Future of Benchmarking

Simon Narduzzi, CSEM / ETH


Which one is the best?
Which one is the best for a pie?


Which one is the best?
Which one is the best with a pi?

Bechmarking challenges and techniques

Benchmarking for Tiny ML systems

Constraint environment
- Sub-mW
- 4 order of magnitude smaller than MLPerf
- Limited memory (SRAM, Flash)

Wide range of use-cases
- Audio wake words
- Visual wake-up words
- Activity recognition for IMU
- Anomaly detection
- AR Glasses
- Etc…

Datasets
- Open-source datasets that are large are not TinyML specific
- *Lack of large, TinyML-focused dataset*

Models
- NN networks are largely used
- Classic ML (Decision Trees, SVMs)
- *No "MobileNet" for TinyML devices*

TinyMLPerf Benchmark structure

Input Type	Use Cases	Model Types	Datasets
Audio	Audio Wake Words Context Recognition Control Words Keyword Detection	DNN CNN RNN LSTM	Speech Commands (Warden, 2018a) Audioset (Gemmeke et al., 2017) ExtraSensory (Vaizman et al., 2017)
Image	Visual Wake Words Object Detection Image Classification Gesture Recognition Object Counting Text Recognition	DNN CNN SVM Decision Trees KNN Linear	Visual Wake Words (Chowdhery et al., 2019) CIFAR10 (Krizhevsky et al., 2009b) MNIST (LeCun & Cortes, 2010) ImageNet (Deng et al., 2009) DVS128 Gesture (Amir et al., 2017)
Physiological / Behavioral Metrics	Segmentation Forecasting Activity Detection	DNN Decision Tree SVM Linear	Physionet (Goldberger et al., 2000) HAR (Cramariuc, 2019) DSA (Altun et al., 2010) Opportunity (Roggen et al., 2010) UCI EMG (Lobov et al., 2018)
Industry Telemetry	Sensing (light, temp, etc) Anomaly Detection Motor Control Predictive Maintenance	DNN Decision Tree SVM Linear Naive Bayes	UCI Air Quality (De Vito et al., 2008) UCI Gas (Vergara et al., 2012) NASA's PCoE (Saxena & Goebel, 2008)

Banbury, Colby R., et al. "Benchmarking TinyML systems: Challenges and direction." *arXiv preprint arXiv:2003.04821* (2020).

Challenges of benchmarking the devices

Consumption variation

- Across devices
- Relative to accuracy

Power management measurement

- Preprocessing
- Datapath
- Firmware
- Peripherals

Limited memory:

- Benchmark might be too big to fit
- Overhead impacts power consumption
- Quantization support

Hardware heterogeneity

- Event-based
- Memory compute
- MCU with different performance, power, capabilities
- No normalisation procedure defined yet.

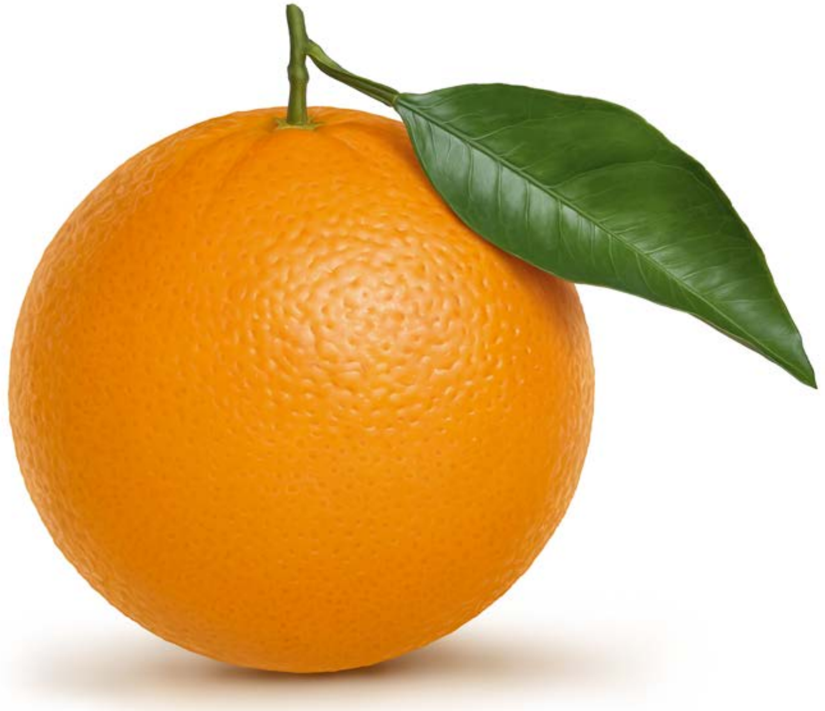
Software heterogeneity

- Hand-coding
- Code generation
- ML interpreter (TensorflowLite), uPython, PyTorchMobile, …

Challenges

Benchmarks should balance between:

1. Portability
2. Comparability
3. Representativeness
4. Many options for model deployment



Open-Closed divisions (from MLPerf)

CLOSED

1. Same preprocessing
2. Same reference model (*or equivalent*)
3. Same training set
4. Same accuracy

OPEN

Everything else

(still supervised / reinforcement learning-based)

https://github.com/mlcommons/training_policies/blob/master/training_rules.adoc#divisions

Open-closed Division : Example

TASK : Visual Wake-up Words / ref: MobileNet

MobileNet V2
MobileNet V1
Q-MobileNet V1

→ CLOSED

SVM
Auto-Encoder
Spiking NN

→ OPEN

Overview of other potential approaches

Unanswered questions from benchmarks

1. *Given a hardware, what is the best model I can get?*

2. *Given a model, what is the best ASIC design I can get?*

3. *Given a model, what will be its performance on hardware platforms?*

1

Given a hardware, what is the best model I can get ?

Brute force? NO!

Reinforcement learning optimizing Neural Network Architecture

Network architecture search for ultra-low power design

Search space

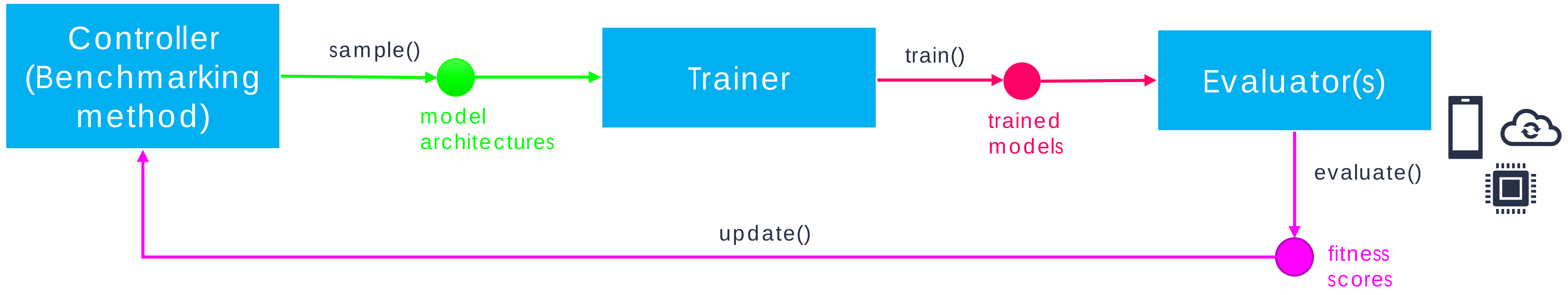
[conv3x3, conv7x7, dense, sepconv,..]

Training constraints

ex: L1, loss, etc…

Deployment

ex: simulation / emulation / real-world deployment



csem

2

Given a model, what is the best design I can get ?

Brute force? NO!

Reinforcement learning optimizing ASIC design for neural network

Hyper-parameters search for ASIC design

Search space

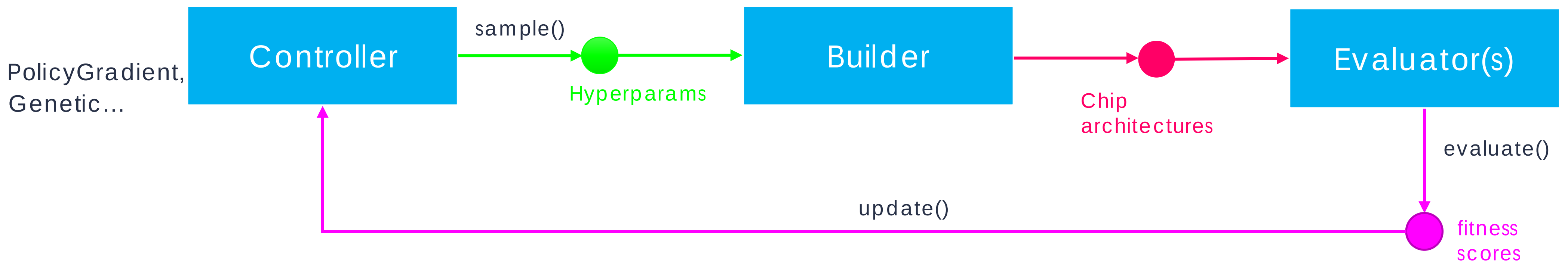
[Memory, CMOS technology…]

Contraints

E.g. Size, fill-factor…

Deployment and fitness score

E.g. penalty for slow processing, etc..



Challenges

- How to choose the NAS algorithm?
- How to make sure this NAS does not diverge for certain hardware?
- Computational time of NAS is quite important
- Need to avoid “cold-start” -> Database of already tested models and accelerators
- Identify good emulation environments

3

Given a model, what will be its performance on hardware platforms?

Run physical test -> takes time!

What if the model does not fit,
but only because of memory?

Projecting the performance of
a model by regression

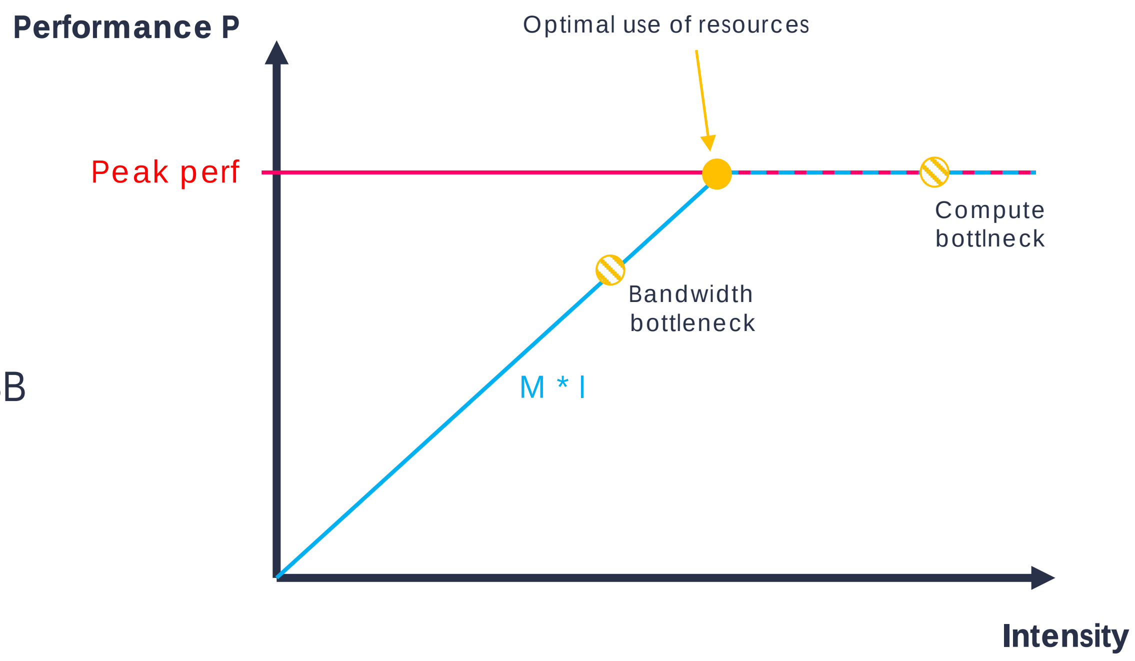
Theoretical Baselines for ML Benchmarking

Peak performance PP [FLOPs/s] **4 GFLOPs/s**

Memory bandwith M [Byte/s] **1 GB/s**

Intensity I [FLOPs/Byte read] 40 FLOPs / 8B = **5 FLOPs/B**

P = min(PP, M * I)

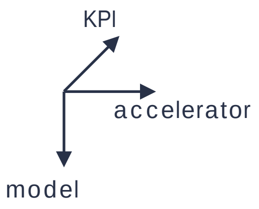


“Roofline: An insightful visual performance model for multicore architectures”, S. Williams et al., 2009

“Evaluating Theoretical Baselines for ML Benchmarking Across Different Accelerators”, M. Blott et al., 2021

Projection-based benchmarks

Accelator1 Accelator2 Accelator3 ... AcceleratorN
Accelator1 Accelator2 Accelator3 ... AcceleratorN



	Accelator1	Accelator2	Accelator3	…	AcceleratorN
Model1	0.88	0.99	0.90	…	0.99
Model2	0.84	0.84	0.85	…	0.81
Model3	0.95	0.97	0.91	…	0.89
Model4	0.81	0.94	0.85	…	0.96
Model5	0.92	0.91	0.88	…	0.98

ModelX

Challenges for Projection-based benchmark

- Cold start problem
- Recommandation-based system?

- How to find good embeddings to allow interpolation?
  - Embeddings for models?
  - Embeddings for accelerators?
  - What about the performance of a model, on a hardware, on a certain dataset?

- How to deal with constraints? Memory, consumption@FPS, etc...?

csem

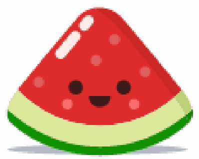
Summary

Remaining challenges:

- Still no “apple-vs-orange” comparison
- Three questions not yet answered
  - What is the best hardware for my model?
  - What is the best model for my hardware?
  - What would be the performance of this model on this hardware?

- Absence of clear “comparison” website for TinyML benchmarks

THANK YOU



simon.narduzzi@csem.ch
narduzzi
Narduzzi