCC on developing a common experimental framework;
- standardizing multi-objective autotuning and co-design workflows;
- extending unified collection of platform information in CK;
- improving and documenting the experimental framework and scoreboard;
- generating reproducible and interactive reports (see examples 1 and 2);
- adding new shared components such as workloads, data sets, tools and platforms;
- automating AE "at the source" by integrating CK workflows with e.g. HotCRP;
- standardizing APIs and meta-descriptions of shared components to make them "marketplace-ready";
- running new ReQuEST competitions for other workloads!

We are also organizing a related ResCuE-HPC workshop at Supercomputing '18 to discuss with the community some of the above issues focusing on how to develop reproducible, customizable and portable workflows for high-performance computing (HPC).
Our long-term vision is to dramatically reduce the complexity and costs of the development and deployment of AI, ML and other emerging workloads. We believe that having an open repository (marketplace) of customizable workflows with reusable components helps to bring together the multidisciplinary community to collaboratively co-design, optimize and autotune computer systems across the full model/software/hardware stack. Systems integrators will also benefit from being able to assemble complete solutions by adapting such reusable components to their specific usage scenarios, requirements and constraints. We envision that our community-driven approach and decentralized marketplace will help accelerate adoption and technology transfer of novel AI/ML techniques similar to the open-source movement.

Acknowledgments

We thank the ReQuEST Advisory Board for their enthusiastic support of our vision; the ReQuEST authors for being very responsive when converting their workflows to the CK format and during artifact evaluation; Flavio Vella and Nikolai Chunosov for their help with unifying and evaluating submissions; Xipeng Shen and James Tuck for their support for organizing the ReQuEST workshop at ASPLOS’18; Craig Rodkin, Asad Ali and Wayne Graves for helping to prepare the ACM DL proceedings with CK workflows, and the CK community for their contributions.

Organizers (A-Z)

- Luis Ceze, University of Washington, USA
- Natalie Enright Jerger, University of Toronto, Canada
- Babak Falsafi, EPFL, Switzerland
- Grigori Fursin, cTuning foundation, France (reproducibility and common workflow framework)
- Anton Lokhmotov, dividiti, UK (industrial relations)
- Thierry Moreau, University of Washington, USA (workshop organization)
- Adrian Sampson, Cornell University, USA
- Phillip Stanley Marbell, University of Cambridge, UK

Advisory board (A-Z)

- Michaela Blott, Xilinx
- Unmesh Bordoloi, General Motors
- Ofer Dekel, Microsoft
- Maria Girone, CERN openlab
- Wayne Graves, ACM
- Vinod Grover, NVIDIA
- Sumit Gupta, IBM
- James Hetherington, Alan Turing Institute
- Steve Keckler, NVIDIA
- Wei Li, Intel
- Colin Osborne, Arm
- Andrew Putnam, Microsoft
- Boris Shulkin, Magna
- Greg Stoner, AMD
- Alex Wade, Chan Zuckerberg Initiative
- Peng Wu, Huawei
- Cliff Young, Google
<table>
<thead>
<tr>
<th>Paper</th>
<th>Artifact available</th>
<th>Artifact functional</th>
<th>Artifact reusable</th>
<th>Results reproduced</th>
<th>Results replicated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highly Efficient 8-bit Low Precision Inference of Convolutional Neural Networks with IntelCaffe</td>
<td><img src="Asset1" alt="Green" /></td>
<td><img src="Asset2" alt="Red" /></td>
<td><img src="Asset3" alt="Red" /></td>
<td><img src="Asset4" alt="Blue" /></td>
<td><img src="Asset5" alt="Blue" /></td>
</tr>
<tr>
<td>Jiong Gong, Haihao Shen, Guoming Zhang, Xiaoli Liu, Shane Li, Ge Jin, Niharika Maheshwari, Evarist Fomenko, Eden Segal</td>
<td><a href="#">Paper DOI</a></td>
<td><a href="#">Artifact DOI</a></td>
<td><a href="#">Original artifact</a></td>
<td><a href="#">CK workflow</a></td>
<td><a href="#">CK results</a></td>
</tr>
<tr>
<td>Optimizing Deep Learning Workloads on ARM GPU with TVM</td>
<td><img src="Asset6" alt="Green" /></td>
<td><img src="Asset7" alt="Red" /></td>
<td><img src="Asset8" alt="Red" /></td>
<td><img src="Asset9" alt="Blue" /></td>
<td><img src="Asset10" alt="Blue" /></td>
</tr>
<tr>
<td>Lianmin Zheng, Tianqi Chen</td>
<td><a href="#">Paper DOI</a></td>
<td><a href="#">Artifact DOI</a></td>
<td><a href="#">Original artifact</a></td>
<td><a href="#">CK workflow</a></td>
<td><a href="#">CK results</a></td>
</tr>
<tr>
<td>Real-Time Image Recognition Using Collaborative IoT Devices</td>
<td><img src="Asset11" alt="Green" /></td>
<td><img src="Asset12" alt="Red" /></td>
<td><img src="Asset13" alt="Red" /></td>
<td><img src="Asset14" alt="Blue" /></td>
<td><img src="Asset15" alt="Blue" /></td>
</tr>
<tr>
<td>Ramyad Hadidi, Jiashen Cao, Matthew Woodward, Michael S. Ryoo, Hyesoon Kim</td>
<td><a href="#">Paper DOI</a></td>
<td><a href="#">Artifact DOI</a></td>
<td><a href="#">Original artifact</a></td>
<td><a href="#">CK workflow</a></td>
<td><a href="#">CK results</a></td>
</tr>
<tr>
<td>Leveraging the VTA-TVM Hardware-Software Stack for FPGA Acceleration of 8-bit ResNet-18 Inference</td>
<td><img src="Asset16" alt="Green" /></td>
<td><img src="Asset17" alt="Red" /></td>
<td><img src="Asset18" alt="Red" /></td>
<td><img src="Asset19" alt="Blue" /></td>
<td><img src="Asset20" alt="Blue" /></td>
</tr>
<tr>
<td>Thierry Moreau, Tianqi Chen, Luis Ceze</td>
<td><a href="#">Paper DOI</a></td>
<td><a href="#">Artifact DOI</a></td>
<td><a href="#">Original artifact</a></td>
<td><a href="#">CK workflow</a></td>
<td><a href="#">CK results</a></td>
</tr>
<tr>
<td>Multi-objective autotuning of MobileNets across the full software/hardware stack</td>
<td><img src="Asset21" alt="Green" /></td>
<td><img src="Asset22" alt="Red" /></td>
<td><img src="Asset23" alt="Red" /></td>
<td><img src="Asset24" alt="Blue" /></td>
<td><img src="Asset25" alt="Blue" /></td>
</tr>
<tr>
<td>Anton Lokhmotov, Nikolay Chunosov, Flavio Vella, Grigori Fursin</td>
<td><a href="#">Paper DOI</a></td>
<td><a href="#">Artifact DOI</a></td>
<td><a href="#">CK workflow</a></td>
<td><a href="#">CK results</a></td>
<td><a href="#">CK results</a></td>
</tr>
</tbody>
</table>