Orchestrating a brighter world NEC

SOL4VE: Running Deep Neural Networks on the NEC SX-Aurora Tsubasa

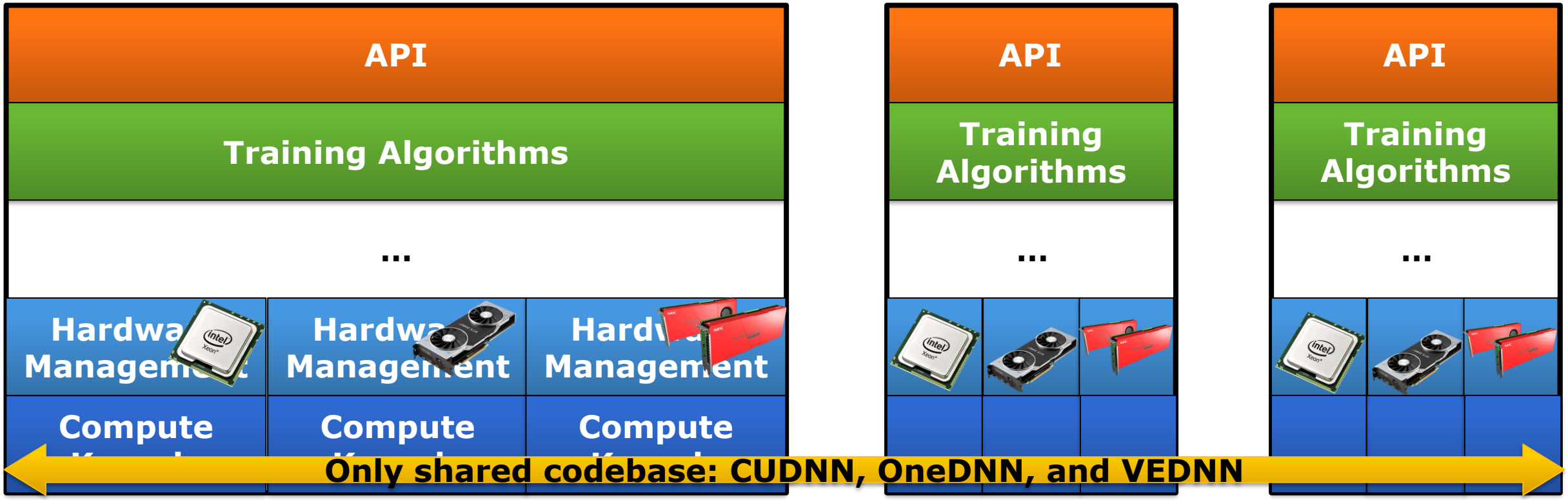
Dr. Nicolas Weber
NEC Laboratories Europe
nicolas.weber@neclab.eu

How AI frameworks work today

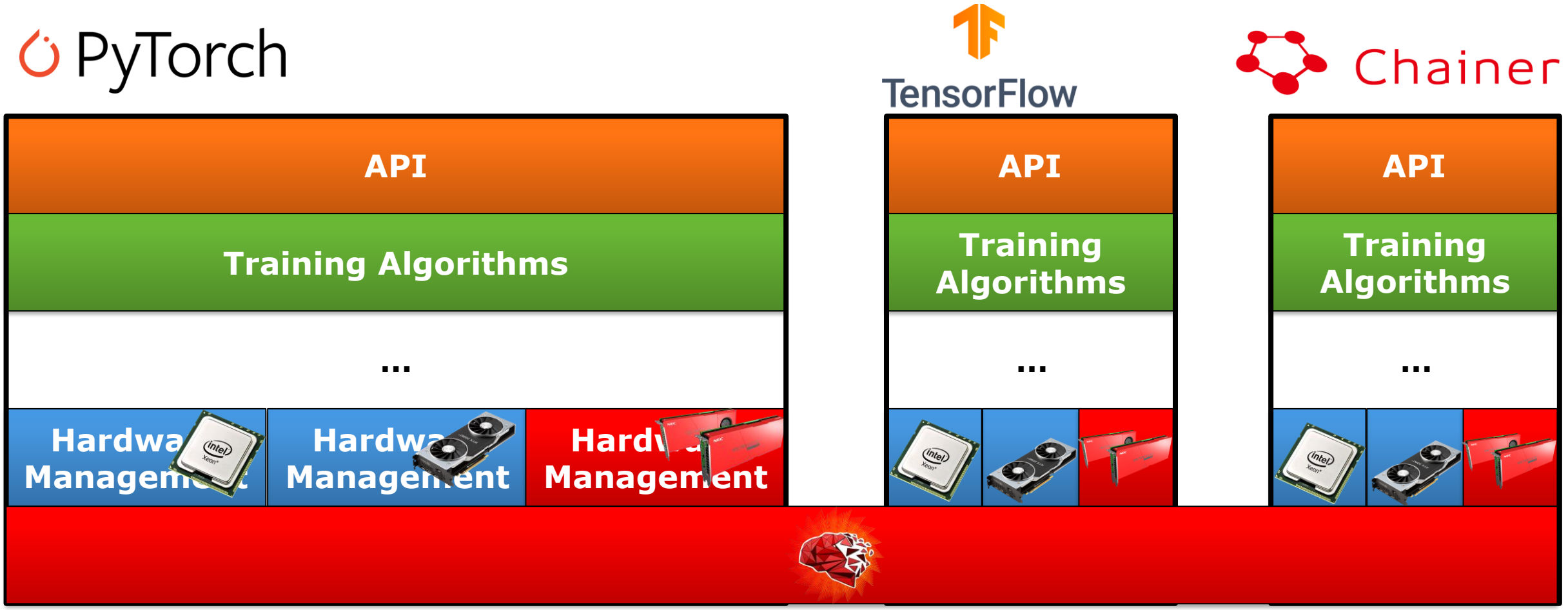
PyTorch

TensorFlow

Chainer



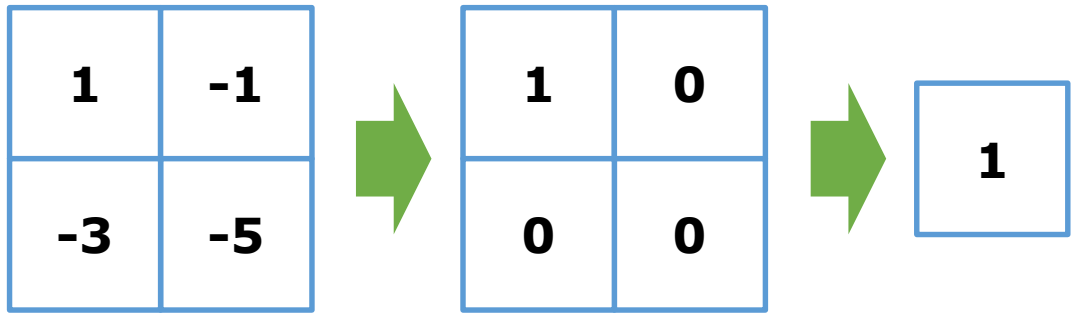
How AI frameworks work today

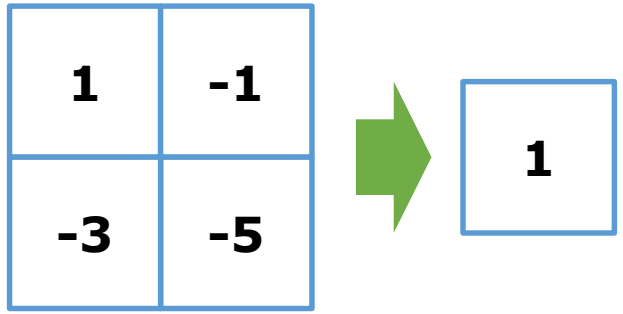


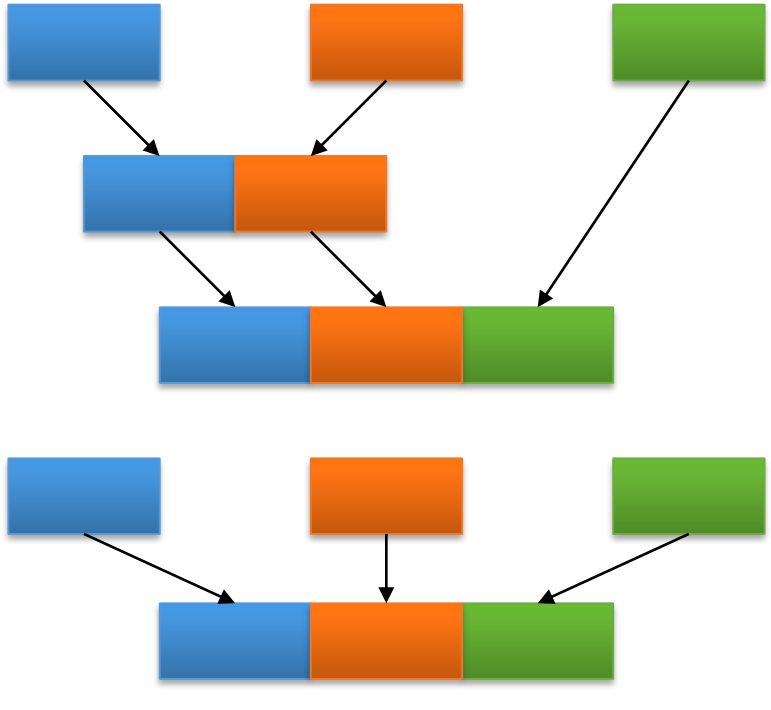
SOL Features: **Performance Optimizations**

High Level Graph optimizations

- MaxPooling(min=-inf) → ReLU or ReLU → MaxPooling(min=-inf) >> MaxPooling(min=0)
- Removal of unnecessary memcopies in chained concat operations (i.e. DenseNet)
- Removal of layers that don't contribute to the output
- Auto-Tuning based planning of low level operators.
- …







SOL Features: **Performance Optimizations**

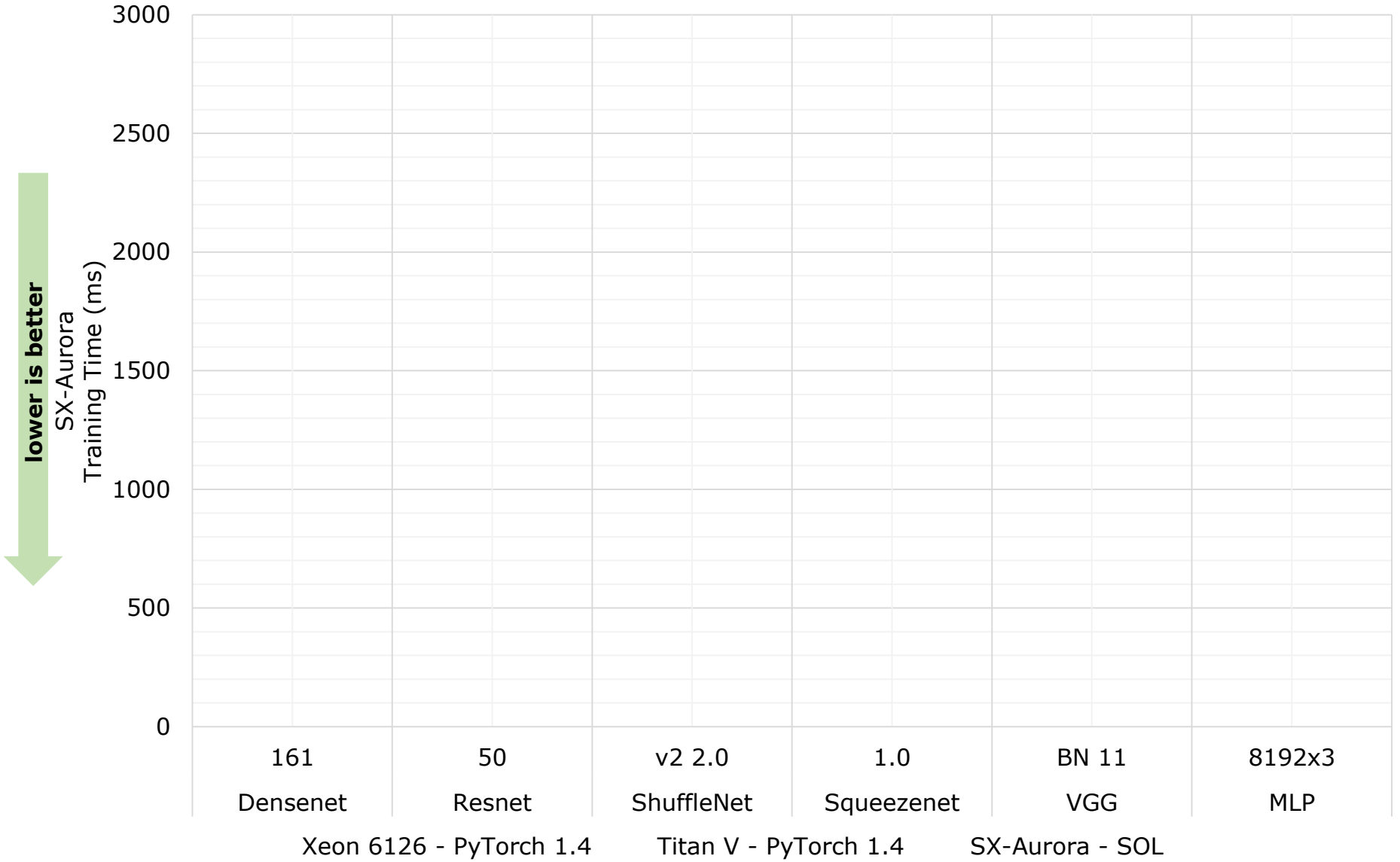
Low Level Optimizations

- Support for various DNN and BLAS libraries (OpenBLAS, MKL, OneDNN, CUDNN, CUBLAS, VEDNN, NLC, NNPack, …)
- **Depth-First Parallelism** based code generator engine for fusing memory bound layers (i.e. Activations, Normalization, Pooling, Lookup-Tables, …) to improve cache utilization.

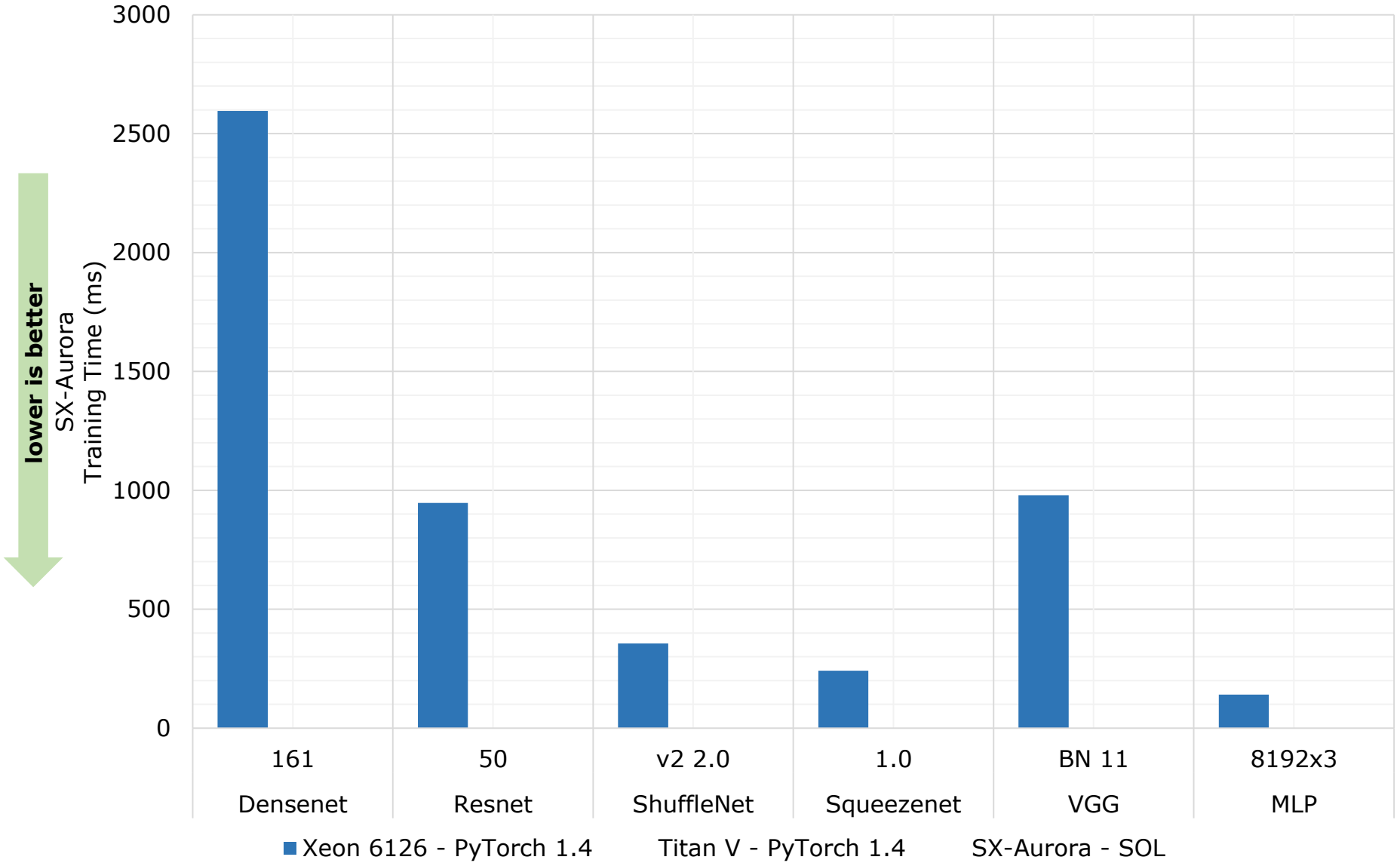
Hardware Specific Optimizations

- For SX-Aurora: only one offloading call to compute entire network (reduces latency)
- …

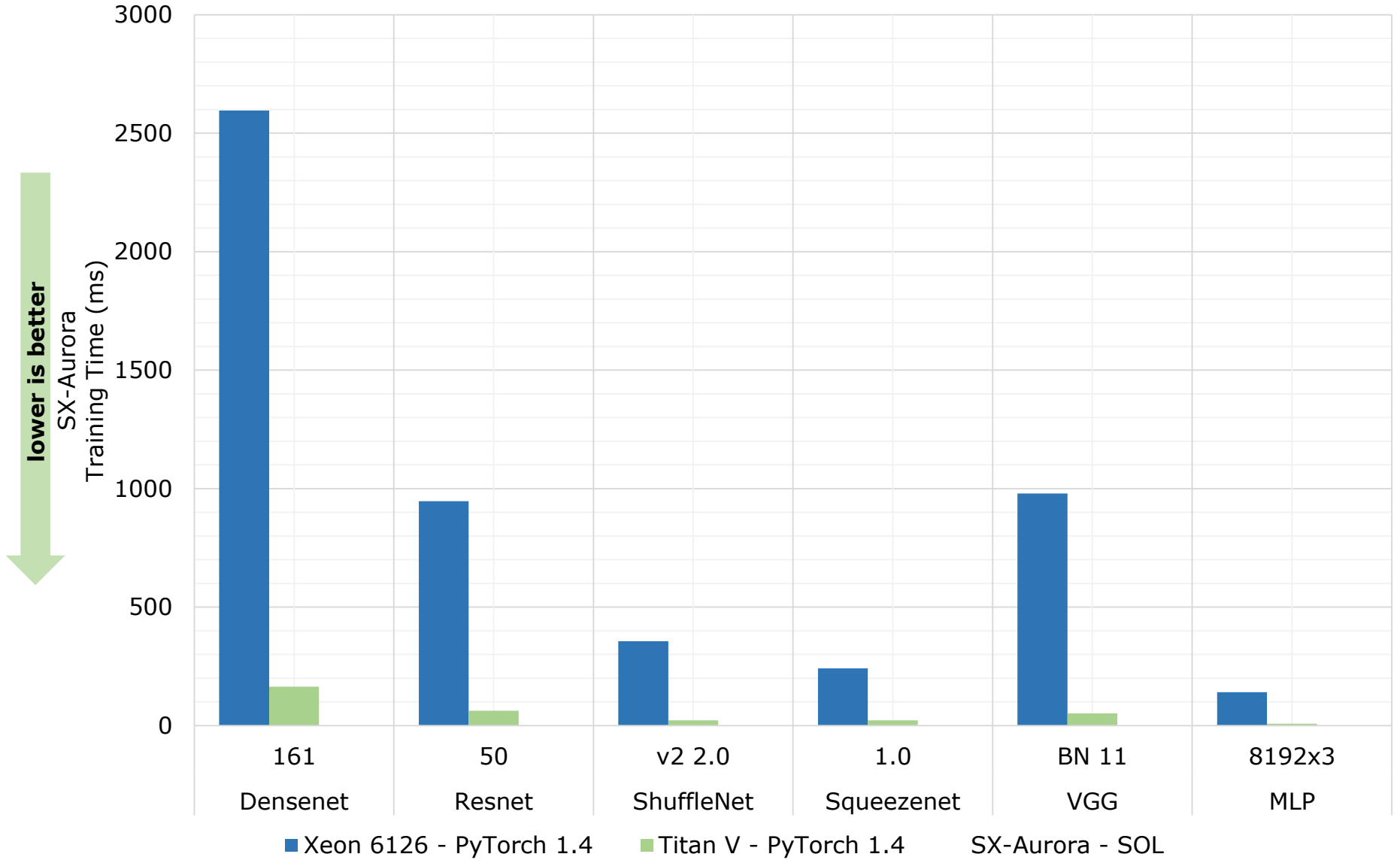
Training Performance (CNN BS=16, MLP BS=64, FP32)



Training Performance (CNN BS=16, MLP BS=64, FP32)

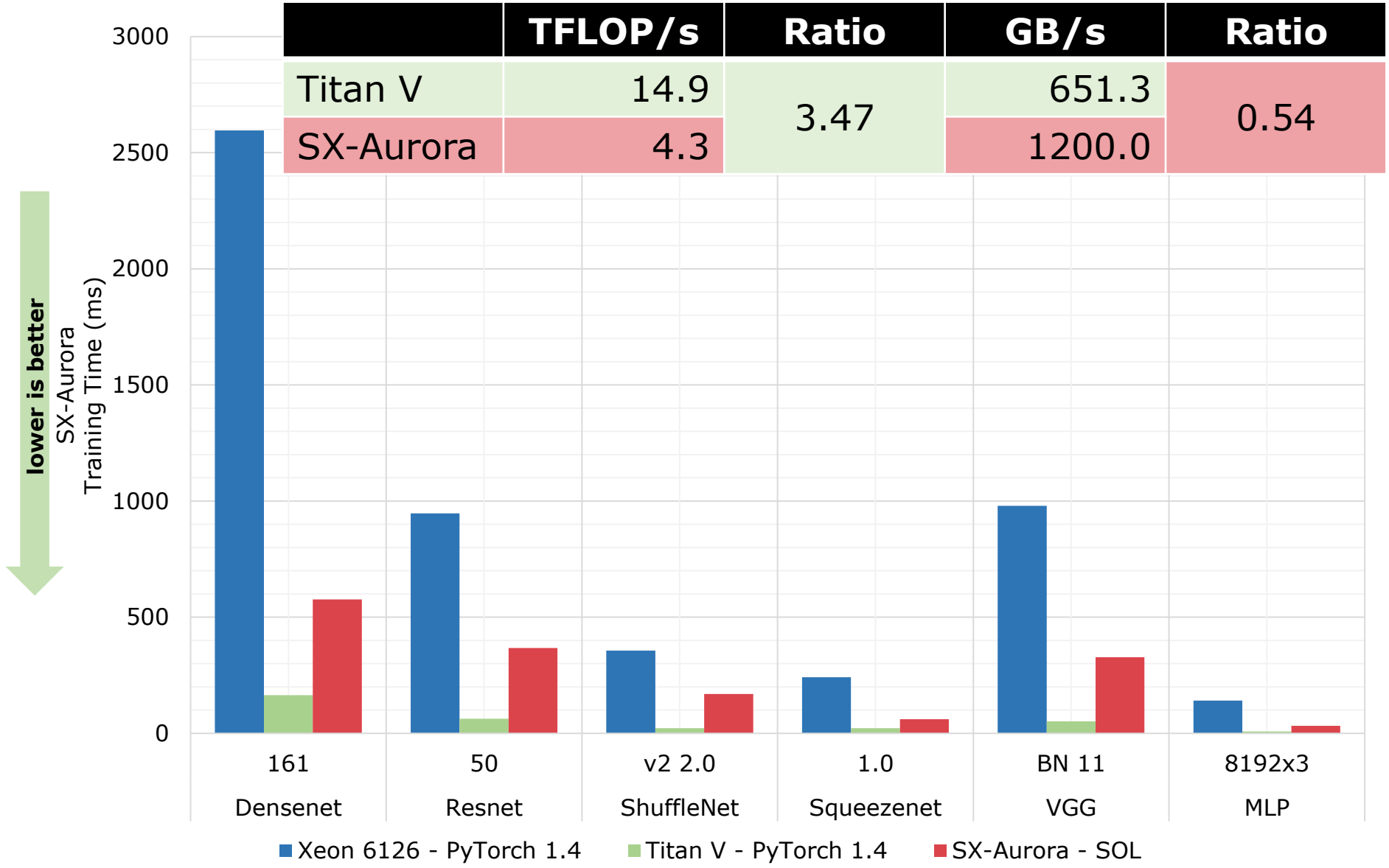


Training Performance (CNN BS=16, MLP BS=64, FP32)

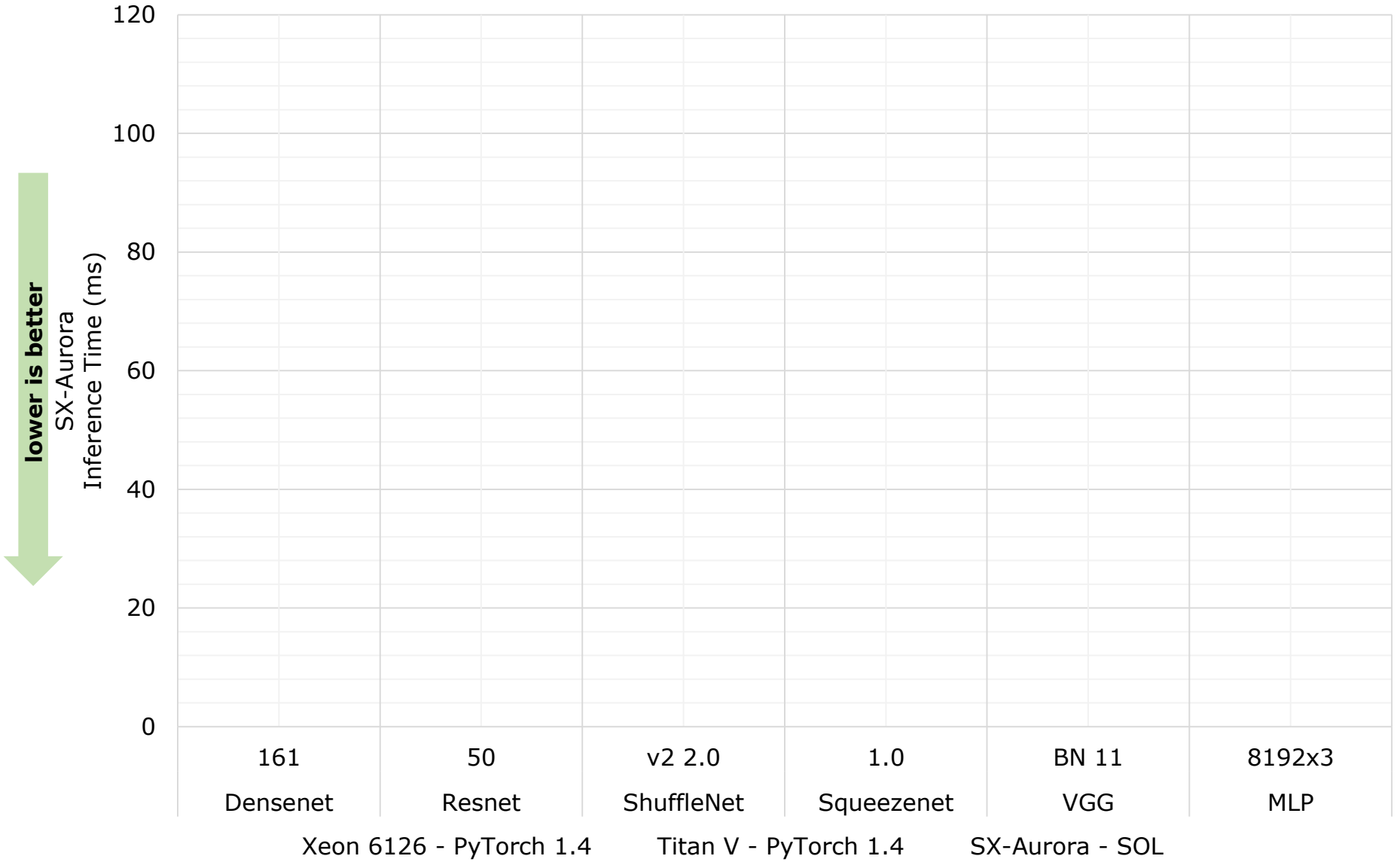


Training Performance (CNN BS=16, MLP BS=64, FP32)

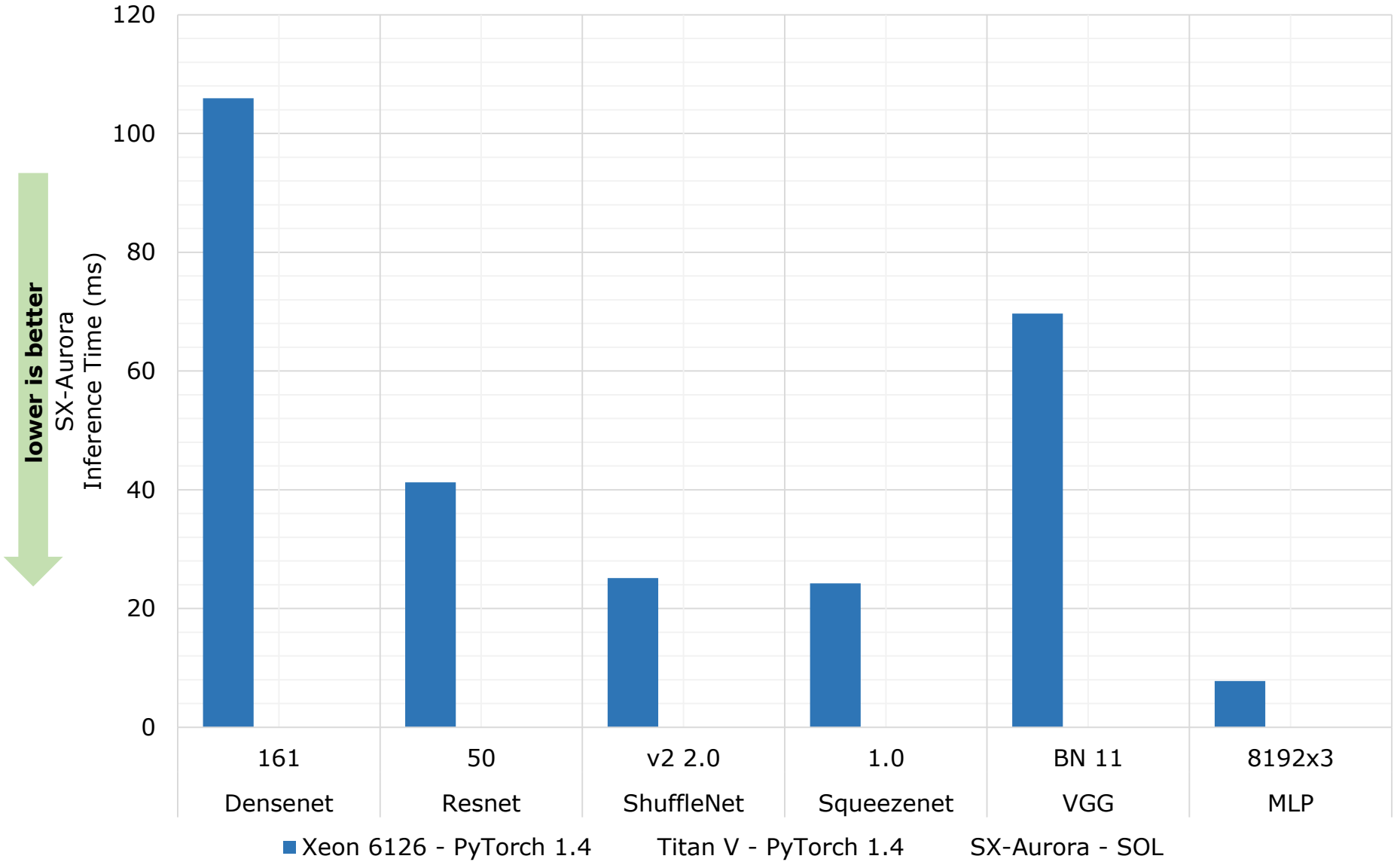
	TFLOP/s	Ratio	GB/s	Ratio
Titan V	14.9	3.47	651.3	0.54
SX-Aurora	4.3		1200.0	



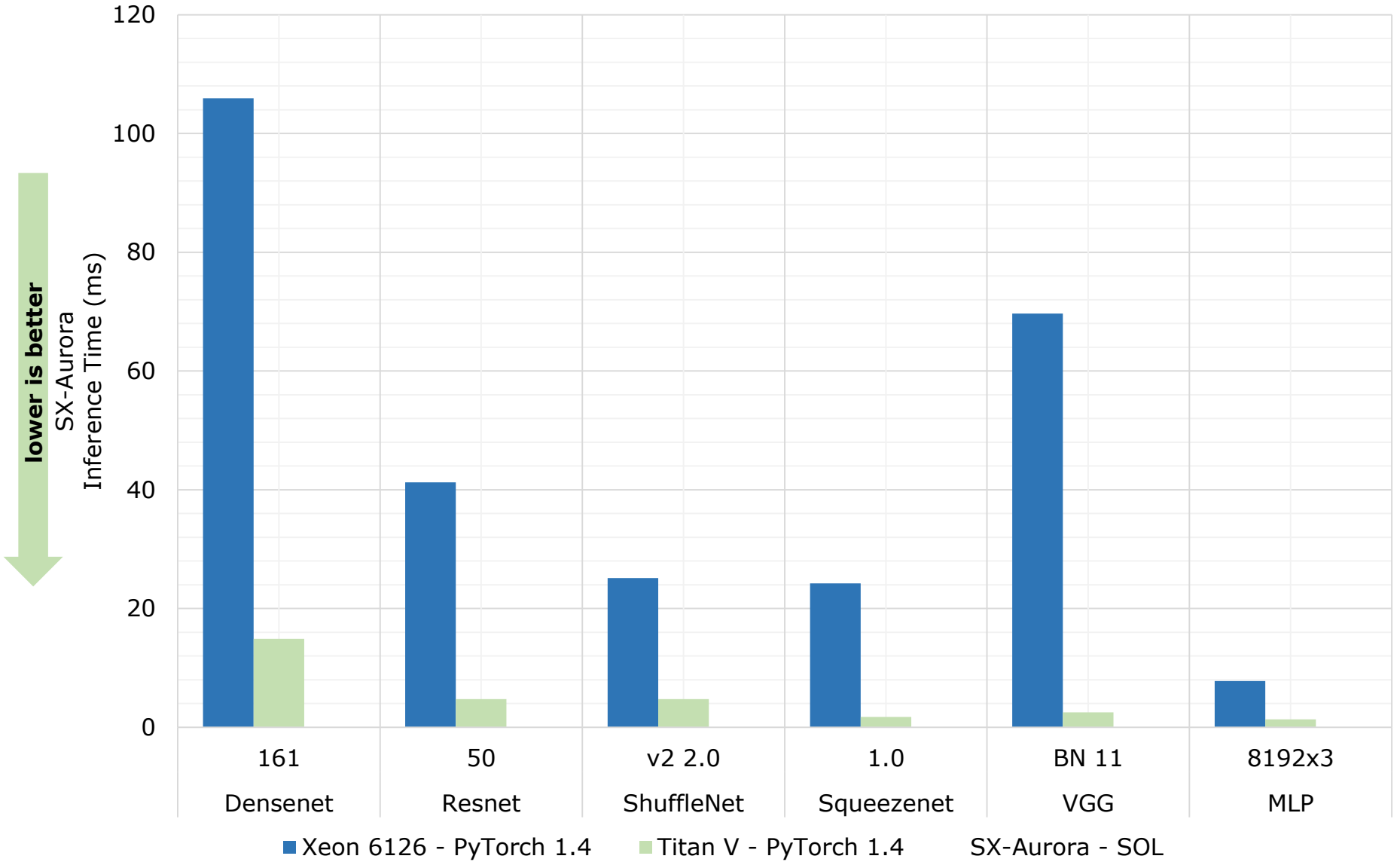
Inference Performance (BS=1, FP32)



Inference Performance (BS=1, FP32)

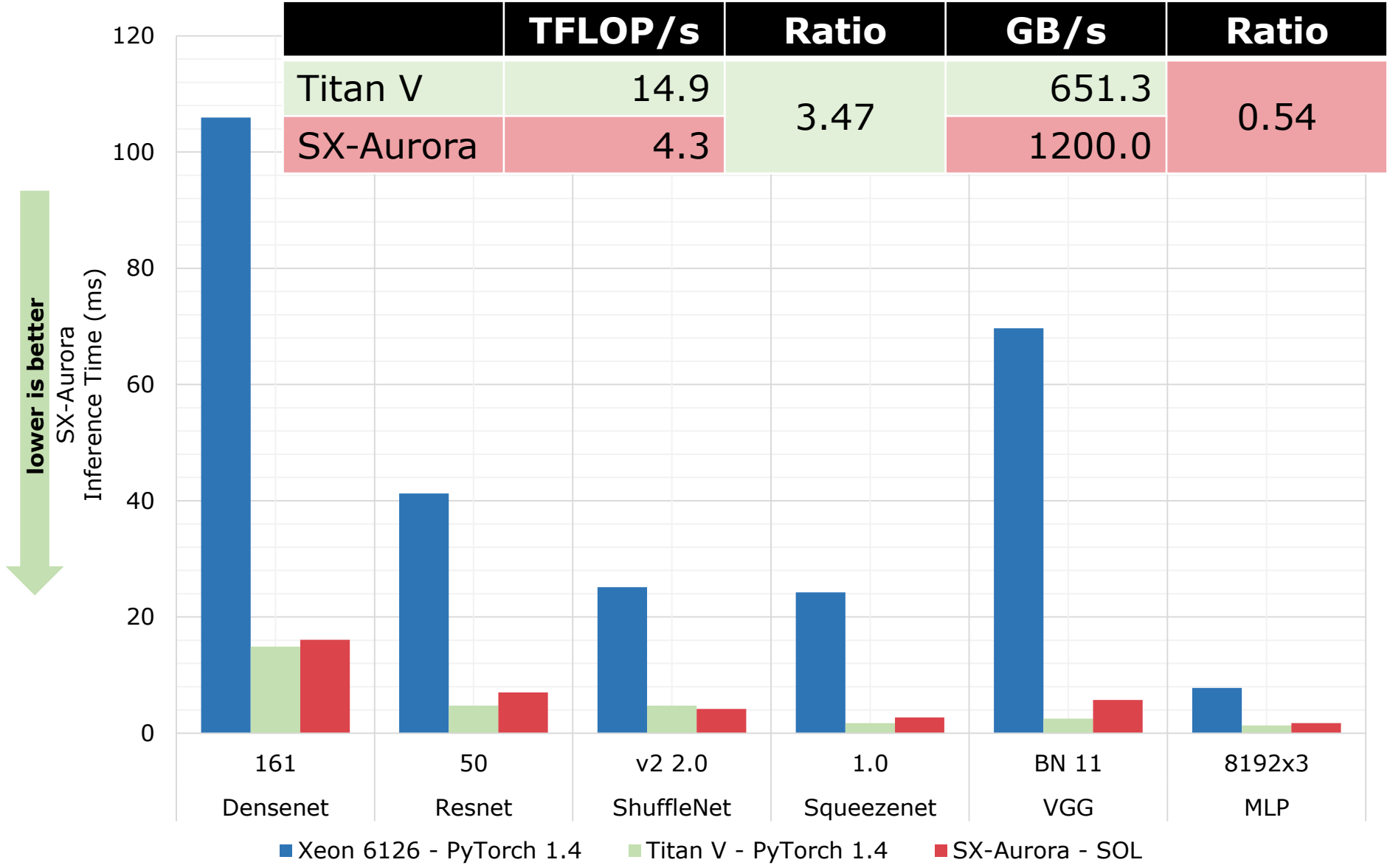


Inference Performance (BS=1, FP32)



Inference Performance (BS=1, FP32)

	TFLOP/s	Ratio	GB/s	Ratio
Titan V	14.9	3.47	651.3	0.54
SX-Aurora	4.3		1200.0	



SOL Features: **Directly run ONNX Models!**

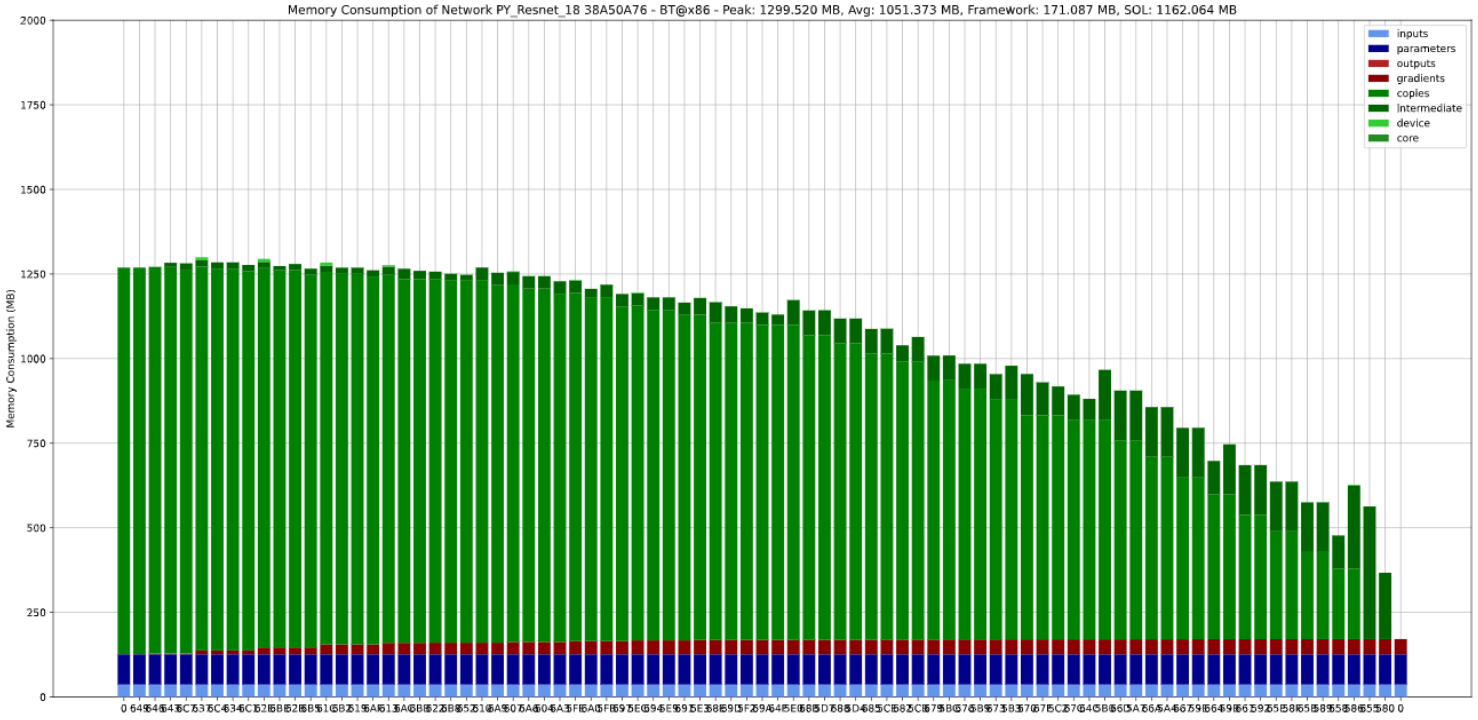
Directly run ONNX Models (coming in next v0.3.2 release)

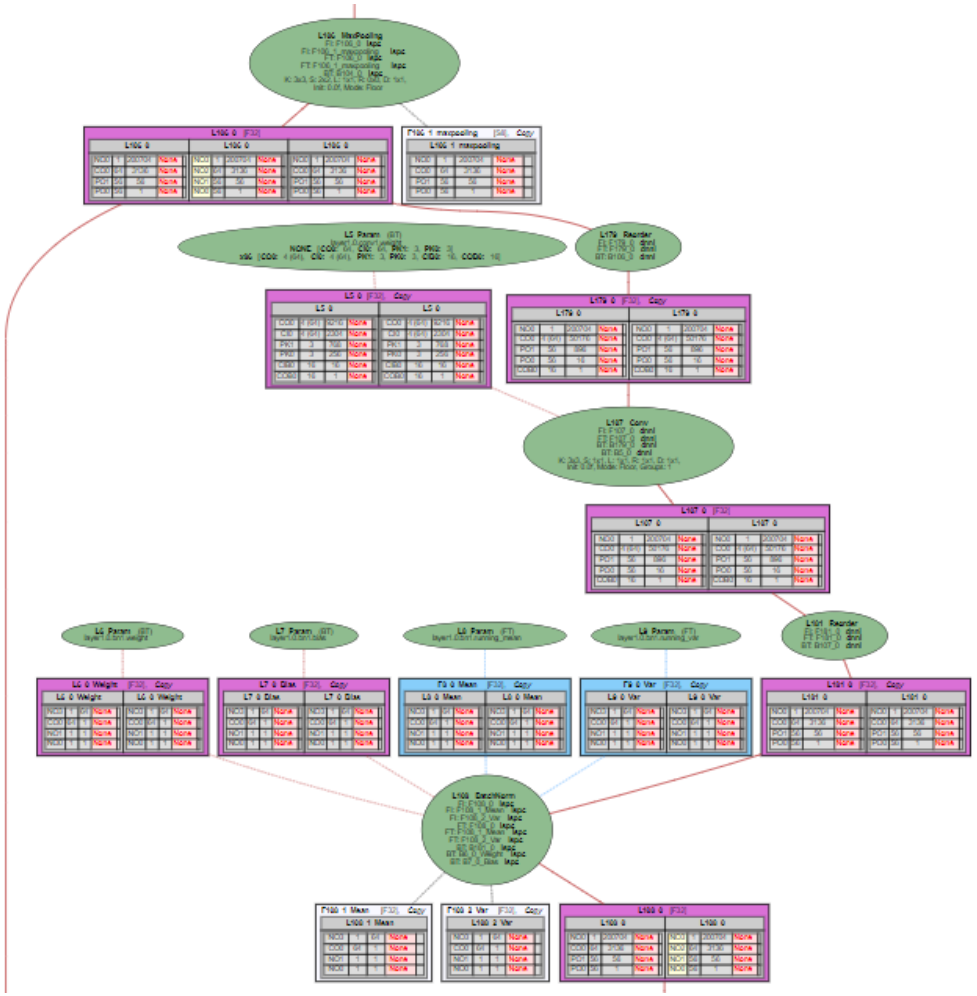
```
# load packages
import sol.onnx as sol
import numpy as np

# load model + optimize with SOL
sol_model = sol.optimize(“myModel.onnx”)
sol.device.set(sol.device.ve, 0)

# run model
input  = np.random.rand(1, 3, 224, 224)
output = sol_model(input)
```

SOL Features: **Advanced Debugging**

Advanced Debugging Capabilities

- investigate structure of your Neural Network
- investigate memory consumption in Inference and Training





SOL Features: **Deployment**

Deployment

- Export your trained Neural Network into your own application!
- Cross compile to any supported hardware:
  - Train on X, deploy to Y
- Removes framework dependencies!

```
sol.deploy(trained_model,
      sol.input(…), …,
target=sol.deployment.shared_l
 ib, device=sol.device.ve, …)
```

```
#ifndef __MyNetwork__
#define __MyNetwork__

#ifdef __cplusplus
extern "C" {
#endif

void predict_init(const int deviceIdx);
int  predict_seed(const int seed);
void predict     (void* ctx, const float* input,
float** output);

#ifdef __cplusplus
}
#endif
#endif
```

PyTorch TensorFlow Chainer

SOL Features: **Overview**

Frameworks

- PyTorch
- Deeplearning4J
- ONNX

Hardware

- NEC SX-Aurora Tsubasa
- X86 CPUs
- ARM64 CPUs
- NVIDIA GPUs

Execution Modes

- Inference
- Training
- Deployment

Convolutional Neural Networks

- Alexnet
- SqueezeNet (1.0, 1.1)
- VGG + BN (11, 13, 16, 19)
- Resnet (18, 34, 50, 101, 152)
- Densenet (121, 161, 169, 201)
- Inception V3
- GoogleNet
- MobileNet (v1, v2)
- MNasNet (0.5, 0.75, 1.0, 1.3)
- ShuffleNet V2 (0.5, 1.0, 1.5, 2.0)
- ResNext (50, 101)
- WideResNet (50, 101)

Multi Layer Perceptron (MLP)

Linear/Logistic Regression

Natural Language Processing

- BERT (PyTorchic + HuggingFace implemenations)

What’s coming in 2021?

Features

- TensorFlow v2
- Numpy
- Recurrent Neural Networks (i.e., LSTM)

Current ideas

- Trade memory consumption for execution time during training
- Cross-Framework execution
  - Load model in TensorFlow, execute in PyTorch
- Custom/User Layer support
- …

How can we get access to SOL?

SOL4VE: **Closed Beta**

Organized by NEC-Germany

Only NEC SX-Aurora support!

GitLab Issue Tracker

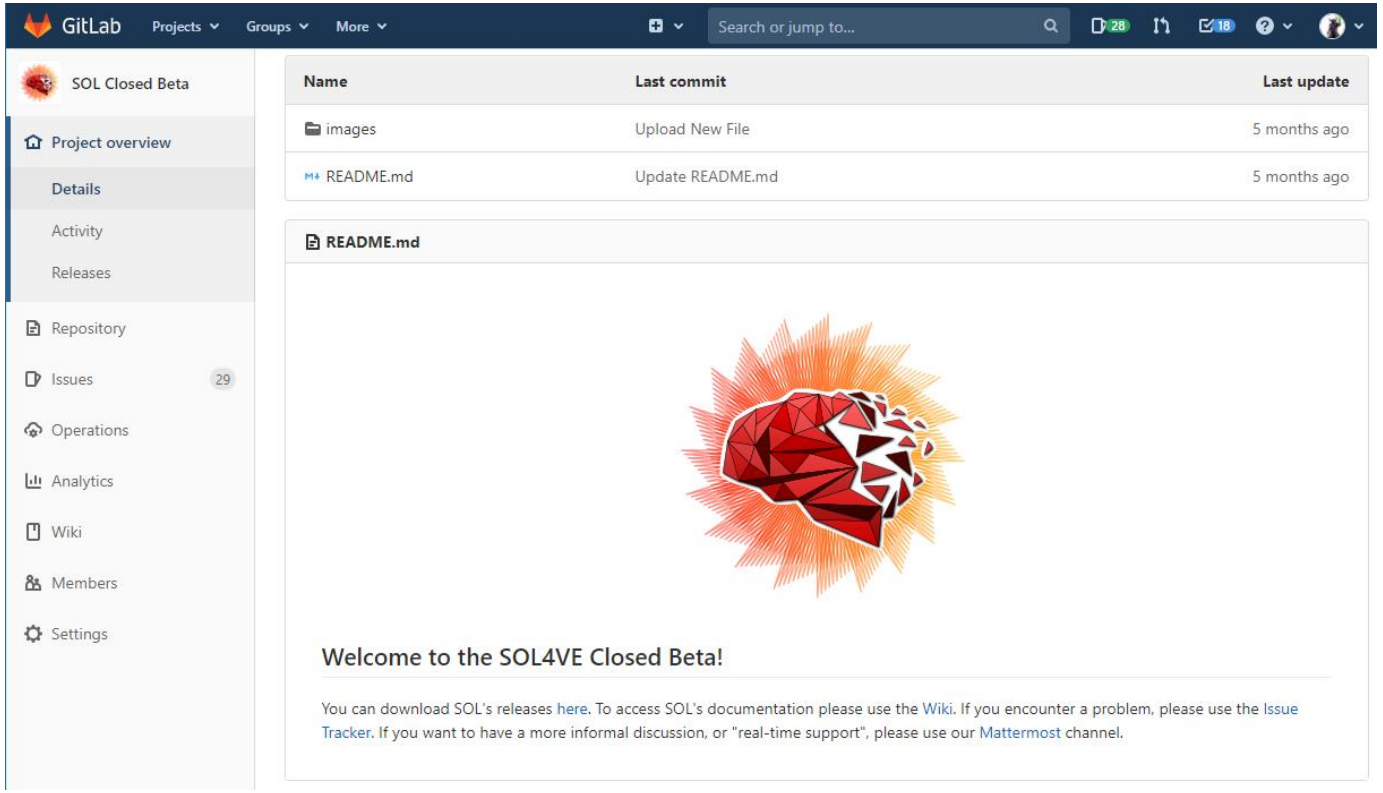
SOL Documentation

PIP installation server

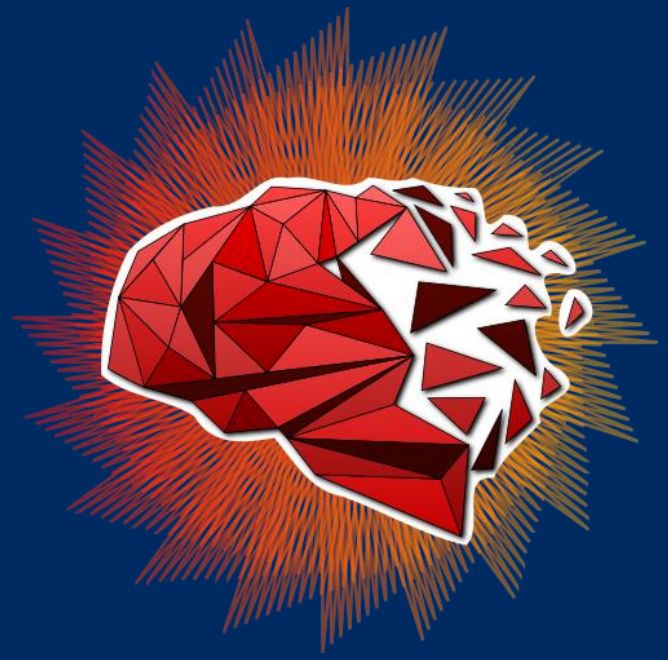
- `pip3 install sol==0.3.1`

Contact:

- Erich.Focht@emea.nec.com
- Nicolas.Weber@neclab.eu



\Orchestrating a brighter world NEC



Dr. Nicolas Weber
Senior Software Engineer
NEC Laboratories Europe
nicolas.weber@neclab.eu

SOL

www.sol-project.org