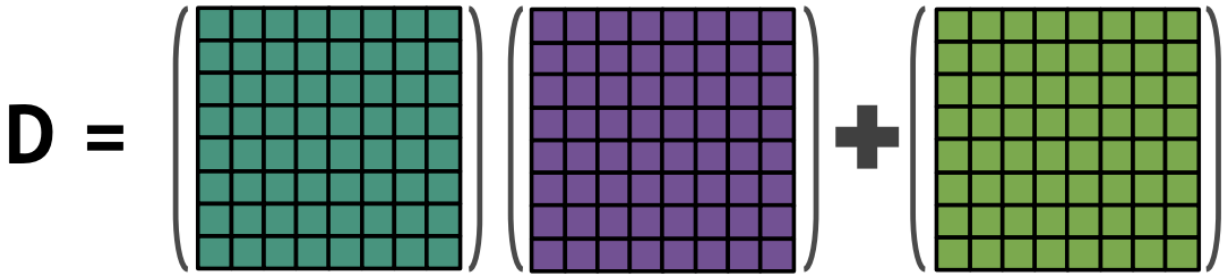
# MixPert: Optimizing Mixed-Precision Floating-Point Emulation on GPU Integer Tensor Cores

Zejia Lin, Aoyuan Sun, Xianwei Zhang, Yutong Lu

SUN YAT-SEN UNIVERSITY

# Tensor-Specialized Architectures (TCUs)

- **Architectures to accelerate matrix multiply-accumulate (MMA)**
  - NVIDIA *Tensor Core*
  - AMD *Matrix Core*
  - **...**

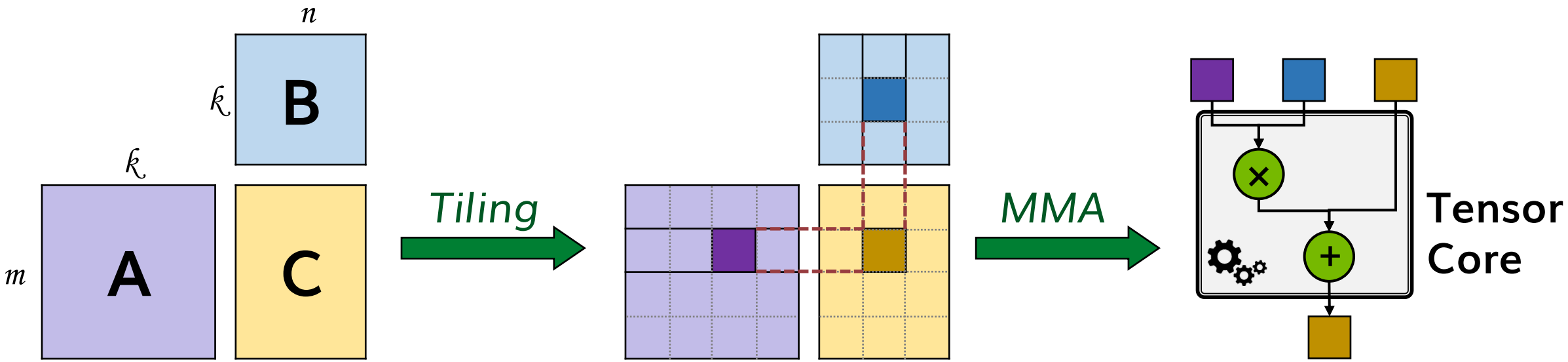


# Tensor-Specialized Architectures (TCUs)

- **Architectures to accelerate matrix multiply-accumulate (MMA)**
  - NVIDIA *Tensor Core*
  - AMD *Matrix Core*
  - ...



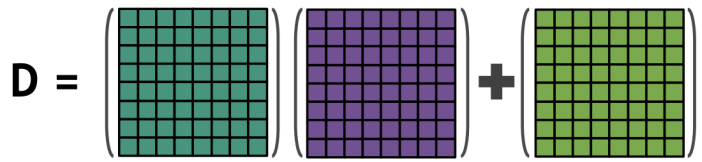

- **MMA as hardware primitives**




SYSU

# TCU Trades Precision For Performance

- **Compared to general-purpose CUDA Cores**
  - Limited data type supports
  - Performant low-precision computation




**Unsupported**

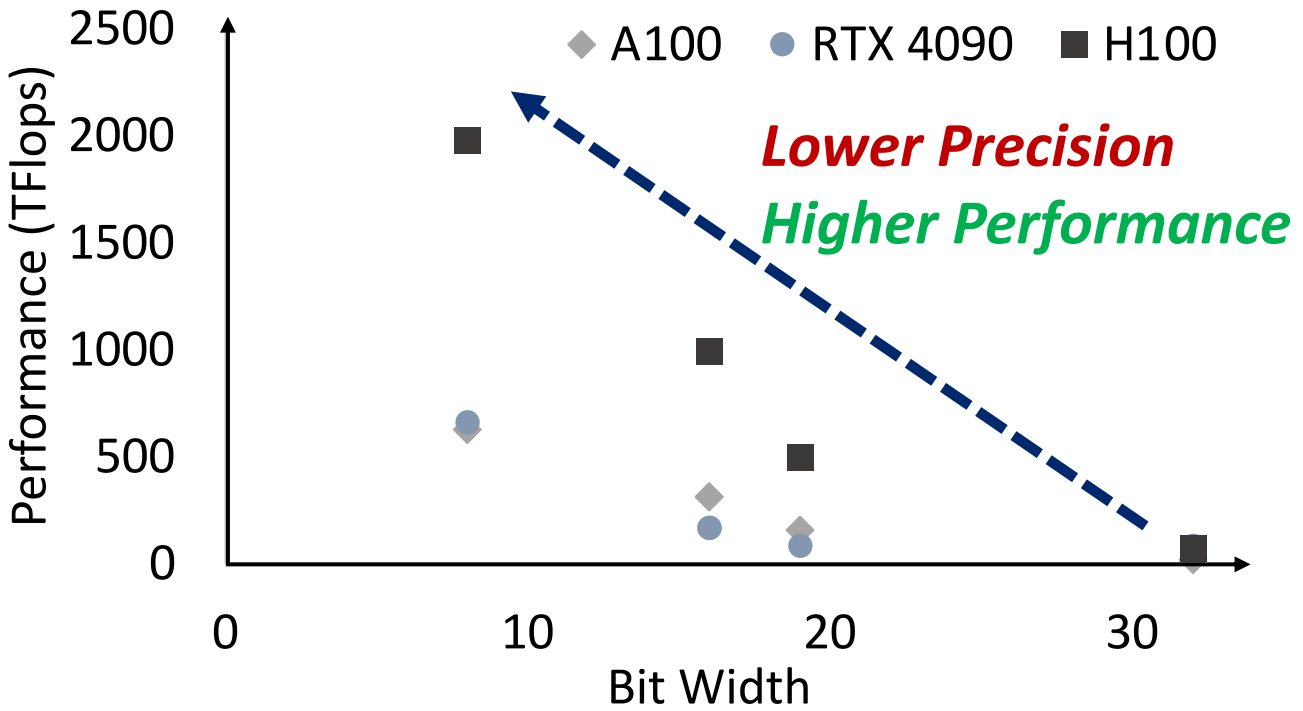
✗ **FP32 × FP32 + FP32**

✓ TF32 × TF32

✓ FP16 × FP16

✓ BF16 × BF16

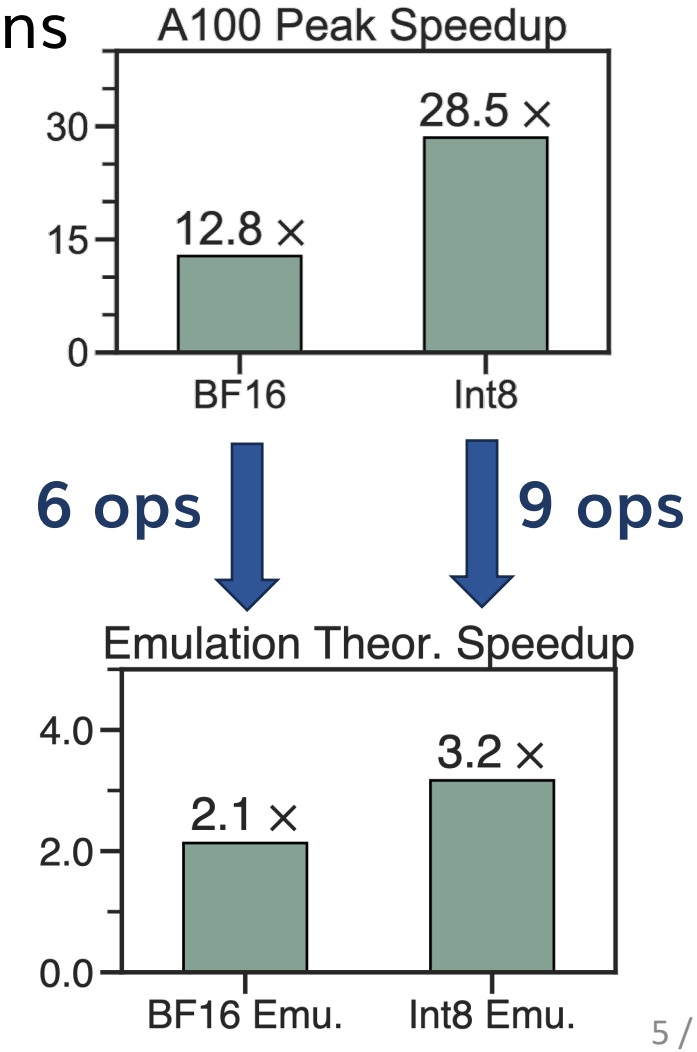
✓ INT8 × INT8 + INT32




SYSU

# Outline

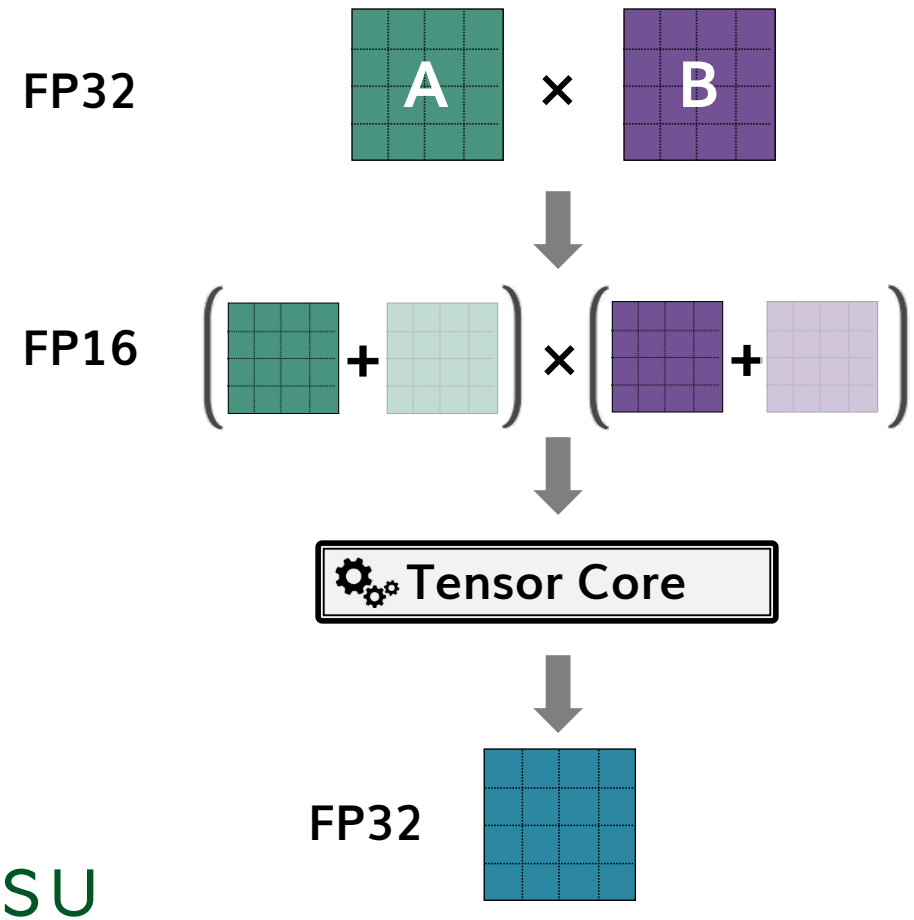
- **To Accelerate FP32 on TCU**
  - Emulation using multiple low-precision operations
- **Challenges For Emulation**
  - Representation & calculation
  - Precision degradation
  - Performance tradeoffs
- **MixPert: Emulation On Integer TCU**
- **Evaluation**

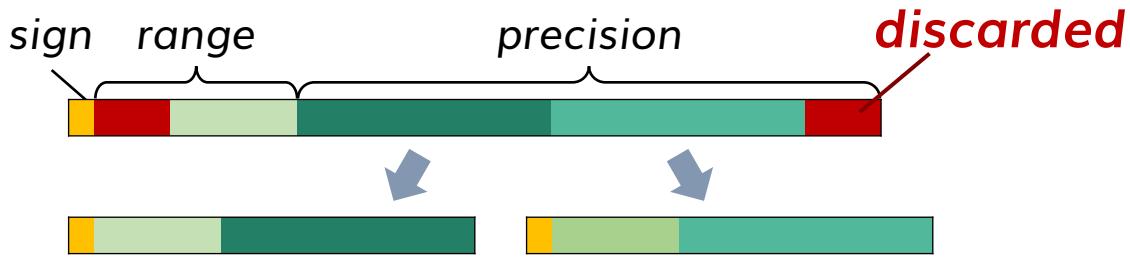



SYSU

# Emulation Degrades Precision

- **Prior works use half-precision for emulation**
  - Example: accelerate FP32 using **3× FP16** multiplications




❶ Scale & Quantization

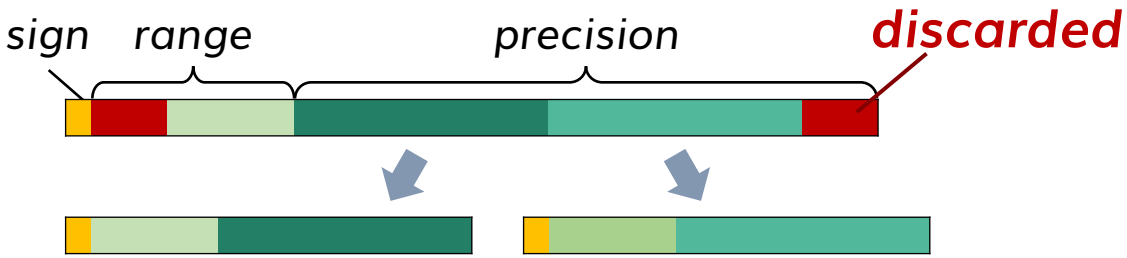
❷ Mixed-precision MMA

❸ Unscale & De-quantization

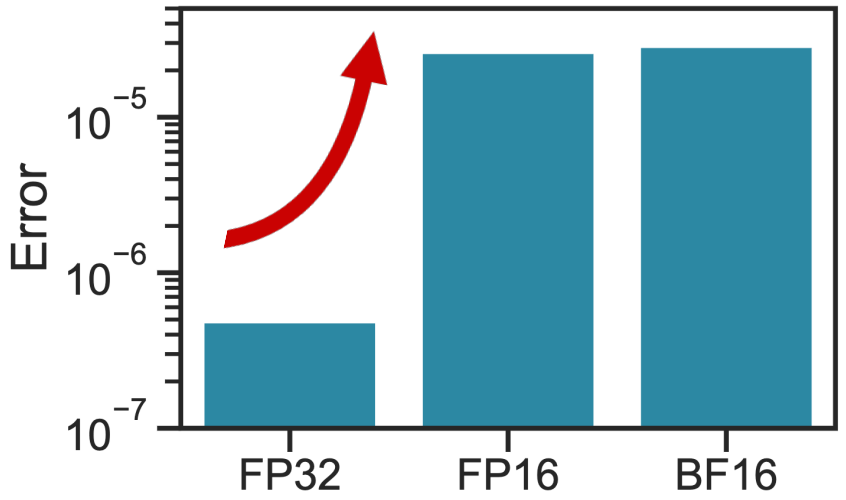
SYSU

# Emulation Degrades Precision (cont.)

- **Quantization rounding introduces error**
  - 3× FP16 *(IPDPSW'18)*
  - 3× TF32 *(PPoPP'21)*
  - 6× BF16 *(ICS'22)*




***540× Precision Degradation***

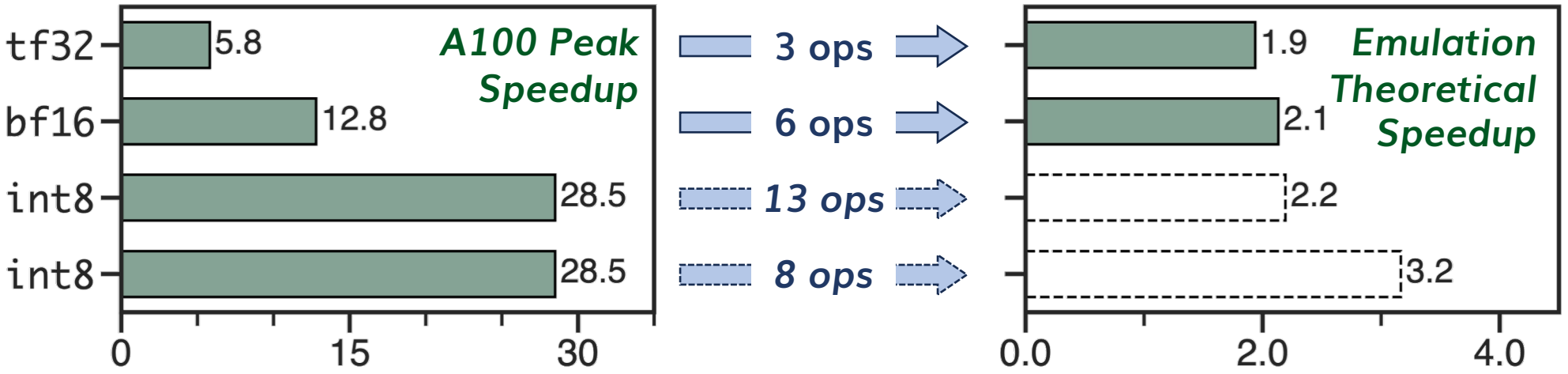



1. Scale & Quantization
2. Mixed-precision MMA
3. Unscale & De-quantization

SYSU

# Exploring INT8's Potential For Emulation

- **Emulation speedup depends on TCU's performance**
  - INT8 is 28.5× faster than FP32, and 2.2× faster than BF16



# Exploring INT8's Potential For Emulation

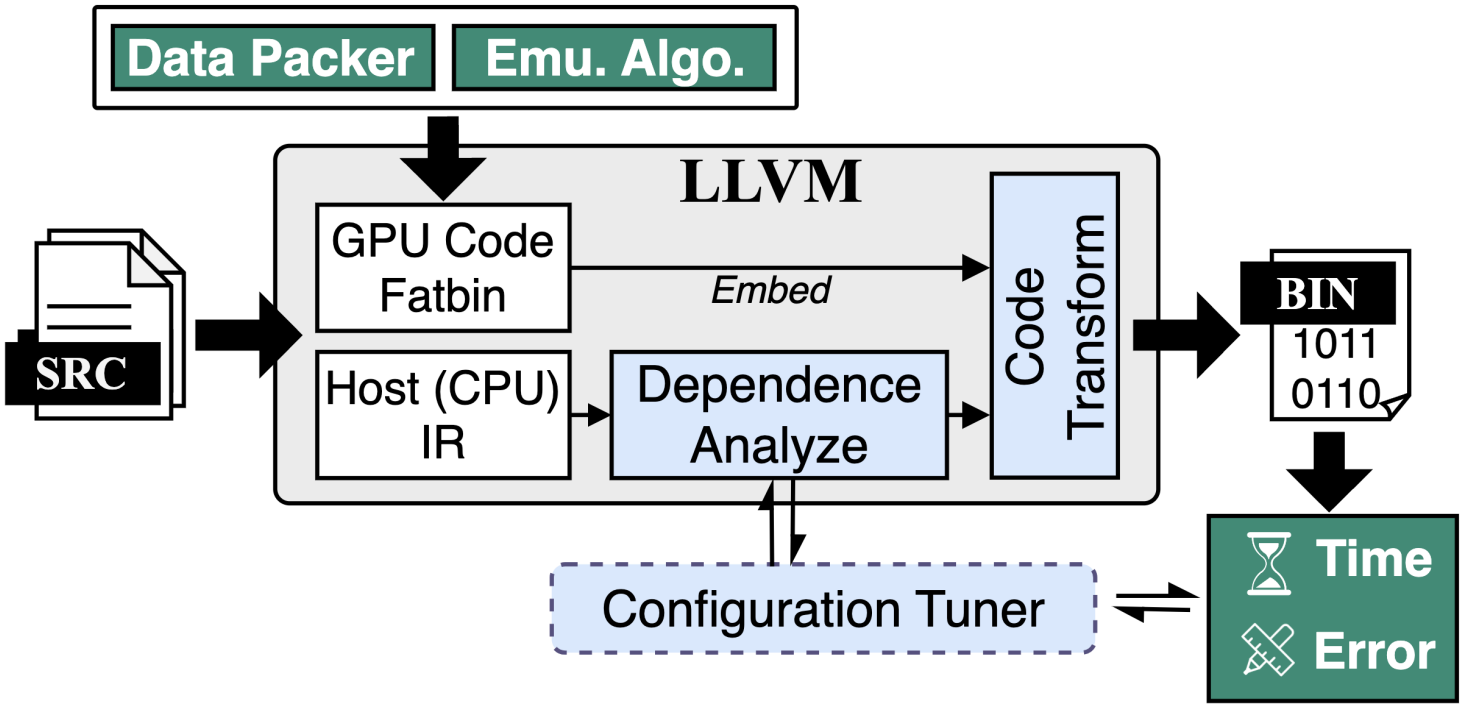
- **Emulation speedup depends on TCU's performance**
  - INT8 is 28.5× faster than FP32, and 2.2× faster than BF16




- **Can we use INT8 for emulation? How to:**
  - Represent the values
  - Emulate MMA on TCU
  - Balance error and speed

SYSU

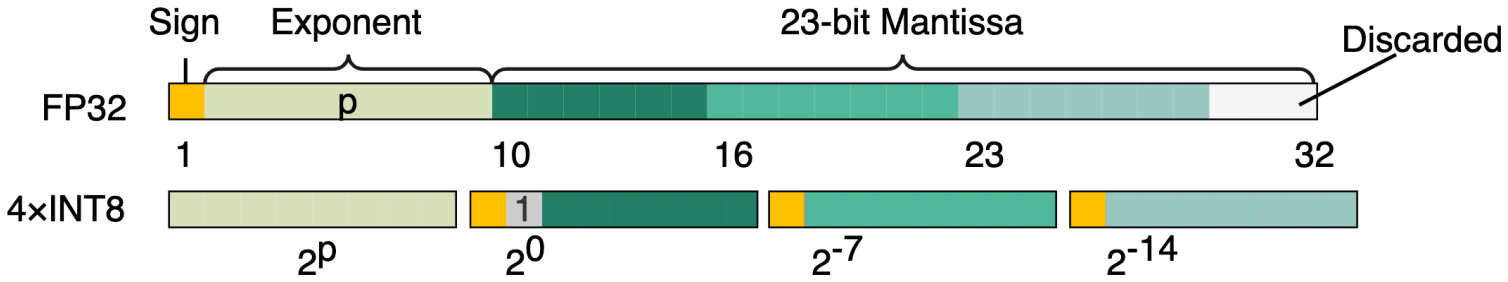
# MixPert – Emulation On Integer Tensor Core



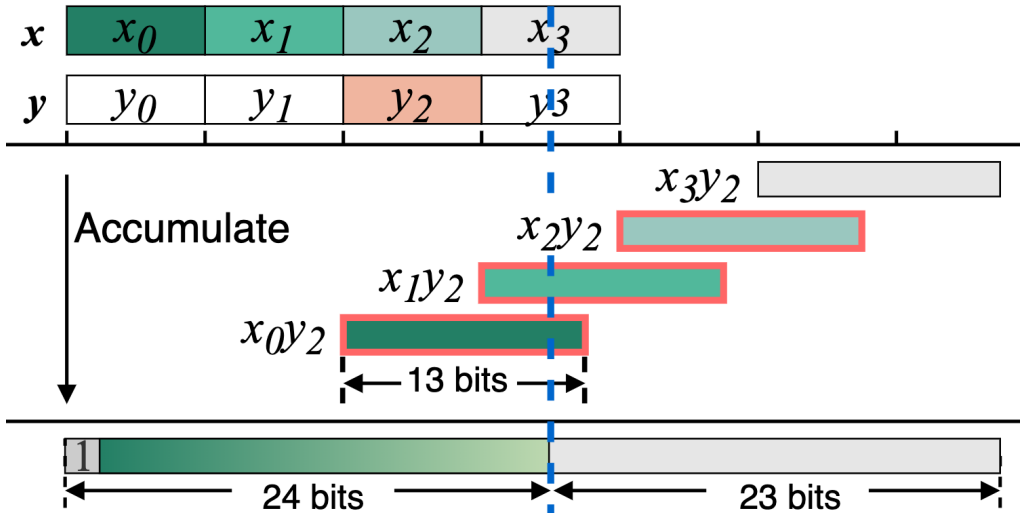

- **MixPert uses INT8 for emulation:**
  - Represent the values ⟹ Pack into 3×INT8 with shared exponents
  - Emulate MMA on TCU ⟹ 6-9 emulation steps
  - Balance error and speed ⟹ Tuning #steps for given error thresholds

# MixPert – Representing the Values

- **Store mantissa bits into three INT8s**
  - Preserving 20 out of 23 bits



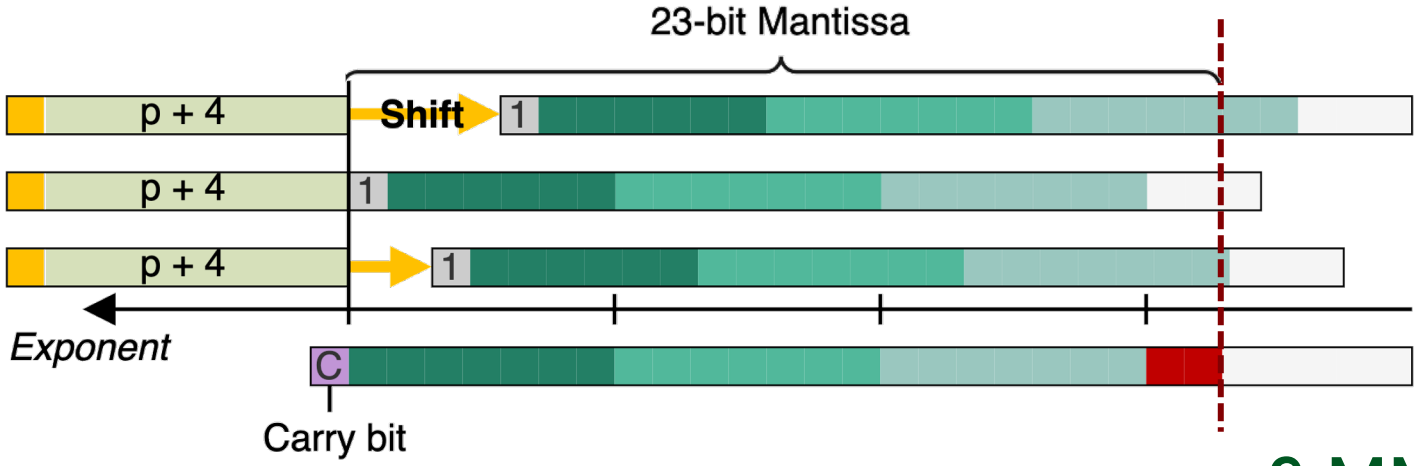

- **Multiplying two scalars with nine INT8 multiplications**




SYSU

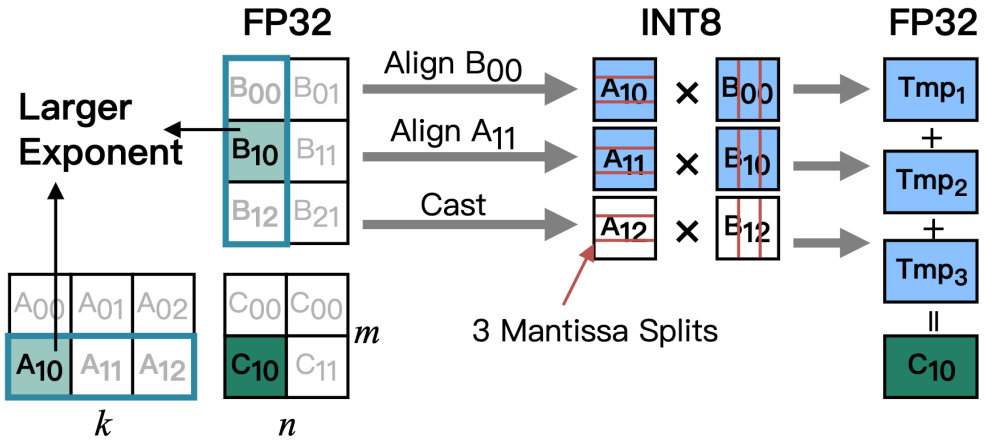
# MixPert – Emulating MMA

- **Share exponent bits when values have similar exponents**
  - Negligible error if data range is smaller than number of elements



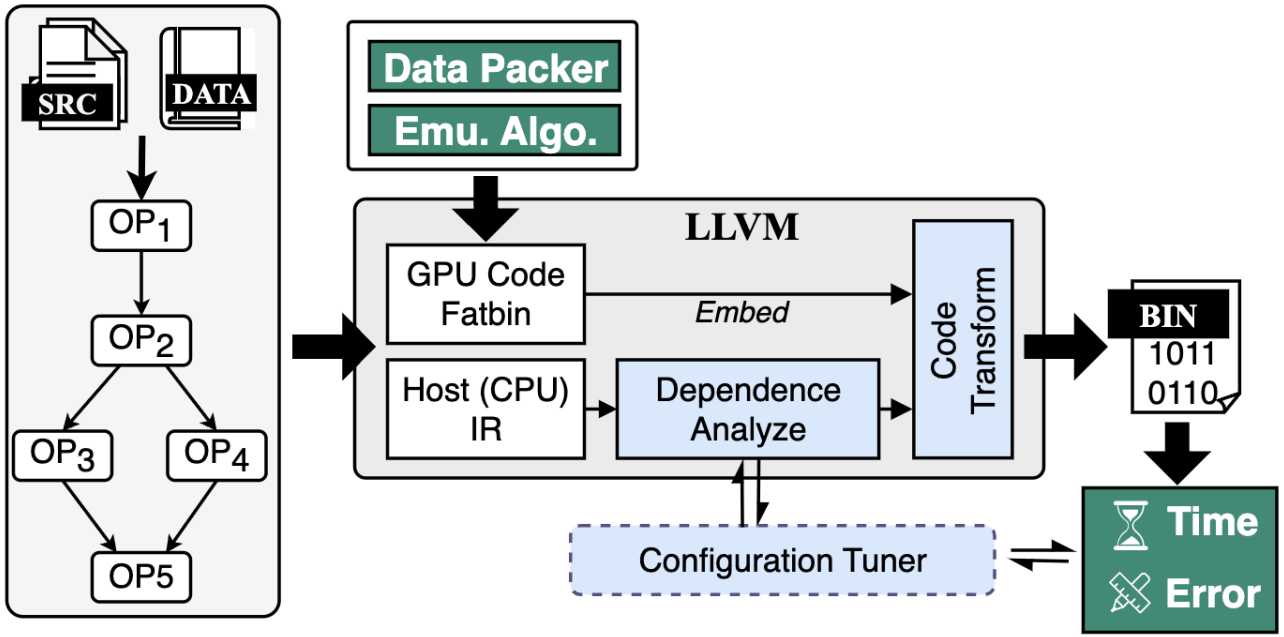

- **The shared exponent is specific to each tile**
  - Restricts error loss to each tile

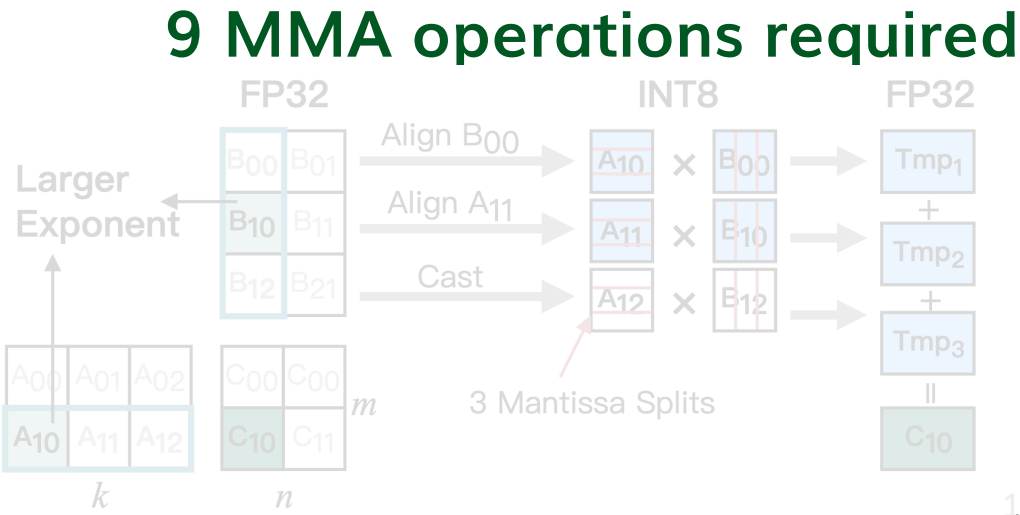
**9 MMA operations required**



# MixPert – Balancing Error And Performance

- **Each operator can have different number of emulation steps**
  - Error-tolerate applications
  - Using 6-9 MMAs to speedup



# Methodology

- **Platforms**
  - NVIDIA A100 & RTX3090
  - CUDA 11.8.0
  - CUTLASS 3.2.1

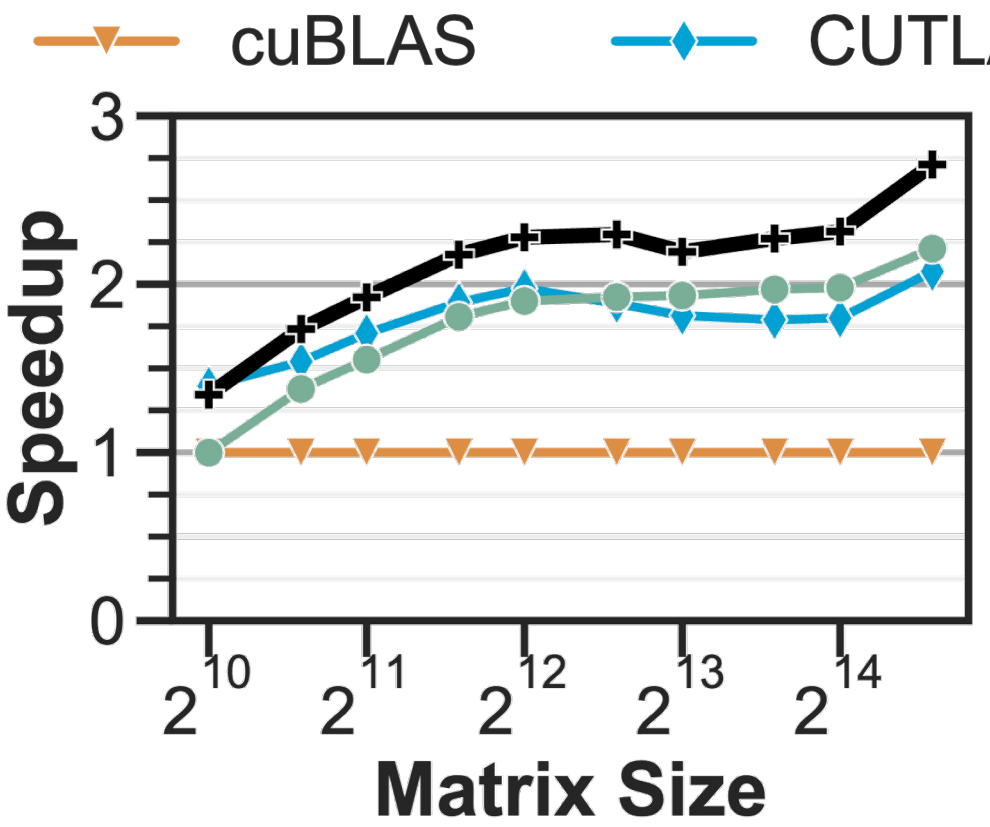
- **Methods**
  - cuBLAS FP32
  - CUTLASS TF32 ×3
  - APE *[ICS'22]* BF16 ×6
  - **MixPert** INT8 ×6-8

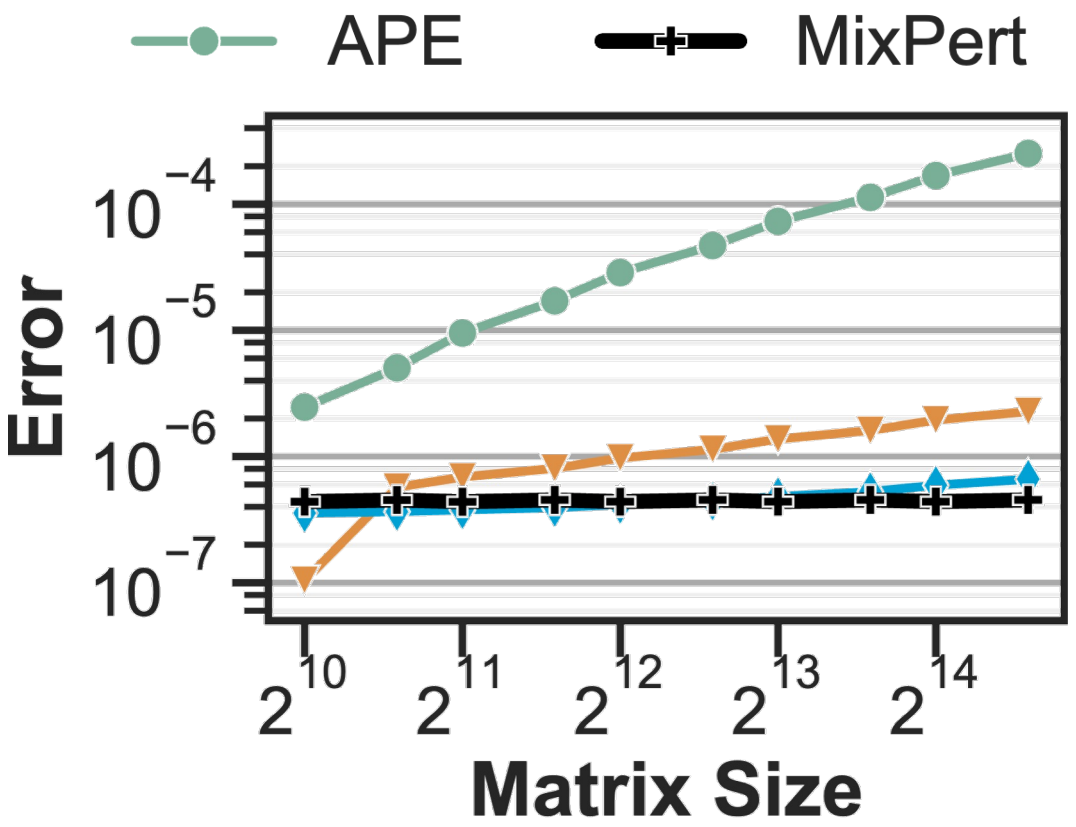
- **Workloads**
  - 8 applications from *Rodinia [IISWC'09]*, APE [ICS'22], and *micro-benchmark*

| Domain | Application |
|---|---|
| Linear algebra | HPL-AI |
| | Cholesky Factorization |
| Genomics | Sparkler |
| Machine Learning | cuBERT |
| | kNN |
| | kMeans |
| Micro-bench | GEMM |
| | MLP |

# Matrix Multiplication Performance & Error

- Average speedup **2.1×** on A100, **1.2×** on RTX 3090
- Average error **$4.46\times10^{-7}$**

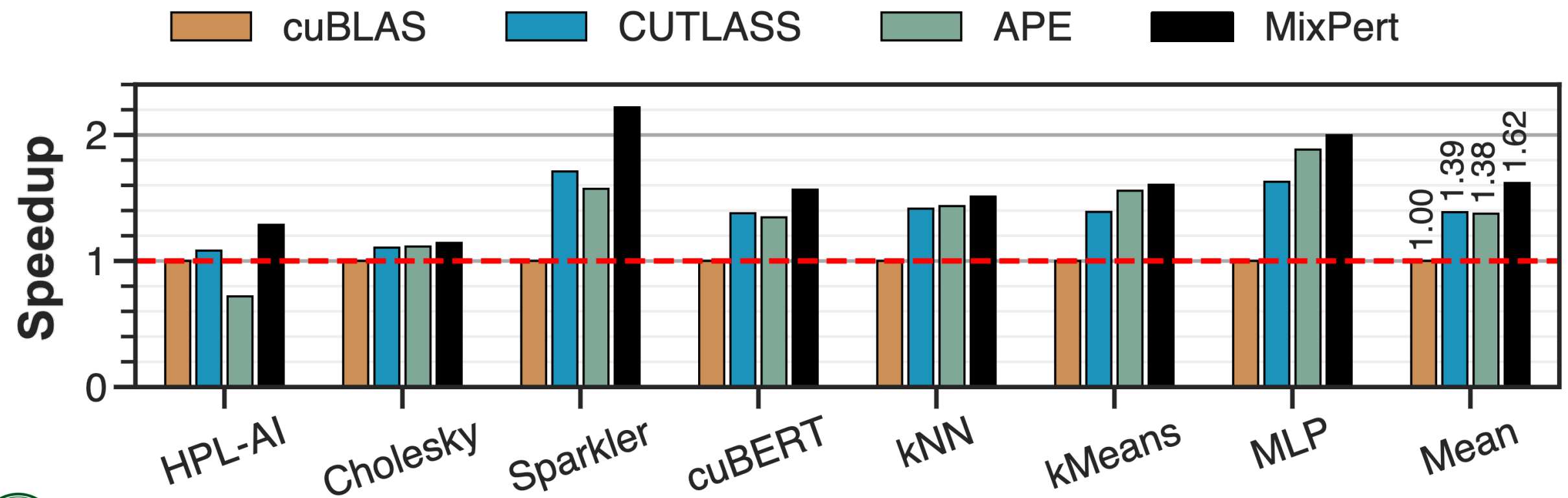



SYSU

# Application Performance

- Average speedup **1.6×**, up to **2.2×**
- HPL-AI and Sparkler uses 6 emulation steps for higher speedup




SYSU

# Conclusion

- **Emulating FP32 on half-precision Tensor Core**
  - Imbalanced performance-precision tradeoffs

- **MixPert: Emulation On Integer Tensor Core**
  - Efficient data representation
  - Tunable emulation steps

- **1.6× to 2.1× computation speedup with controlled error**

## Thank You!

中山大學
SUN YAT-SEN UNIVERSITY

Zejia Lin, Aoyuan Sun, Xianwei Zhang, Yutong Lu