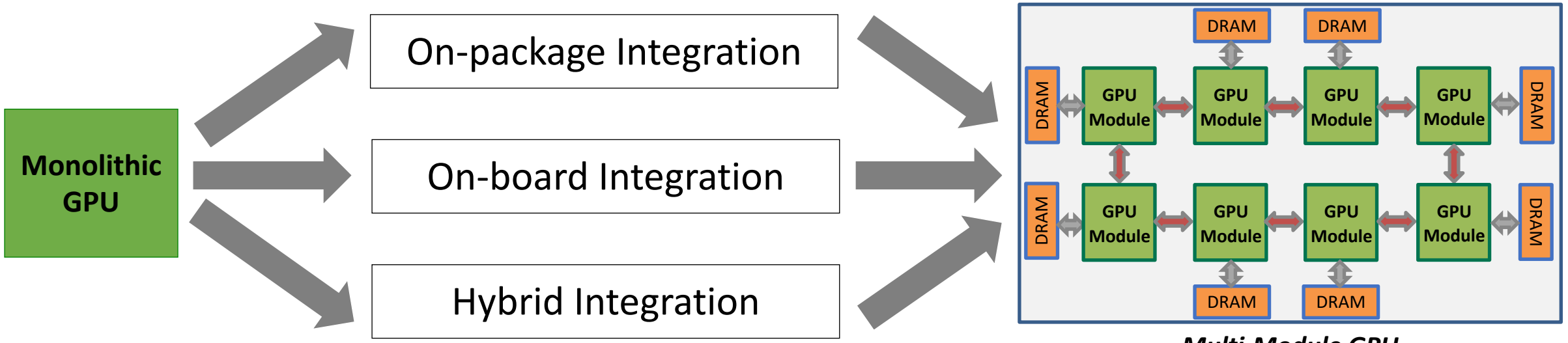
25th IEEE International Symposium on High-Performance Computer Architecture

Understanding the Future of Energy Efficiency in Multi-Module GPUs

Akhil Arunkumar*, Evgeny Bolotin#, David Nellans#, Carole-Jean Wu*

*Arizona State University, #NVIDIA

Multi-Module GPUs



Multi-Module GPU

On-package Integration

Utilize organic package / interposer

- Arunkumar et al., ISCA '17
- Vijayaraghavan et al., HPCA '17

On-board Integration

Utilize PC board

- Milic et al., ISCA '17
- NVIDIA DGX, HGX

Hybrid Integration

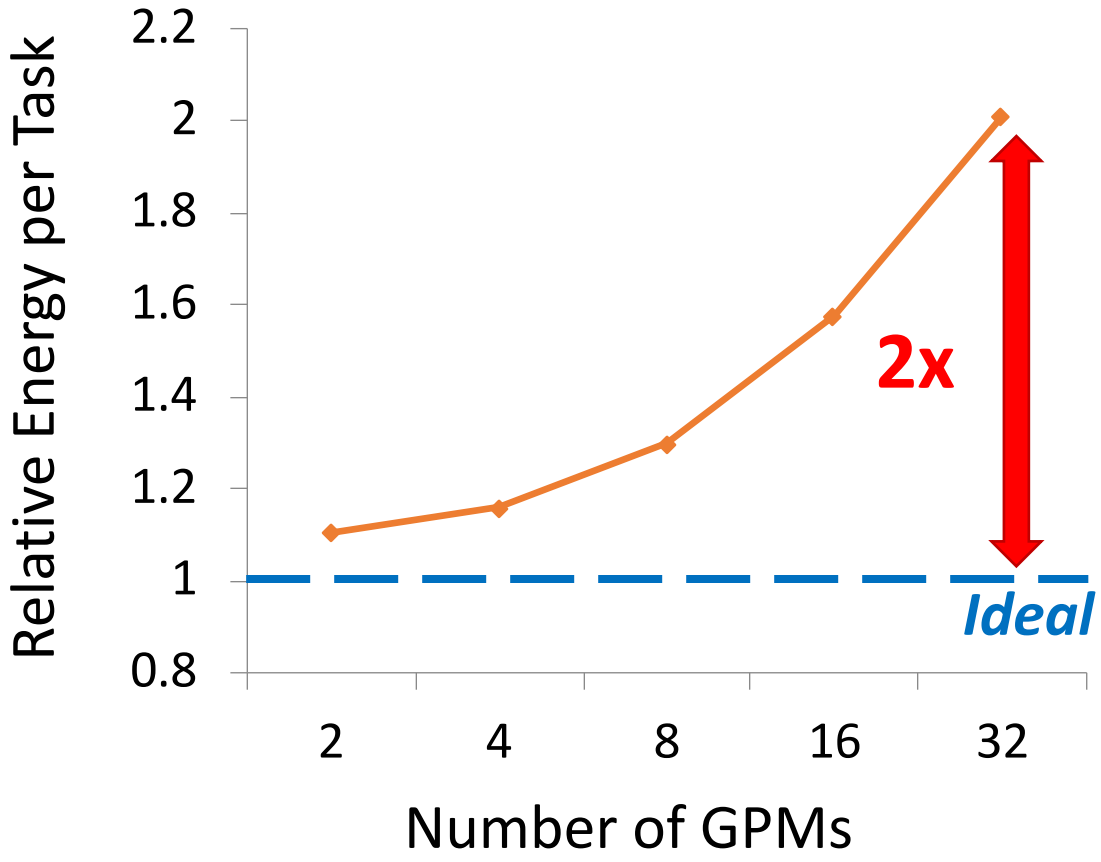
Utilize package and PC board

- Dally et al., VLSI '18

Prior works have focused only on the performance aspect.

Energy Cost of Multi-Module Scaling

- Energy cost per task could double!
  - 32 GPMs integrated on-board consumes *2x* the energy of 1 GPM ☹
- ***What are the energy efficiency limitations?***
- ***Where are the bottlenecks?***



Outline

- Introduction and background
- GPU energy estimation framework – GPUJoule
- Energy efficiency scaling metric – EDPSE
- Energy efficiency trends in future multi-module GPUs
- Conclusion

GPU Energy Estimation – Prior Work

- Bottom-up GPU energy estimation[1][2][3]:
  - Estimate energy cost of each microarchitectural component
  - Hard to keep current as GPUs evolve
- Top-down instruction-based energy estimation[4][5][6]:
  - Estimate energy cost of instruction operations executed
  - Flexible and agile as microarchitecture evolves

Top-down energy model is well suited for GPUs

[1] Hong and Kim, "An integrated GPU power and performance model", ISCA '10
[2] Leng et al., "GPUWattch: Enabling energy optimizations in GPGPUs", ISCA '13
[3] Guerreiro et al., "GPGPU power modeling for multi-domain voltage-frequency scaling", HPCA '18
[4] Kestor et al., "Quantifying the energy cost of data movement in scientific applications", IISWC '13
[5] Pandiyan et al., "Quantifying the energy cost of data movement for Emerging Smartphone Workloads on Mobile Platforms", IISWC '13
[6] Shao et al., "Energy characterization and instruction-level energy model of Intel's Xeon Phi Processor", ISLPED '13

Our Contribution: The GPUJoule Framework

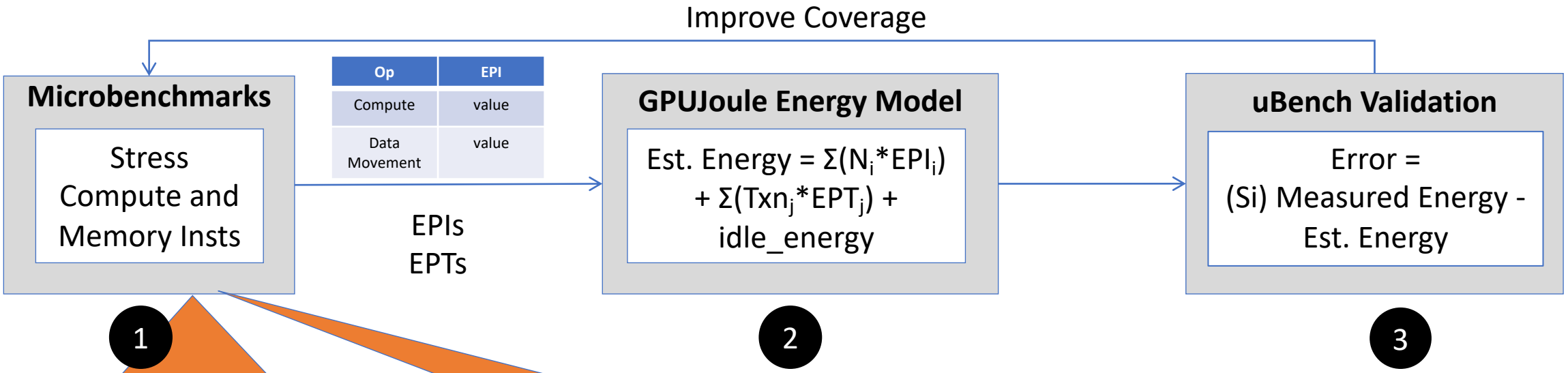
- Key Idea:
  - Estimate the energy-per-instruction (EPI) for each compute instruction type
  - Estimate the energy-per-transaction (EPT) for each memory transaction type
  - GPU-Energy (per-application):

$$= \Sigma(N_i \times EPI_i) + \Sigma(Txn_j \times EPT_j) + idle\_energy$$

Energy to execute compute instructions

Energy to execute data movement instructions

GPUJoule Energy Modeling Methodology



Compute Instruction Microbenchmarks

For *i = 0* to *i < num_iterations* do:

```
__asm_volatile (
    "fma.rn.f32 %r3, %r1, %r3, %r2;"
    …)
```

Memory Instruction Microbenchmarks

```
ptr = (void **)(&array[index])
```

For *i = 0* to *i < num_iterations* do:

```
    ptr = (void**)(*ptr)
```

GPUJoule Validation

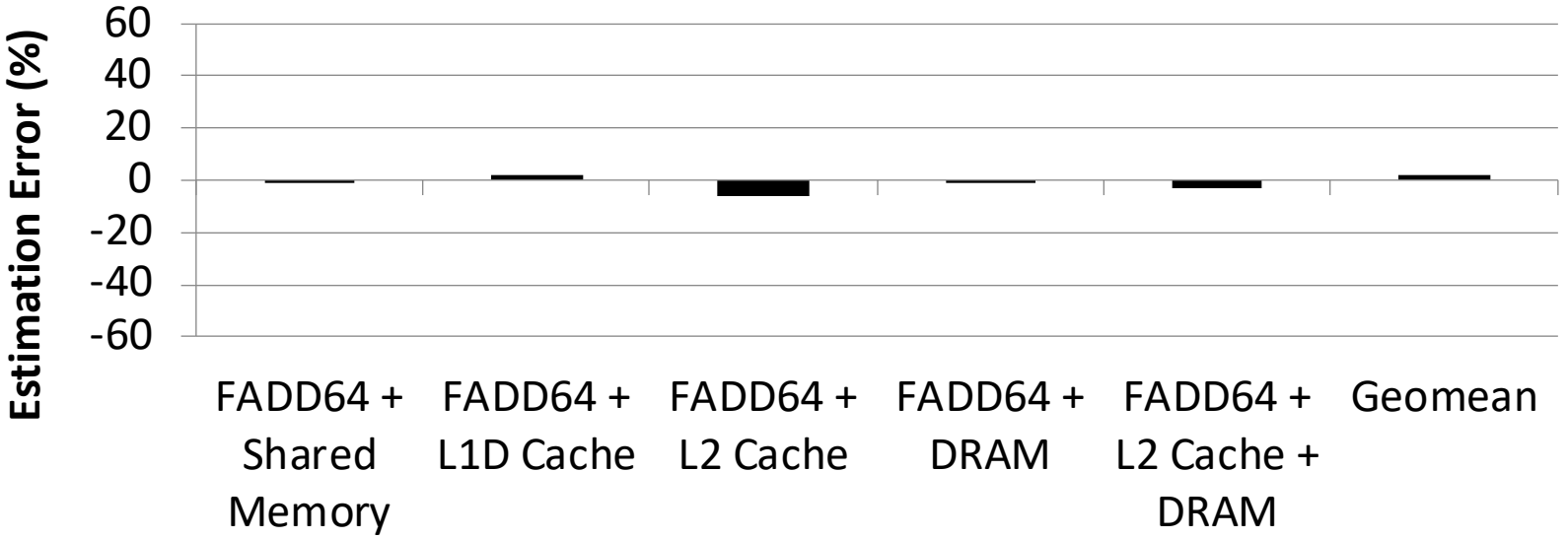
- GPU platform
  - Nvidia Tesla K40 GPU
    - 15 SMs, 16 – 48 KB L1 cache,
    - 1.5 MB L2 cache, 12 GB, 280 GB/s GDDR5 Memory
    - On-board power sensors for power measurement
- Workloads
  - Validation microbenchmarks ➔ compute instruction + data movement operations
  - Real GPU applications from Rodinia, CORAL & Stream suites

Tesla K40 Energy Characteristics

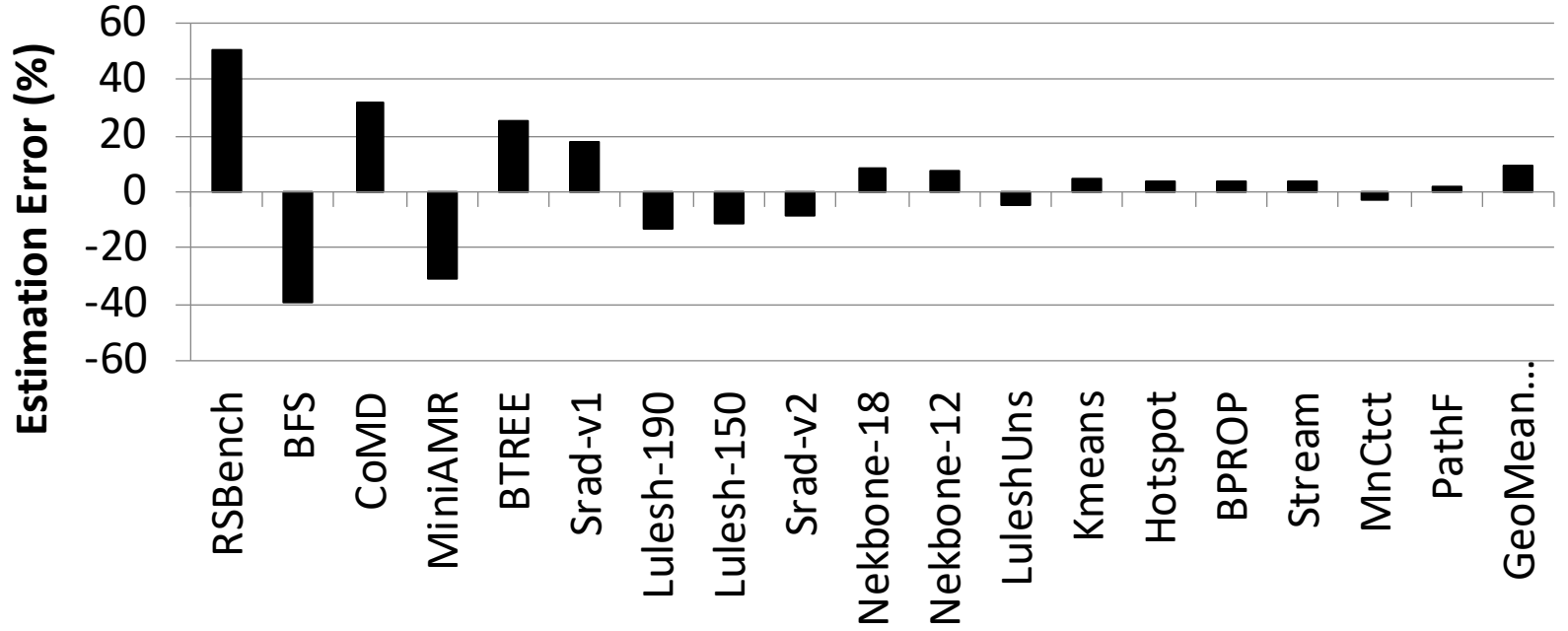
Inst or Op	EPI (nJ)	EPT (pJ/bit)
DADD, FFMA	0.15, 0.05	-
IADD, IMAD	0.07, 0.15	-
LOG2, SINE	0.03, 0.10	-
Shd Mem -> Reg, L1 -> Reg	-	5.32, 5.85
L2 -> L1	-	15.48
DRAM -> L2	-	30.55

- EPI influenced by bit width, and functional unit
- EPT influenced by the level of memory hierarchy
  - DRAM -> Register costs 9x more than L1 -> Register
  - DRAM -> Register costs 80x more than floating point compute

GPUJoule Accuracy



98% Accuracy



90% Accuracy

Outline

- Introduction and background
- GPU energy estimation framework – GPUJoule
- Energy efficiency scaling metric – EDPSE
- Energy efficiency trends in future multi-module GPUs
- Conclusion

Quantifying Energy Efficiency: EDP Scaling Efficiency

- EDP and ED^2 well suited for comparing systems with similar resources
- For strong scaled systems: Energy-Delay-Product Scaling Efficiency (EDPSE)

$$EDPSE = \frac{EDP_1}{N} \times \frac{1}{EDP_N}$$

- Evaluates performance, energy costs, and resource scaling together
- Systems can be expected to achieve an EDPSE threshold in the future
  - 50% EDPSE ➔ "Energy efficiency scales to *50% of the ideal* with strong scaling"

Outline

- Introduction and background
- GPU energy estimation framework – GPUJoule
- Energy efficiency scaling metric – EDPSE
- Energy efficiency trends in future multi-module GPUs
- Conclusion

Methodology

- Performance Simulations:
  - Model GPUs with 1 – 32 GPU modules
  - Distributed CTA scheduling, first touch page placement, ring interconnect

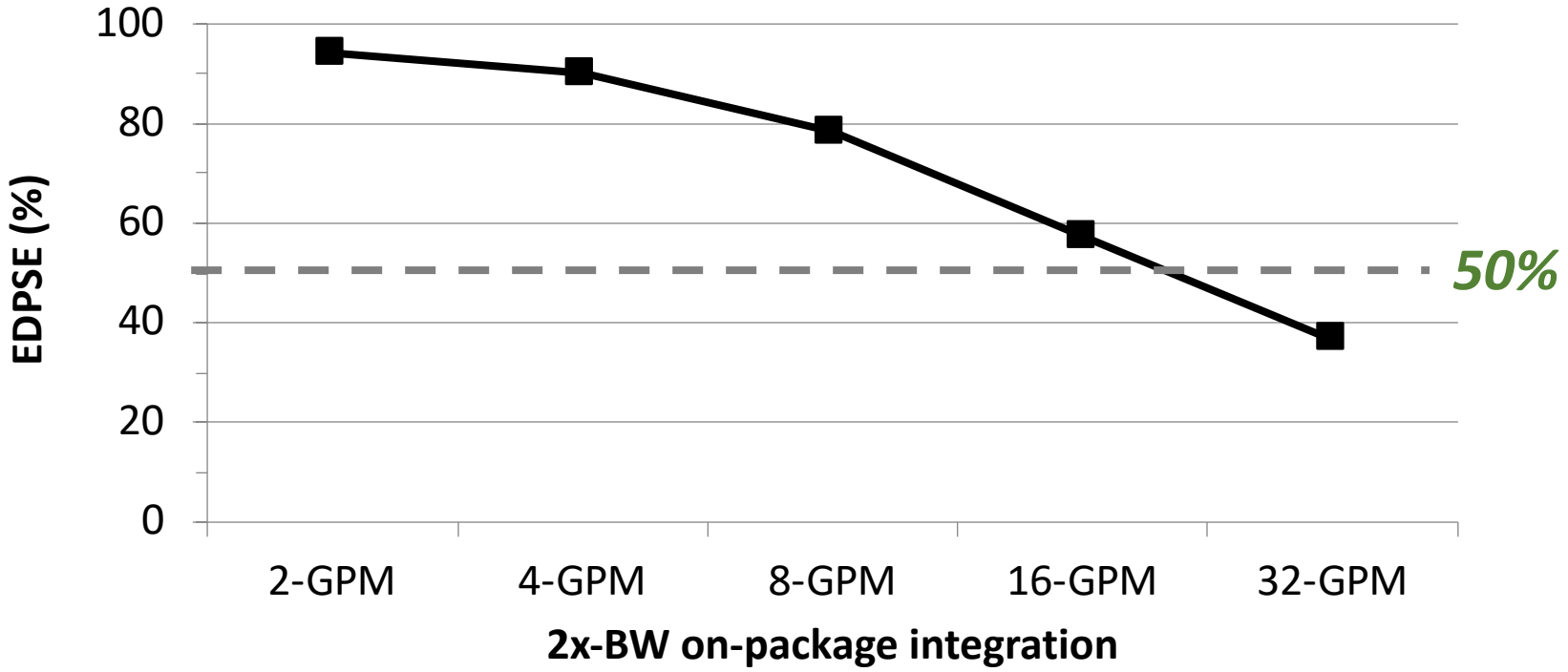
BW Config Name	I/0 BW	DRAM BW	I/O to DRAM BW Ratio	Integration Domain
1x-BW	128 GB/s	256 GB/s	1:2	On-Board
2x-BW	256 GB/s	256 GB/s	1:1	On-Package
4x-BW	512 GB/s	256 GB/s	2:1	On-Package

- Energy Modeling:
  - EPI and EPT values from GPUJoule
  - Augmented with HBM Memory & Inter-GPM data movement energy costs

	Energy Cost
HBM DRAM -> L2 Cache[1]	21.1 pJ/bit
On-Package Inter-GPM[2]	0.54 pJ/bit
On-Board Inter-GPM[3]	10 pJ/bit

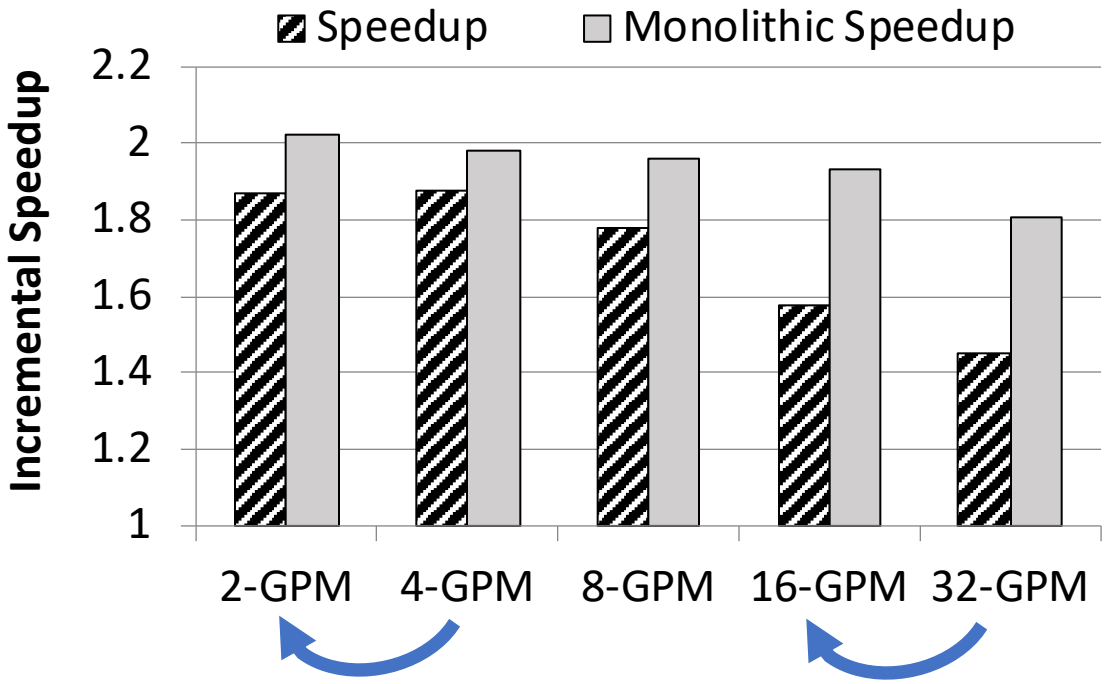
[1] O'Connor et al., "Fine-Grained DRAM: Energy-Efficient DRAM for Extreme Bandwidth Systems", MICRO 2017
[2] Poulton et al., "A 0.54 pJ/b 20 Gb/s Ground-Referenced Single-Ended Short-Reach Serial Link in 28 nm CMOS for Advanced Packaging Applications", JSSC 2013
[3] Dally, W., "Challenges for Future Computing Systems", Keynote, HiPEAC 2015

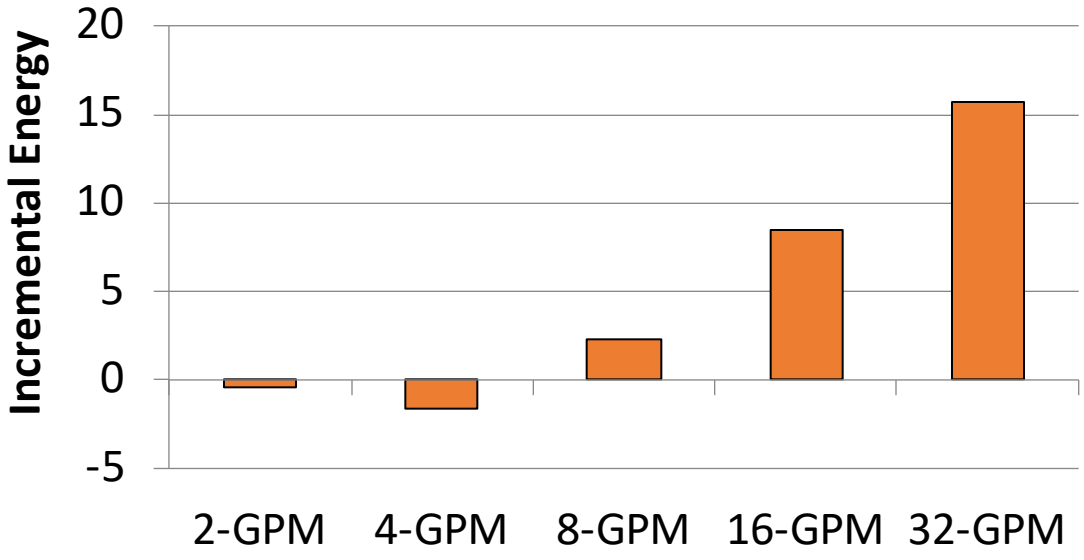
EDP Scaling Efficiency of Future GPUs



- EDPSE reduces drastically with increase in GPMs
- Multi-Module GPUs face energy efficiency limitations at scale

Diminishing Trend in Energy Efficiency Scaling



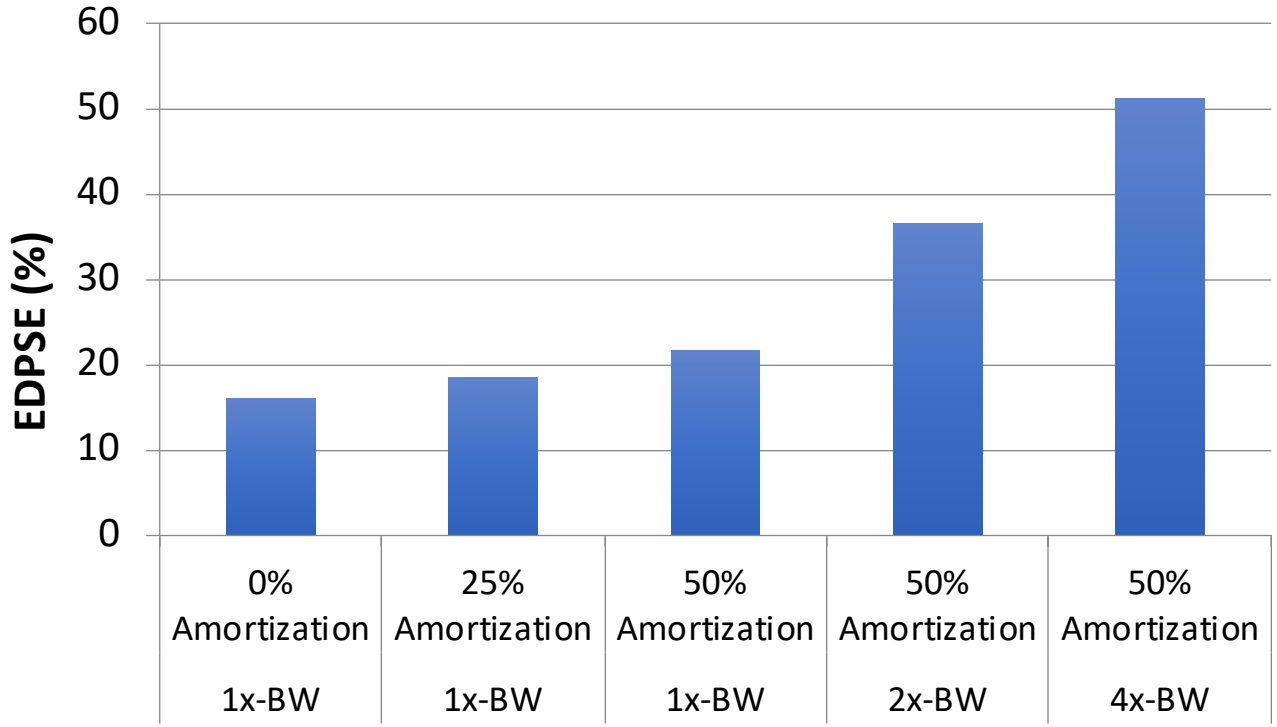


- Speedup reduces as number of modules increase
- Energy cost increases as number of modules increase

NUMA-effects lead to performance loss and energy increase

On-package integration and constant energy amortization

- Multi-module GPUs suffer from high constant energy overheads
  - VRMs, power delivery network, system I/O etc.
- On-package integration allows amortization of these overheads



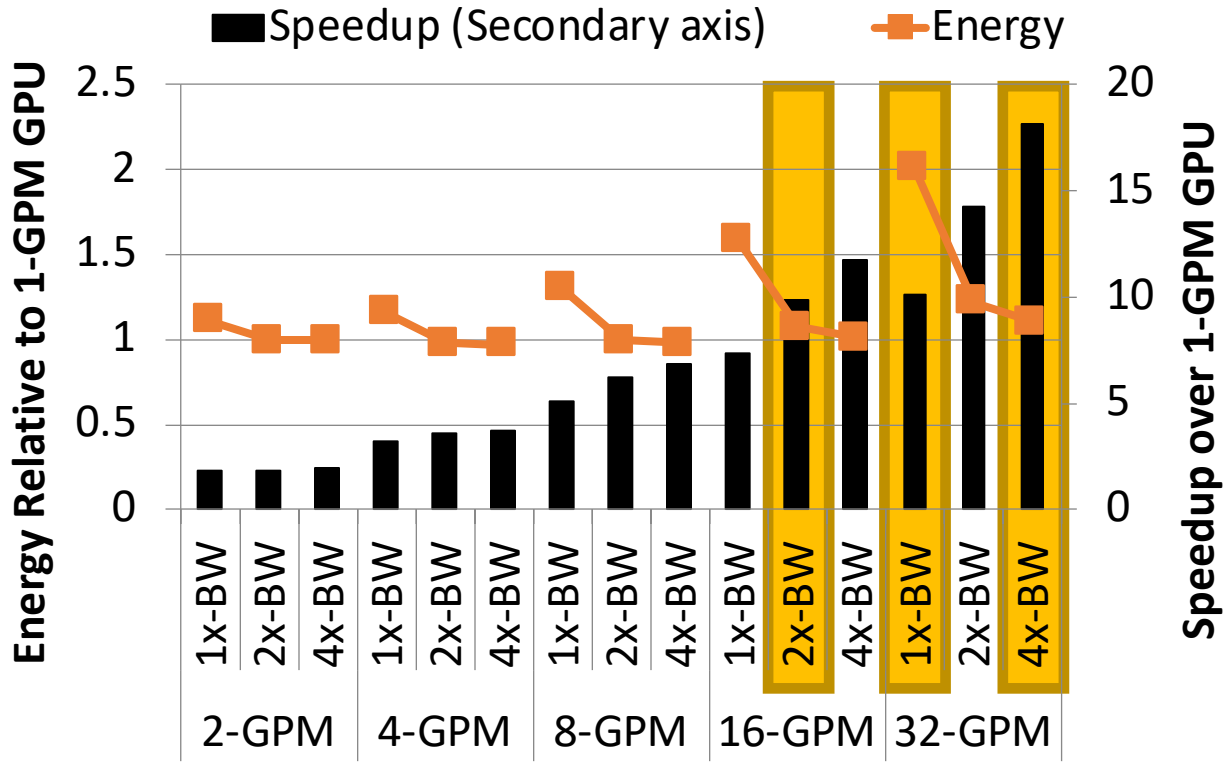
Higher link BW and tighter integration yields better energy efficiency scaling

Speedup & Energy Consumption

- Speedup is dependent on bandwidth
- Energy consumption drops with speedup

- Only increasing GPMs might not help
  - 16-GPM with 2xBW has same performance as 32-GPM with 1xBW
  - Consumes only half the energy!

- Path to an efficient 32-GPM GPU
  - Increase bandwidth to 4x-BW.
  - Utilize on-package integration
  - Reduce energy consumption by 45%



Conclusions

- Developed GPUJoule Instruction level GPU energy estimation framework
  - Achieves 90% accuracy compared to real silicon energy measurements
  - Open sourced at github.com/akhilarunkumar/GPUJoule_release
- Identify key energy efficiency trends in future GPUs
  - Energy efficiency scaling reduces as number of modules increase
  - NUMA effects lead to suboptimal performance and energy consumption
  - Inter-module bandwidth and tighter integration of components (on package integration) lead to higher energy efficiency

25th IEEE International Symposium on High-Performance Computer Architecture

Understanding the Future of Energy Efficiency in Multi-Module GPUs

Thank you

Akhil Arunkumar*, Evgeny Bolotin#, David Nellans#, Carole-Jean Wu*

*Arizona State University, #NVIDIA

Impact of On-Board Switch