Luthier: Bridging Auto-Tuning and Vendor Libraries for Efficient Deep Learning Inference

Yongin Kwon, JooHyoung Cha, Sehyeon Oh, Misun Yu, Jeman Park, Jemin Lee

Electronics and Telecommunications Research Institute (ETRI), South Korea

Motivation

Problems with Existing Approaches

Auto-tuning Compilers

- ✓ High flexibility
- ✓ Automation support
- × Very long tuning time
- × No asymmetric multicore support

Vendor Libraries

- ✓ Immediate execution
- ✓ Optimized kernels
- × Lack of flexibility
- × Static workload distribution

Optimization Space Complexity

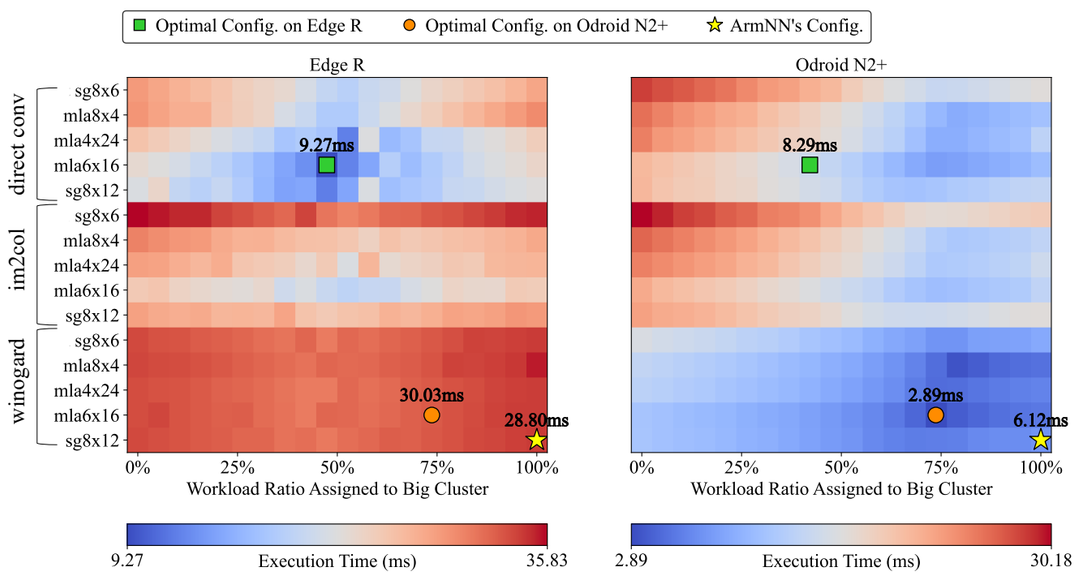

Figure 1. Heatmap of execution time for ResNet18's 2nd conv layer showing vastly different performance across kernel and workload distribution combinations

Key Innovation

Luthier combines the best of both worlds: leverages vendor-optimized kernels while providing automated optimization through machine learning-based cost models for asymmetric multicore processors.

System Architecture

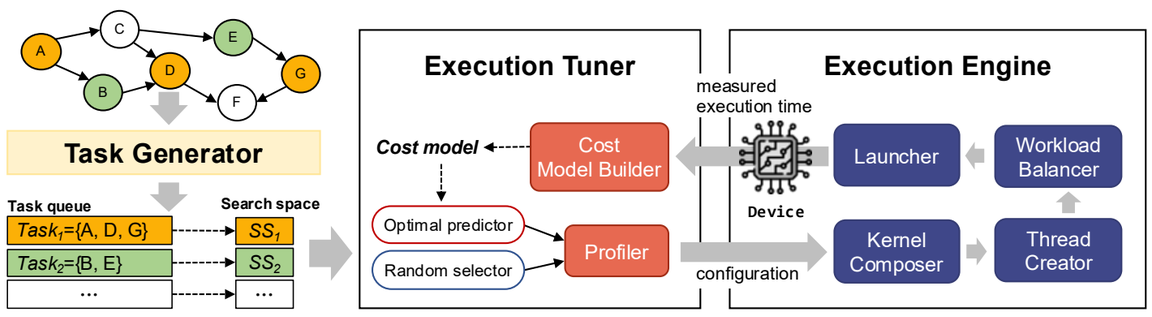

Figure 2. Luthier's system architecture with three main components

Three Core Components:

1. **Task Generator:** Groups operations, reduces ResNet101's 104 ops to 23 tasks
2. **Execution Tuner:** XGBoost-based cost model for optimal configuration
3. **Execution Engine:** Manages kernel execution on CPU/GPU platforms

Experimental Setup

Hardware Platforms

Cluster	Edge R	Odroid N2+	SD865
		Big Cluster	
μArch	A72	A73	Kryo 585 Gold
# Cores	2	4	1
Frequency	1.8 GHz	2.4 GHz	2.84 GHz
		Middle Cluster	
μArch	-	-	Kryo 585 Gold
# Cores	-	-	3
Frequency	-	-	2.42 GHz
		Little Cluster	
μArch	A53	A53	Kryo 585 Silver
# Cores	4	2	4
Frequency	1.4 GHz	2.0 GHz	1.8 GHz
		Mobile GPU	
μArch	Midgard	Bifrost	Adreno 600 Series
# Cores	4	6	2
Frequency	800MHz	800 MHz	587MHz

Performance Results

End-to-End Performance Comparison

	Edge R			Odroid N2+		
Model	**Luthier**	AutoTVM	Ansor	**Luthier**	AutoTVM	Ansor
ResNet18	**1.4x(1.6h)**	1.3x(27.1h)	0.8x(29.2h)	**1.7x(1.6h)**	1.9x(10.1h)	1.4x(26.0h)
ResNet50	**1.6x(2.0h)**	1.2x(19.5h)	1.1x(27.1h)	**2.0x(1.9h)**	1.8x(6.2h)	1.5x(27.1h)
ResNet101	**1.6x(2.7h)**	1.1x(51.5h)	1.0x(46.2h)	**2.0x(2.6h)**	1.7x(23.2h)	1.4x(34.7h)
AlexNet	**1.6x(0.4h)**	0.8x(33.4h)	0.6x(32.7h)	**1.3x(0.4h)**	0.6x(53.7h)	1.4x(25.8h)
VGG16	**1.5x(0.9h)**	1.0x(33.3h)	0.5x(30.4h)	**1.6x(0.8h)**	1.2x(31.0h)	0.8x(25.8h)
GoogLeNet	**1.4x(2.4h)**	1.3x(49.8h)	1.1x(28.5h)	**1.6x(2.3h)**	1.6x(22.6h)	1.5x(23.2h)
MobileNetV2	**1.1x(1.6h)**	1.7x(43.2h)	1.8x(62.7h)	**1.5x(1.5h)**	1.8x(49.8h)	2.8x(23.8h)
Average	**1.5x(1.7h)**	1.1x(39.2h)	0.9x(36.5h)	**1.7x(1.7h)**	1.4x(28.5h)	1.5x(27.0h)

Table 1. Speedup relative to ArmNN baseline (tuning time in parentheses)

Convolution Performance Comparison

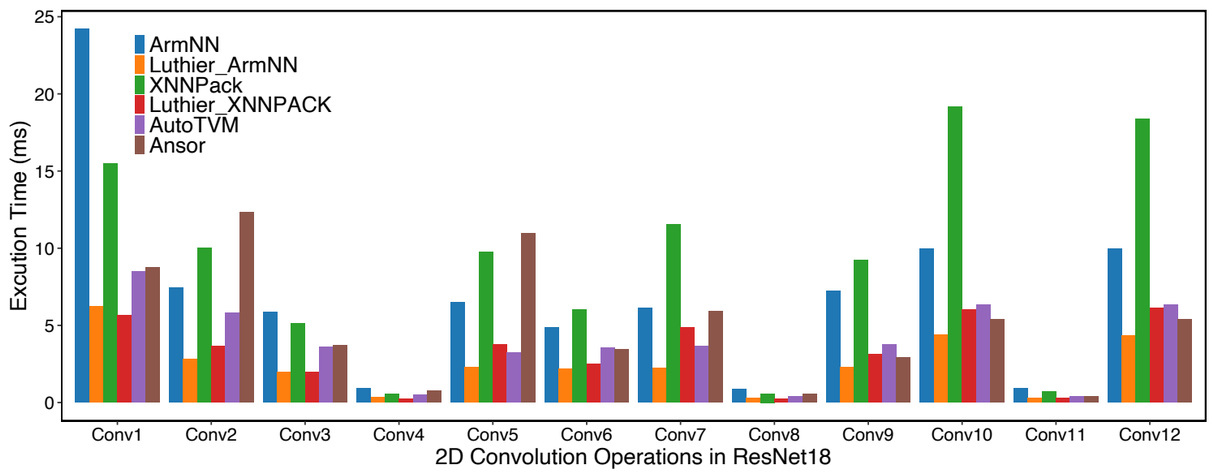

Figure 3. Performance comparison on convolution operations

GEMM Performance

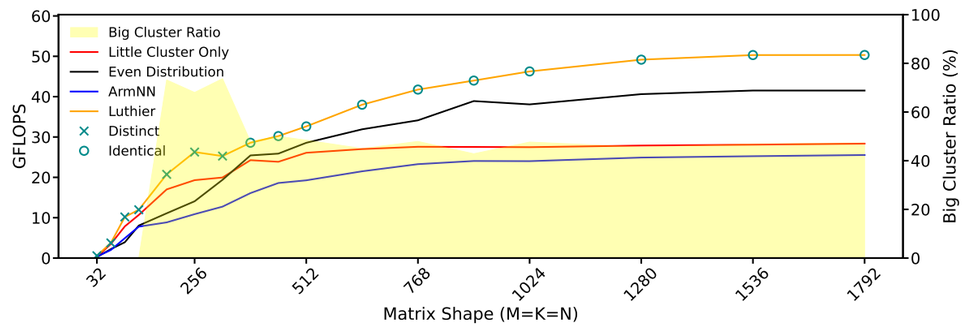

Figure 4. GEMM performance achieving 50 GFLOPS vs ArmNN's 23 GFLOPS

Impact of Tuning Knobs

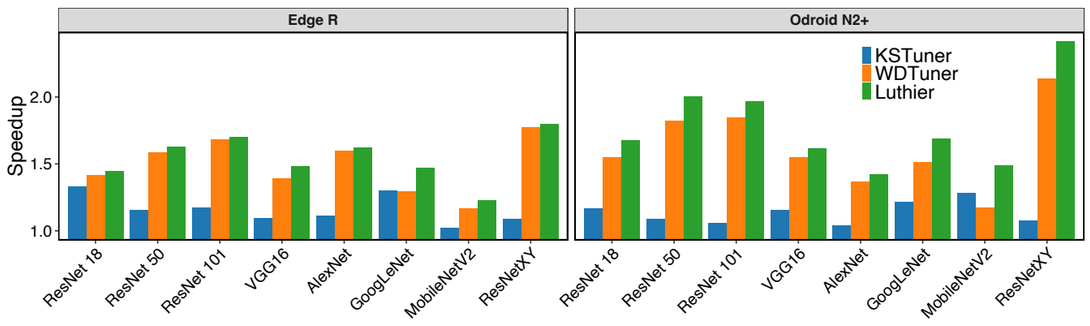

Figure 5. Performance impact of different tuning configurations

Key Results

- **Vision Models:** Up to 2.4x speedup on ResNetXY, 2.0x on ResNet50/101
- **GEMM Operations:** 50 GFLOPS vs ArmNN's 23 GFLOPS (2.2x improvement)
- **Transformer Models:** 1.8x speedup for BERT-base and GPT2-small
- **Tuning Efficiency:** 95% time reduction (1.7h vs 39.2h for AutoTVM)
- **Platform Coverage:** CPU, GPU, and 3-cluster architectures (SD865)
- **GPU Performance:** Up to 2.8x speedup on Mali-G52 GPU
- **Workload Distribution:** More impactful than kernel selection alone

Conclusion

Luthier bridges auto-tuning compilers and vendor libraries:

- Combines vendor kernels with automated tuning for asymmetric multicores
- Achieves 2.0x speedup with 95% reduction in tuning time