Disintegrating Manycores: Which Applications Lose and Why?

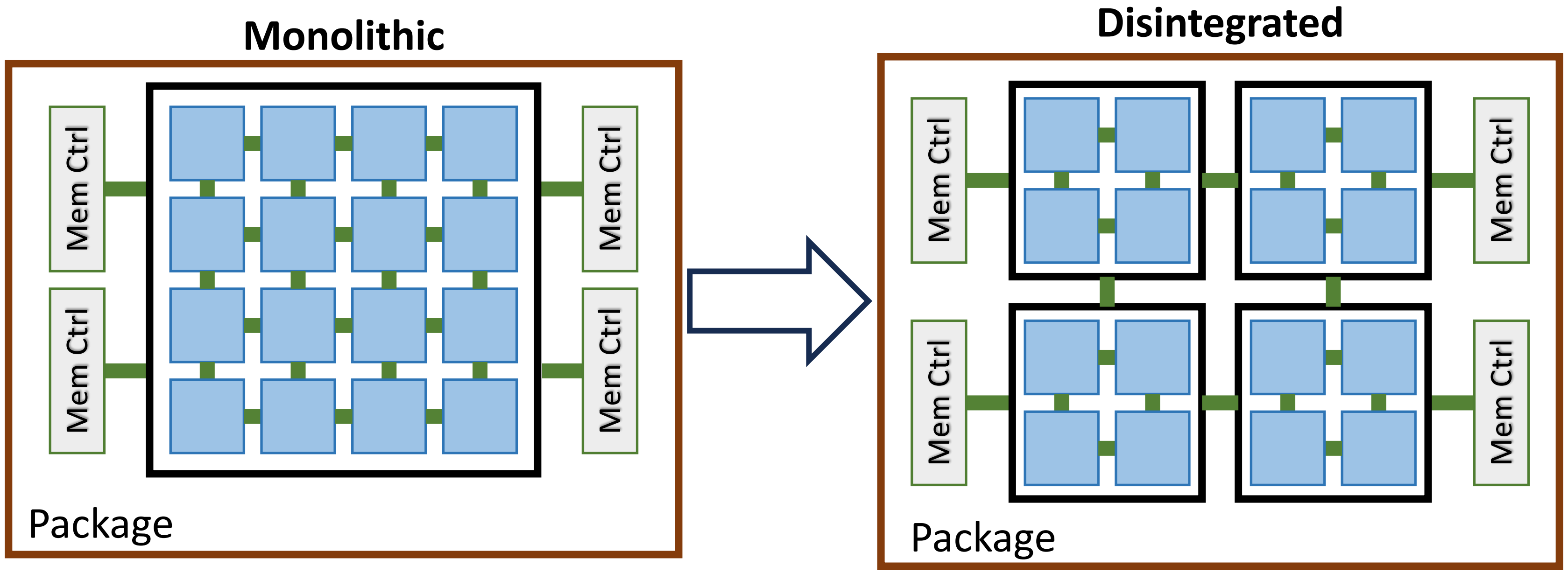
Isidor R. Brkić, Mark C. Jeffrey

28 October 2023

NoCArc 2023

UNIVERSITY OF TORONTO

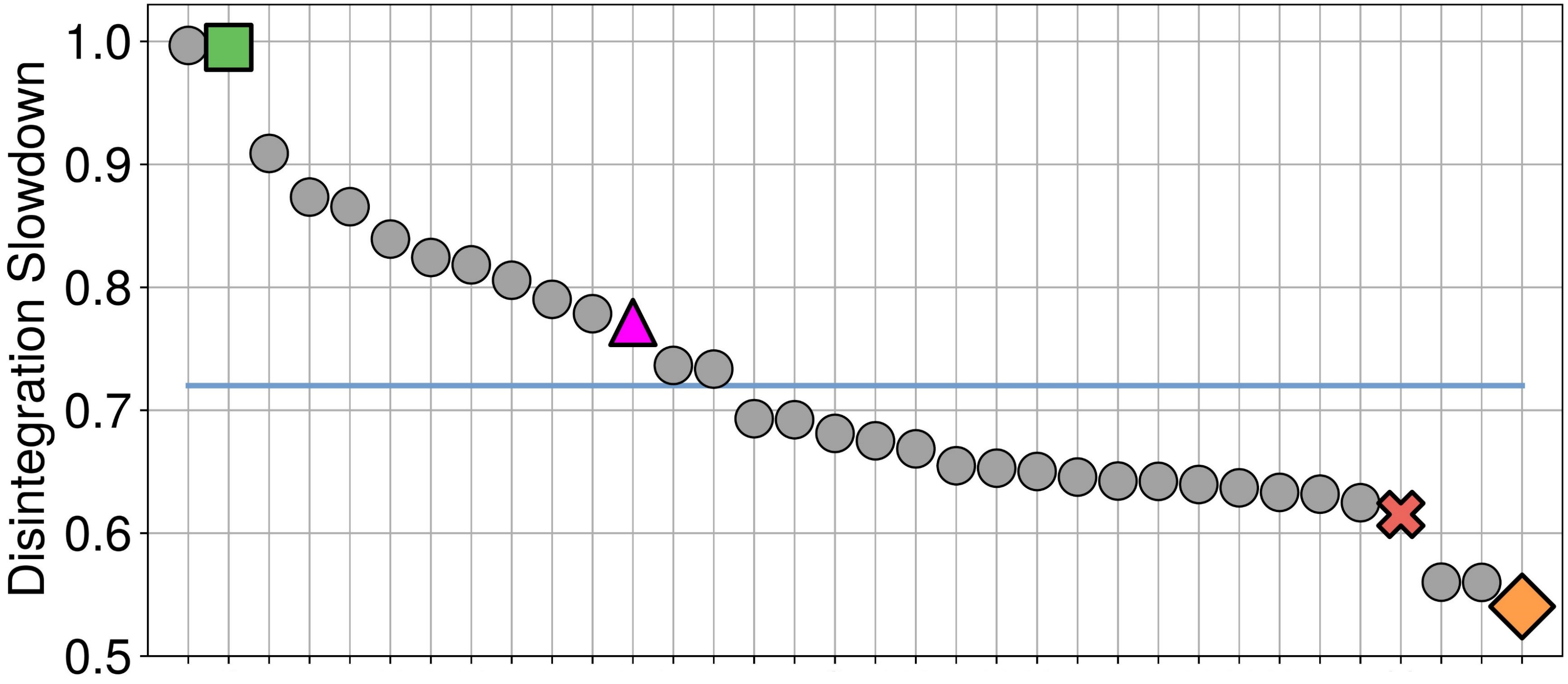
Disintegration Saves Cost ($)



Company	Architecture	Chiplets	"Cores"
Intel	CPU	4	60
AMD	CPU	9-13	64
Apple	CPU/GPU/Accelerator	2	20/64/32
NVIDIA	GPU	4	256
NVIDIA	Accelerator	36	576
Xilinx	FPGA	?	N/A

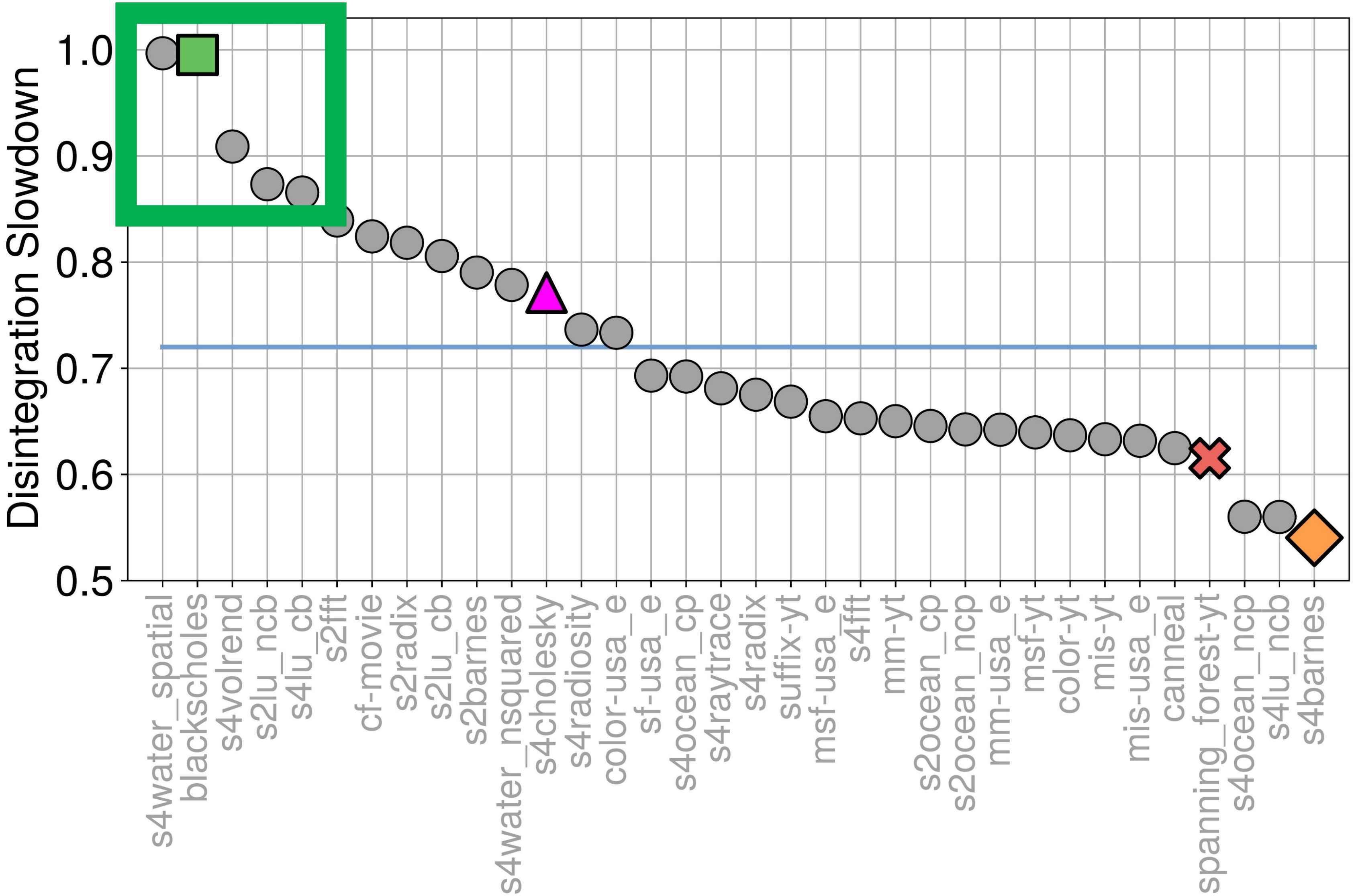
Disintegration Harms Performance

There has been no broad study of how disintegration impacts performance



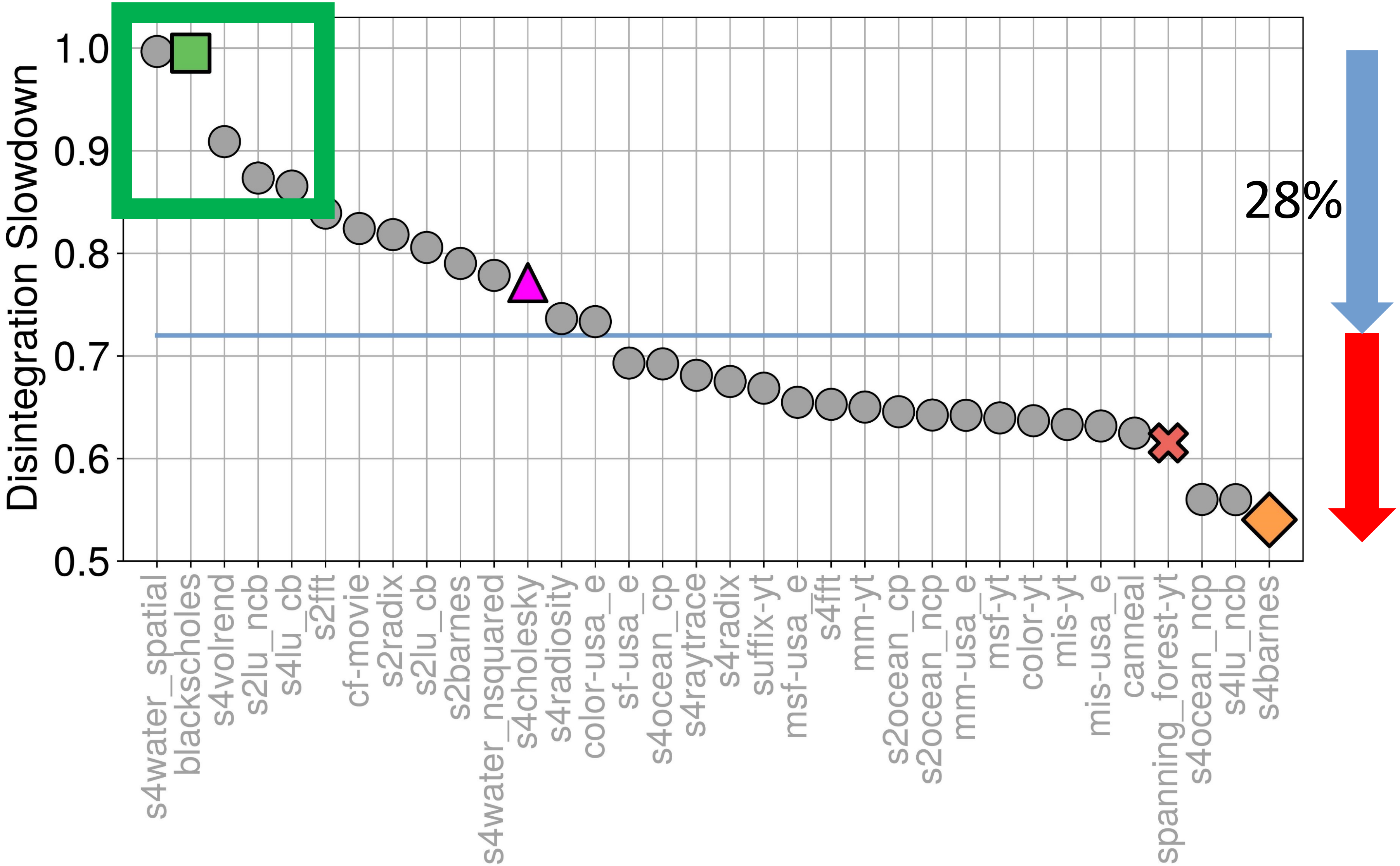
Disintegration Harms Performance

There has been no broad study of how disintegration impacts performance



Disintegration Harms Performance

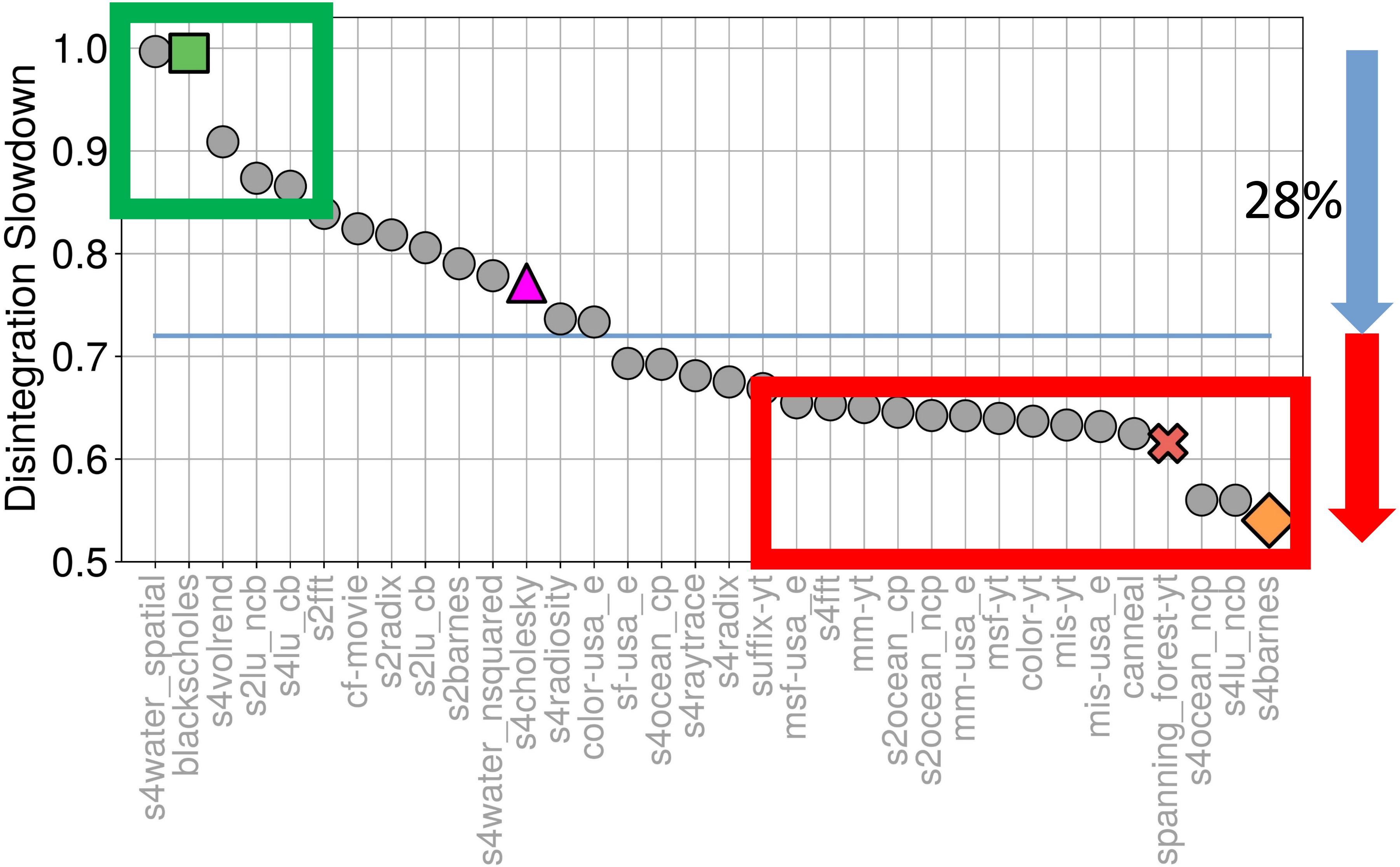
There has been no broad study of how disintegration impacts performance



Applications are impacted differently

Disintegration Harms Performance

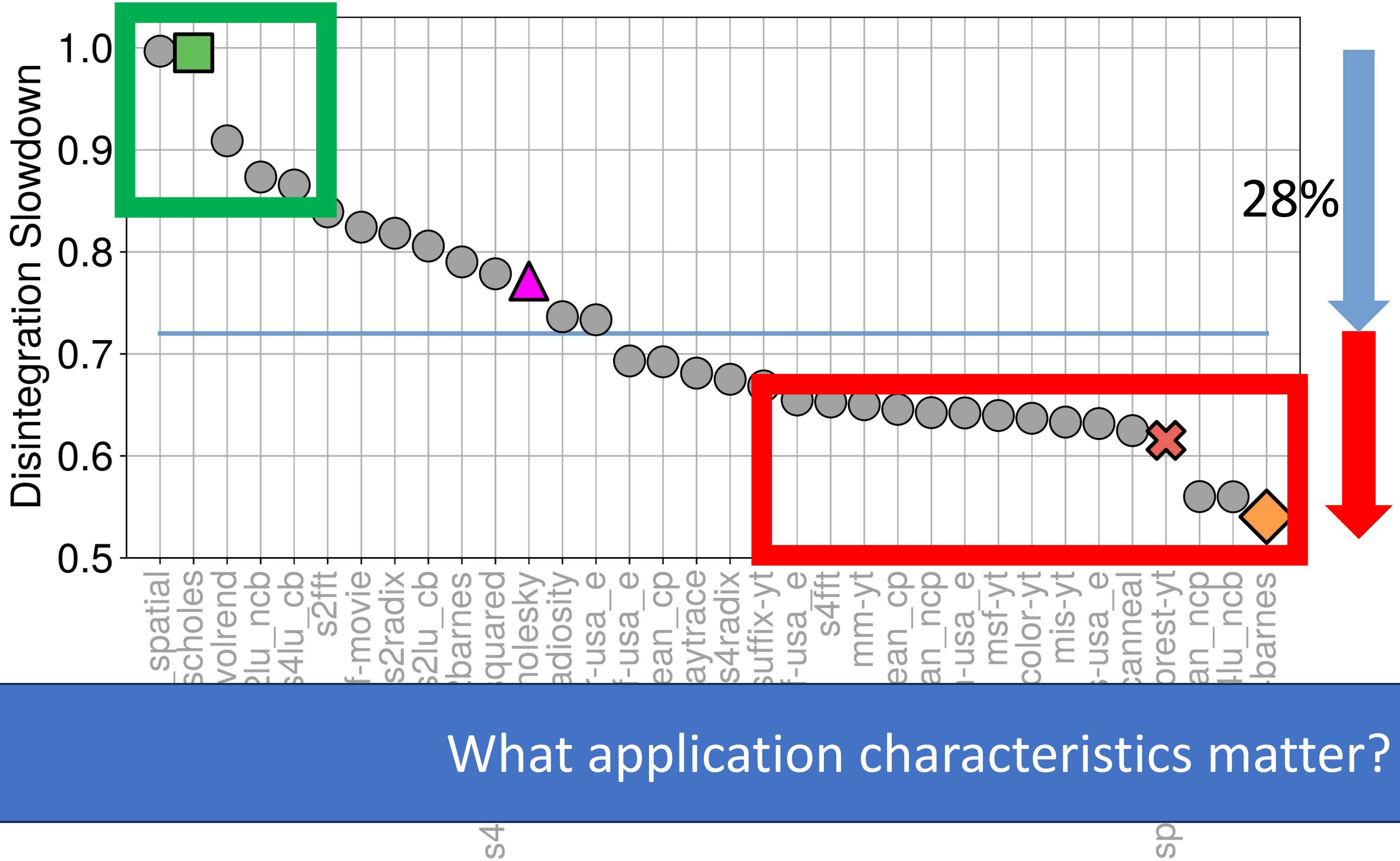
There has been no broad study of how disintegration impacts performance



Applications are impacted differently

Disintegration Harms Performance

There has been no broad study of how disintegration impacts performance



Applications are impacted differently

What application characteristics matter?

Executive Summary

We demonstrate that:

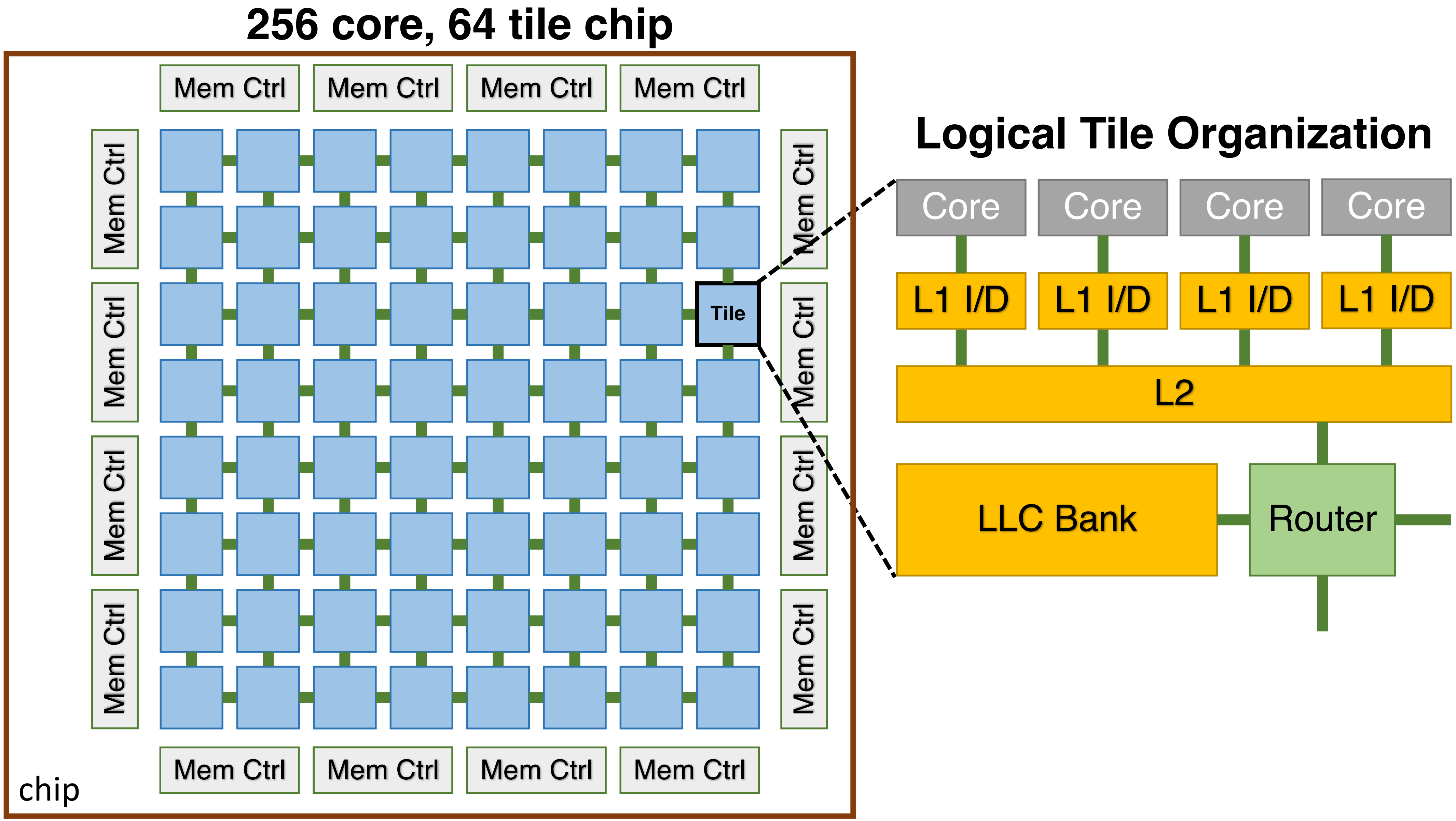
Unsurprisingly, scalable applications still scale on disintegrated systems

Disintegration penalty varies significantly across applications and is not correlated with monolithic speedup

Data sharing and network injection bandwidth lead to larger disintegration penalty

Methodology

Baseline Manycore Monolith



Many Ways to Connect Chiplets

Most Integrated

Highest cost

3D stacked

Active silicon interposer

Passive silicon interposer

Embedded multi-die interconnect bridge (EMIB)

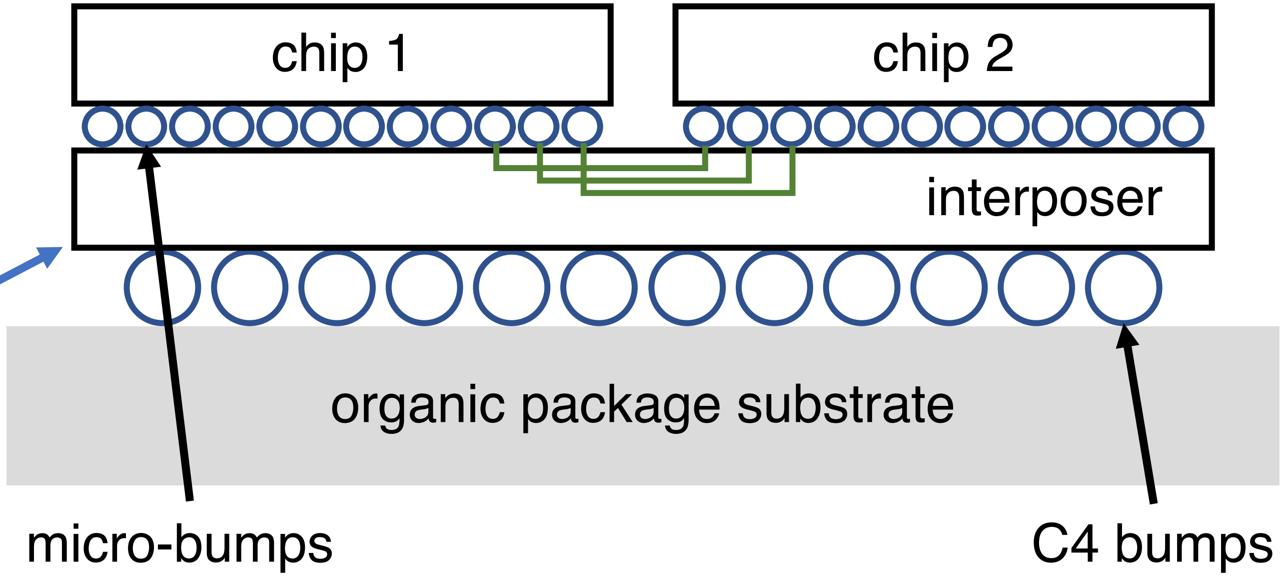
Multi-chip module (MCM)

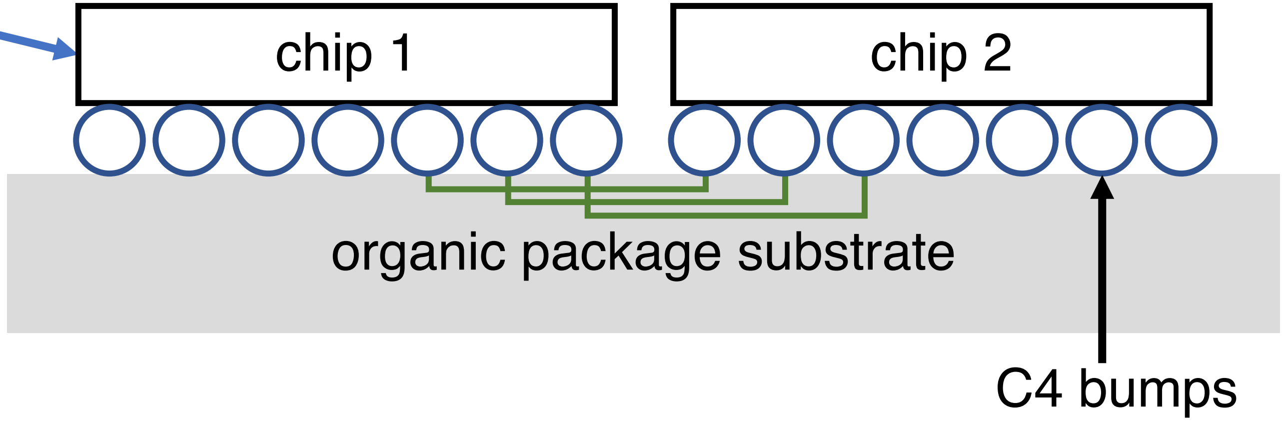
Multi-socket

Multi-board

Least Integrated

Lowest Cost





Many Ways to Connect Chiplets

Most Integrated

Highest cost

3D stacked

Active silicon interposer

Passive silicon interposer

Embedded multi-die interconnect bridge (EMIB)

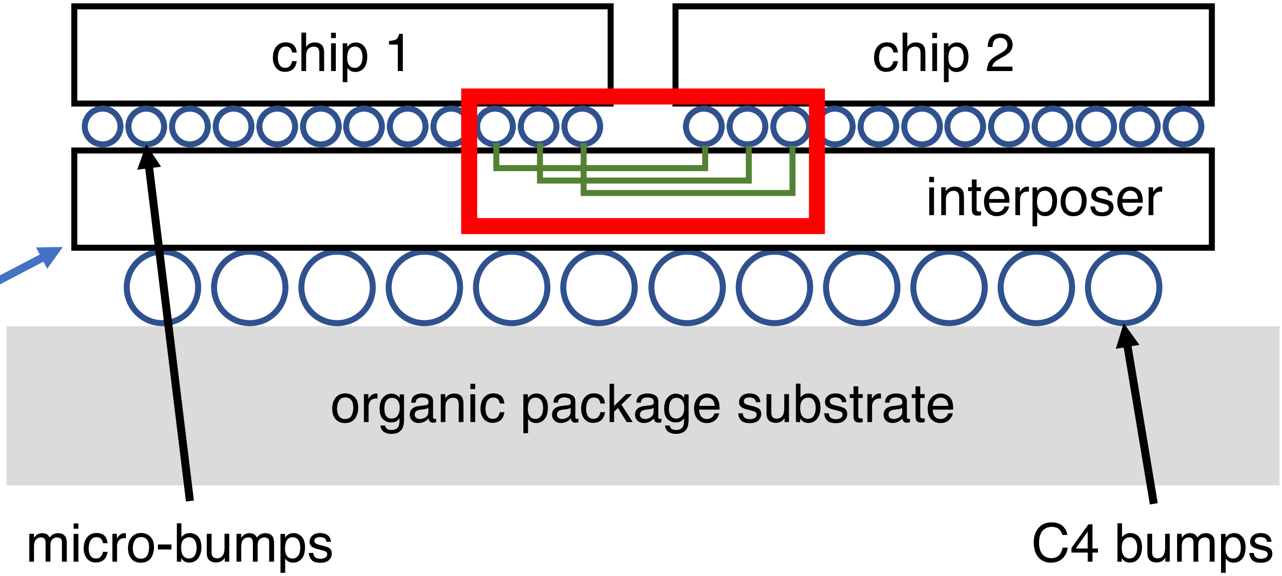
Multi-chip module (MCM)

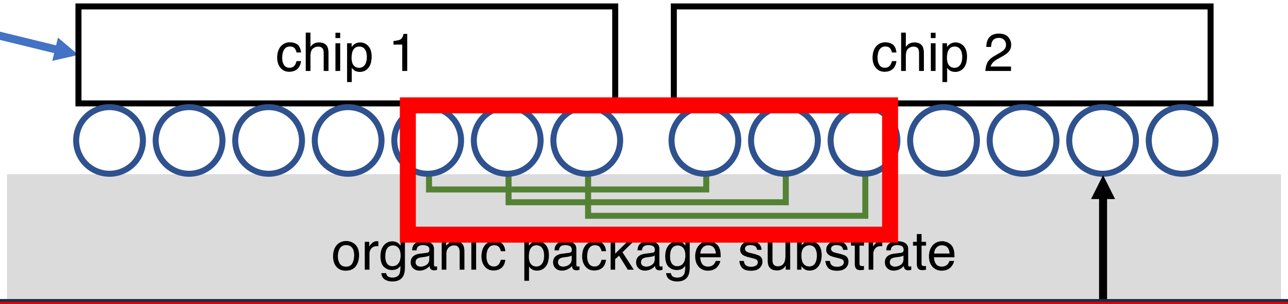
Multi-socket

Multi-board

ntegrated

t Cost





Disintegration constrains the inter-chip bandwidth and latency

Methodology

Diverse suite of 29 multithreaded benchmarks

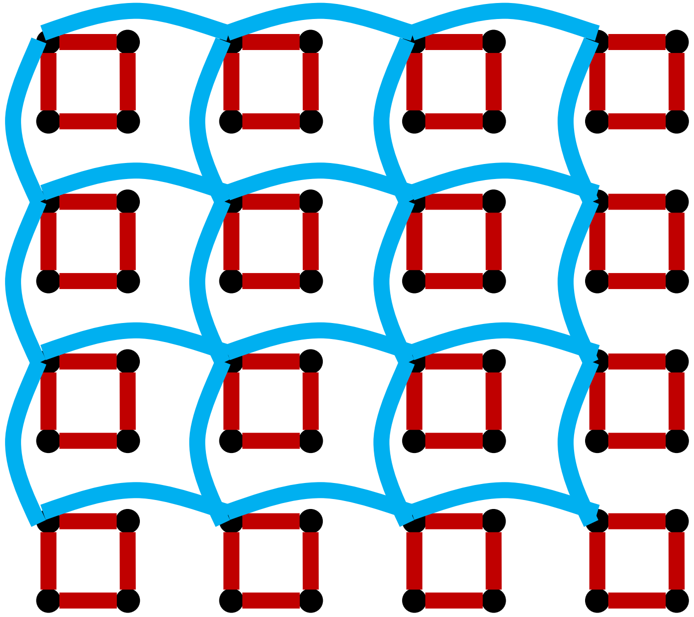
- Splash-2x, Splash-4, Parsec, PBBS, etc.
- Regular and irregular parallel algorithms
- 10 runs/benchmark

Cycle level simulator[1]

- 256 cores, 2-wide out-of-order
- 4 cores/tile, 1 on-chip router/tile
- 16 chips, 16 cores/chip

Implement 3 network topologies

- Monolith (mesh)
- MCM: Hierarchical Mesh



MCM

1: https://github.com/SwarmArch/sim

Motivation

The disintegration slowdown varies across applications

Why Disintegrate? → To Increase Performance/$

$$\frac{Perf_{Disintegrated}(A)}{\$_{Disintegrated}} > \frac{Perf_{Monolith}(A)}{\$_{Monolith}}$$

Why Disintegrate? → To Increase Performance/$

$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)} > \frac{\$_{Disintegrated}}{\$_{Monolith}}$$

Why Disintegrate? → To Increase Performance/$

$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)} > \frac{\$_{Disintegrated}}{\$_{Monolith}}$$

↑ Disintegration slowdown

- Varies per ***A***pplication

↑ Cost savings

- Fixed for a system
- e.g., 0.59x for AMD EPYC[1]

1: Naffziger et al., ISCA 2021

Why Disintegrate? → To Increase Performance/$

This work →

$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)} > \frac{\$_{Disintegrated}}{\$_{Monolith}}$$

Disintegration slowdown

- Varies per ***A***pplication

Cost savings

- Fixed for a system
- e.g., 0.59x for AMD EPYC[1]

What is the performance penalty of disintegration across applications?

1: Naffziger et al., ISCA 2021

Why Disintegrate? → To Increase Performance/$

This work →

$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)} > \frac{\$_{Disintegrated}}{\$_{Monolith}}$$

Disintegration slowdown

- Varies per ***A***pplication

Cost savings

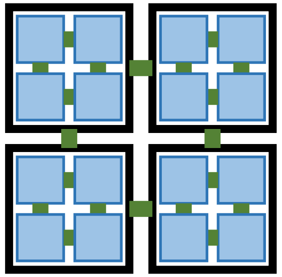
- Fixed for a system
- e.g., 0.59x for AMD EPYC[1]

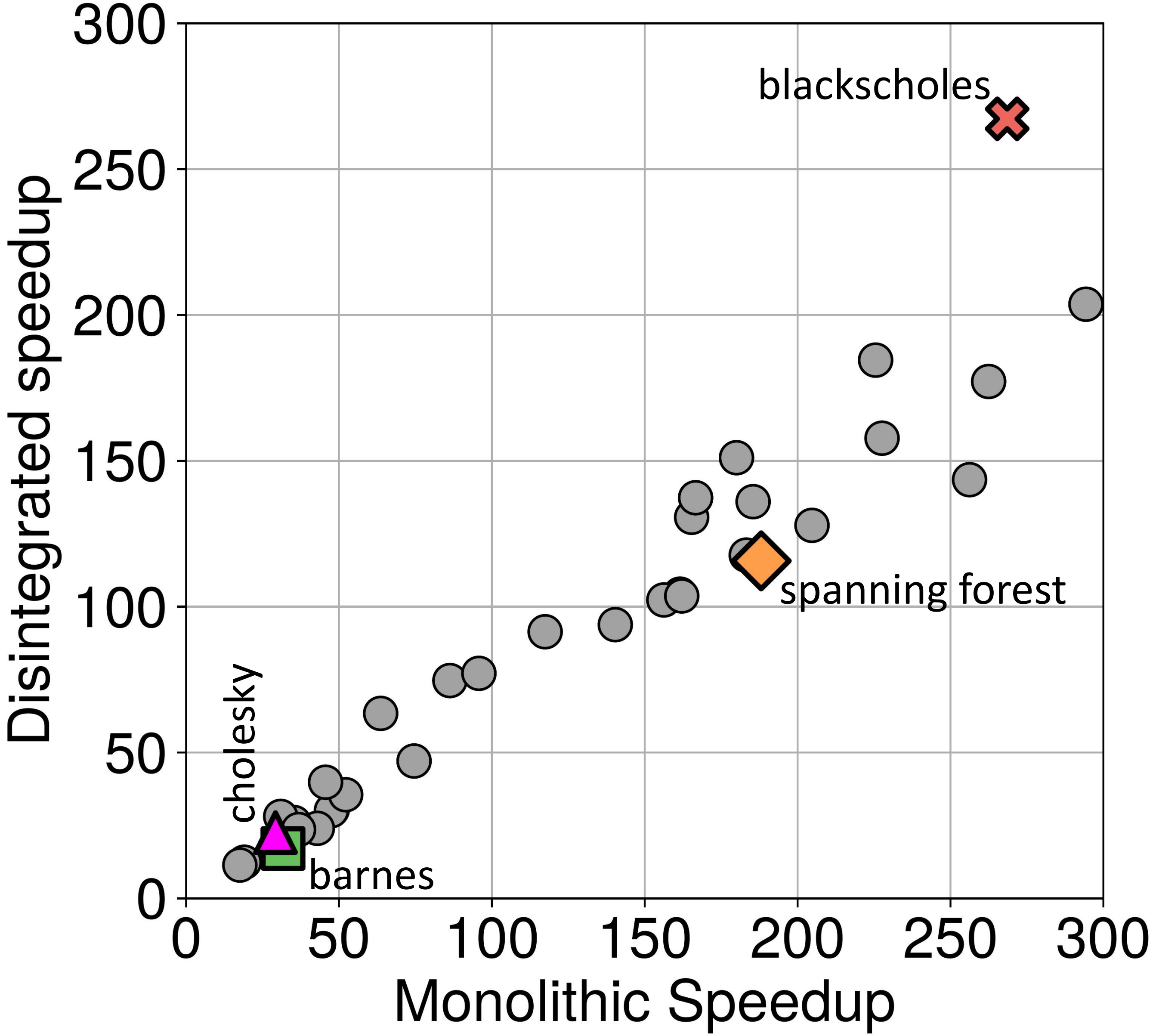
What is the performance penalty of disintegration across applications?

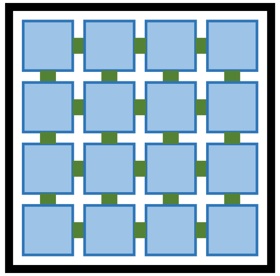
What application characteristics lead to slowdown on disintegrated systems?

1: Naffziger et al., ISCA 2021

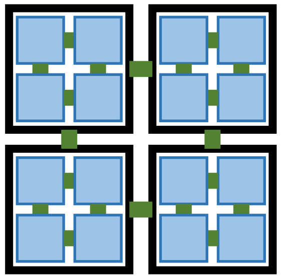
Unsurprisingly, Applications Continue to Scale

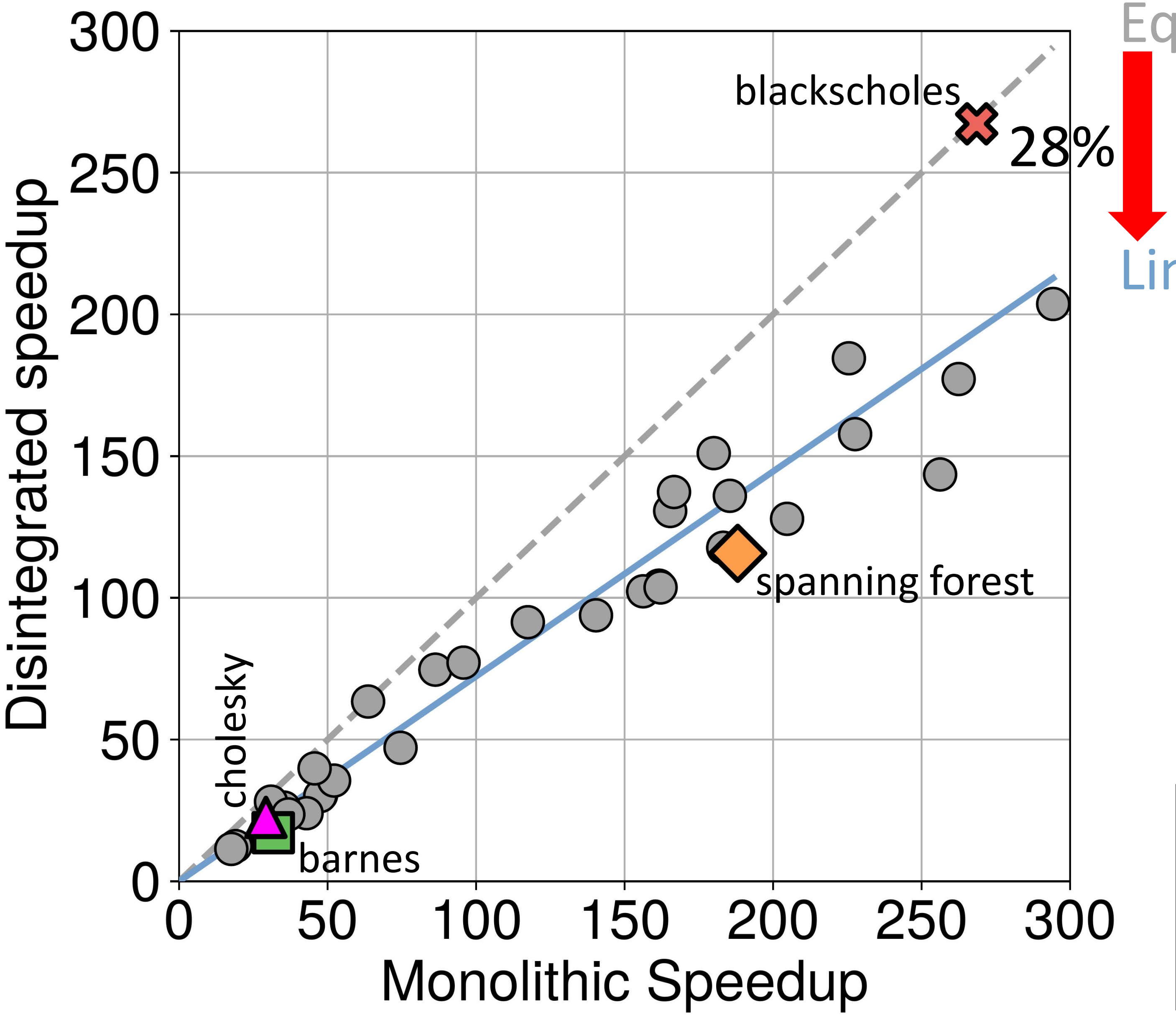






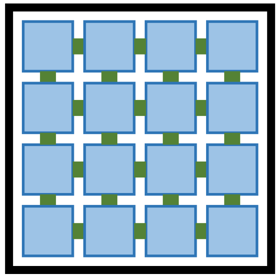
Unsurprisingly, Applications Continue to Scale



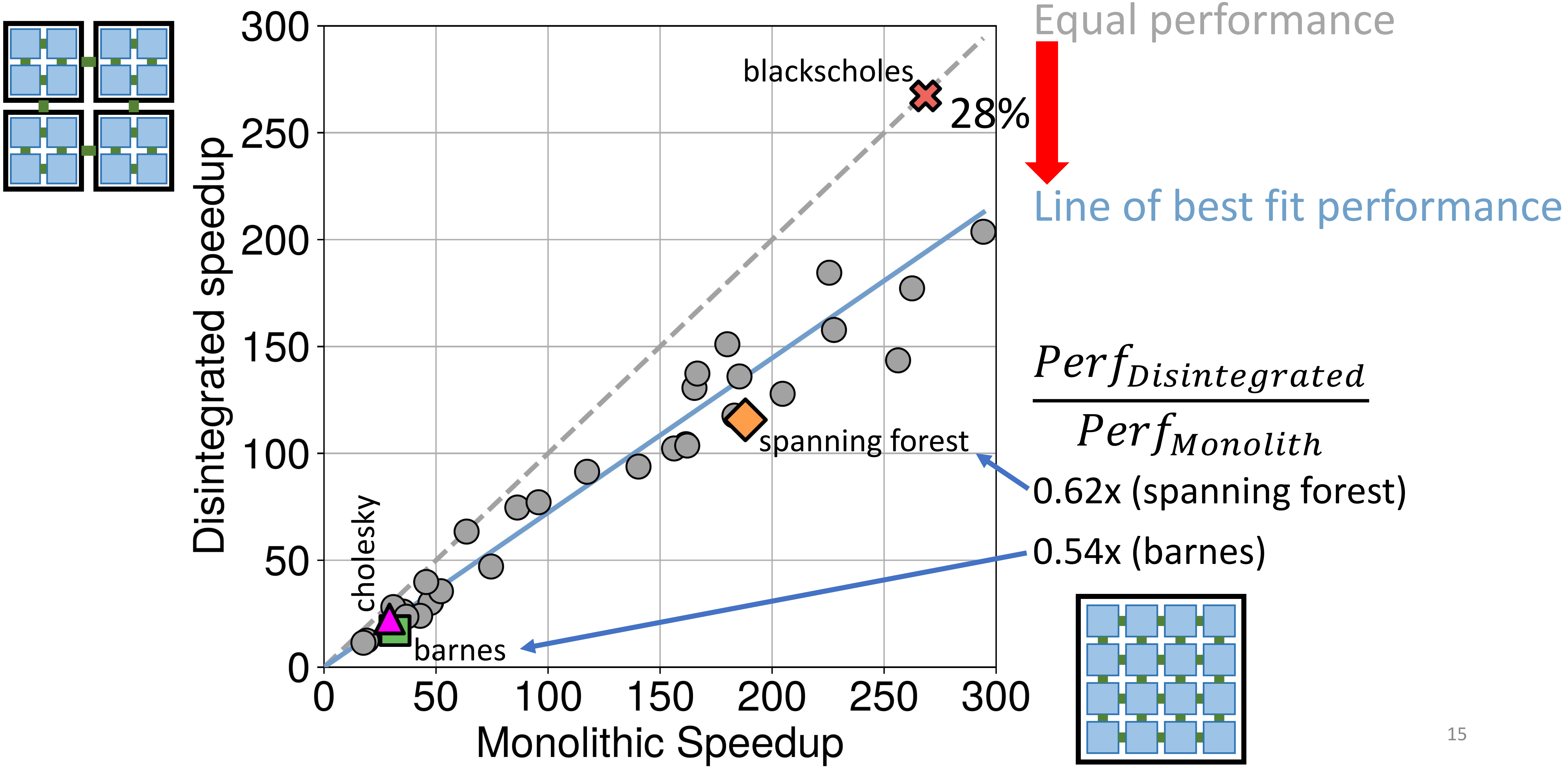


Equal performance

Line of best fit performance



Unsurprisingly, Applications Continue to Scale



$$\frac{Perf_{Disintegrated}}{Perf_{Monolith}}$$

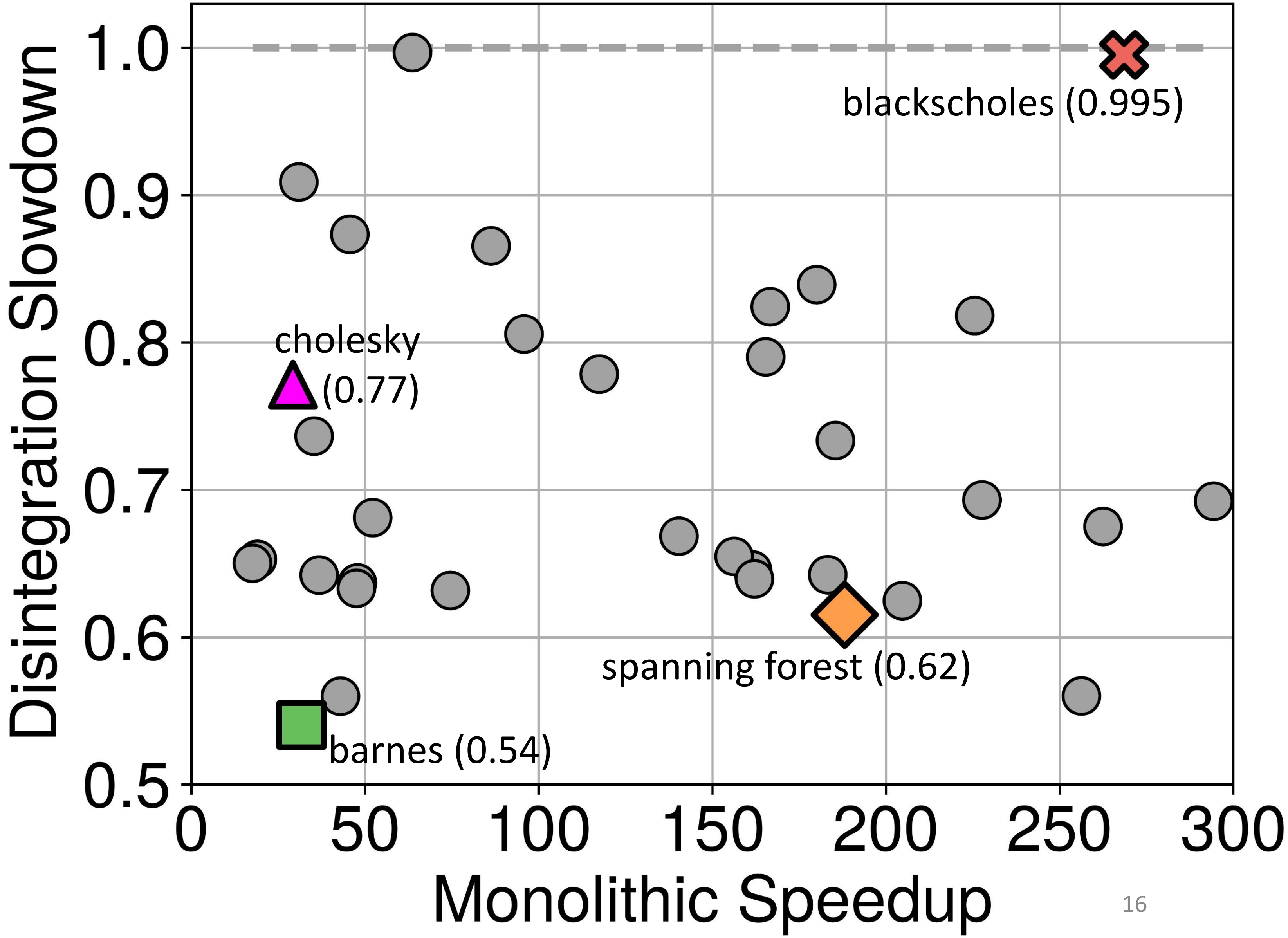
0.62x (spanning forest)

0.54x (barnes)

Large Variance in Application Slowdown

Disintegration Slowdown

$$= \frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)}$$

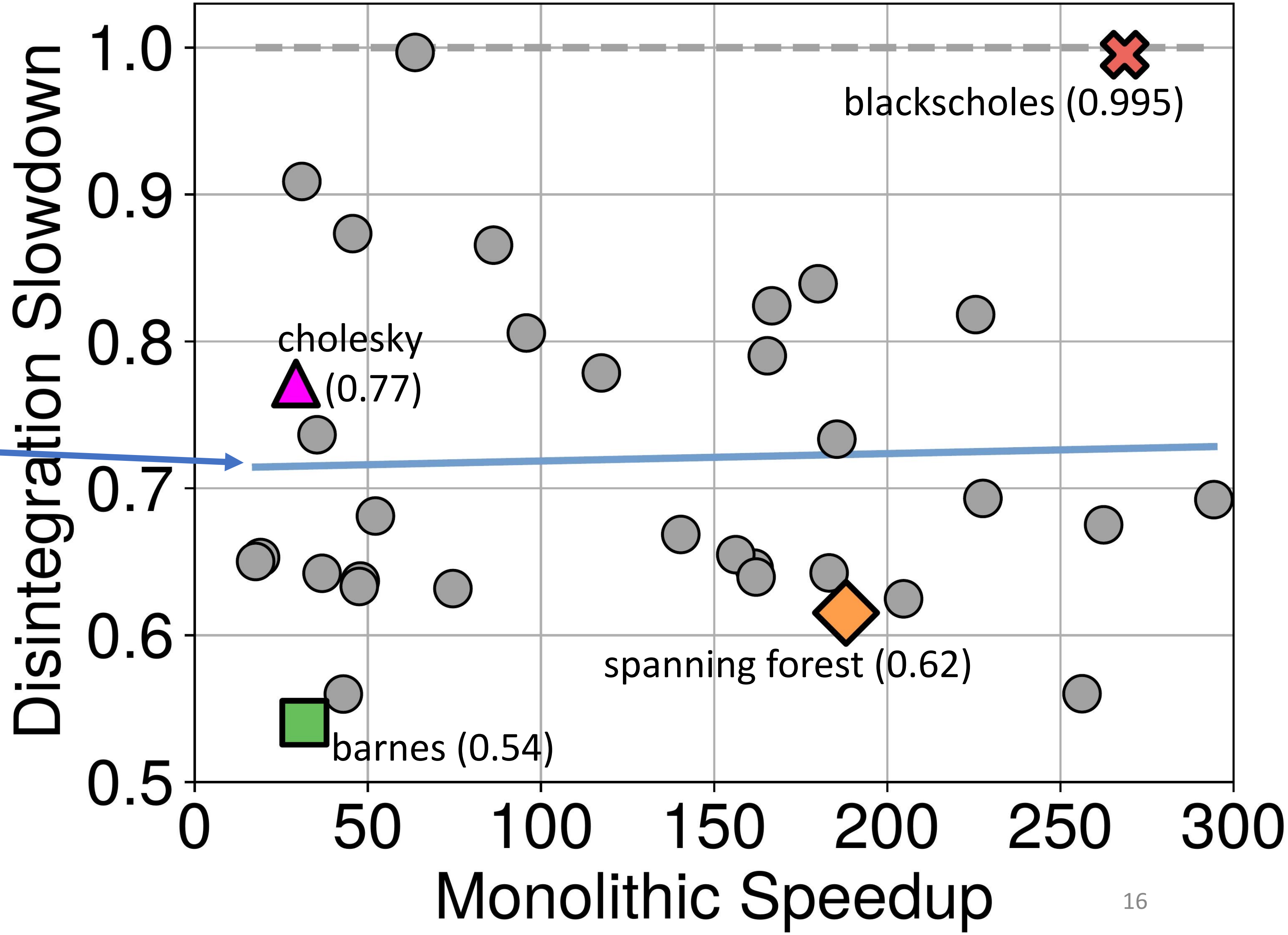


Large Variance in Application Slowdown

Disintegration Slowdown

$$= \frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)}$$

0.72x – average disintegration slowdown

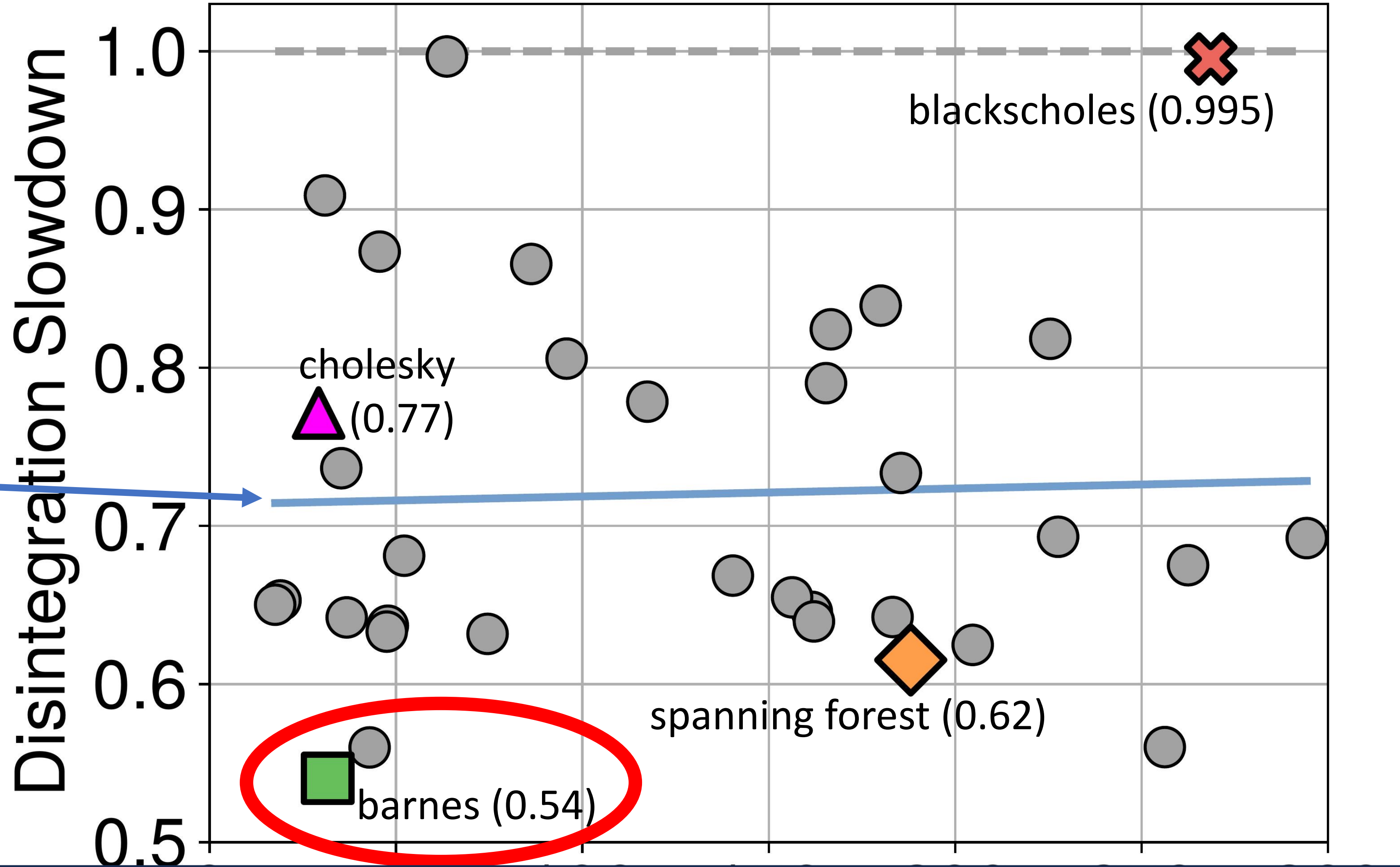


Large Variance in Application Slowdown

Disintegration Slowdown

$$= \frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)}$$

0.72x – average disintegration slowdown

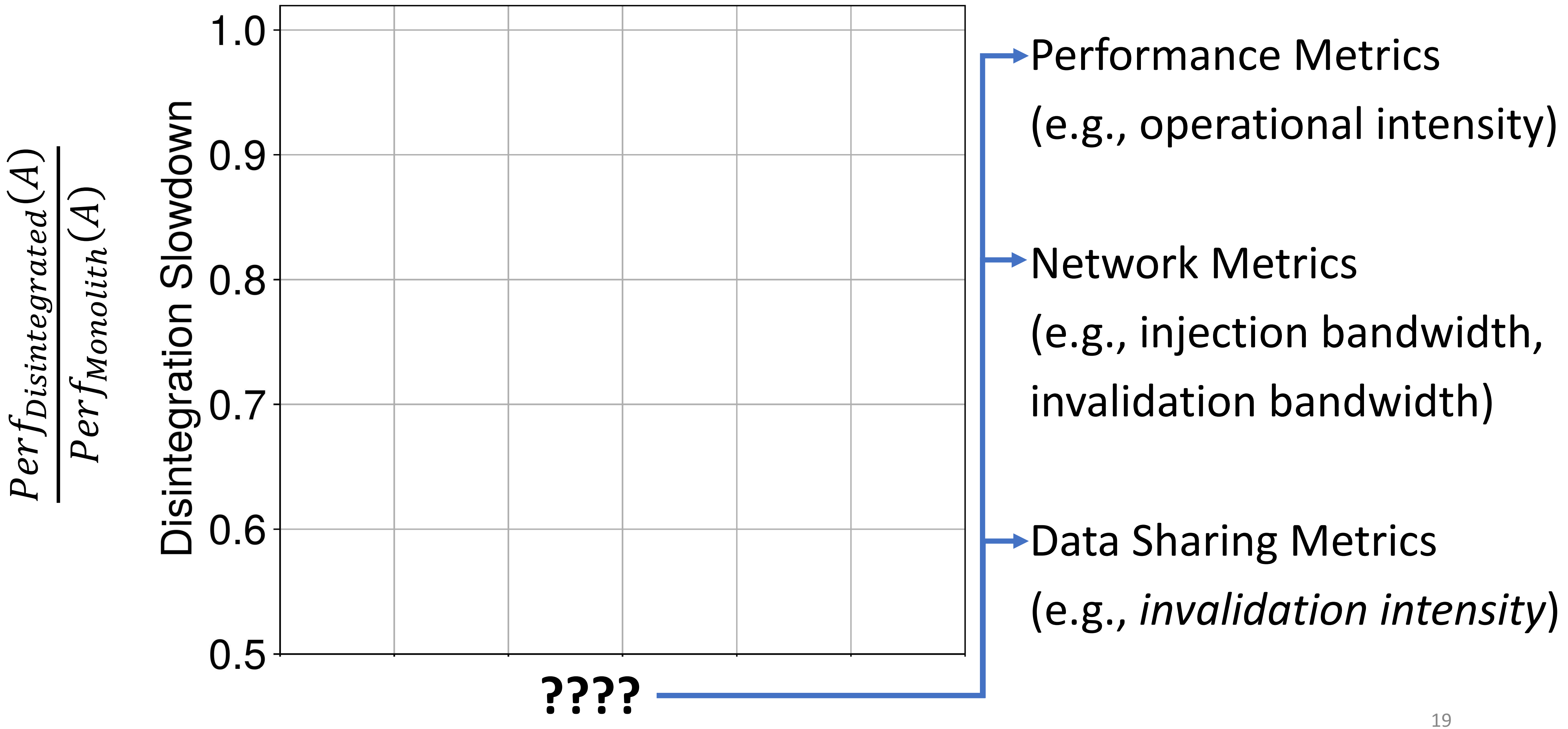


Monolithic scalability does not predict disintegration slowdown

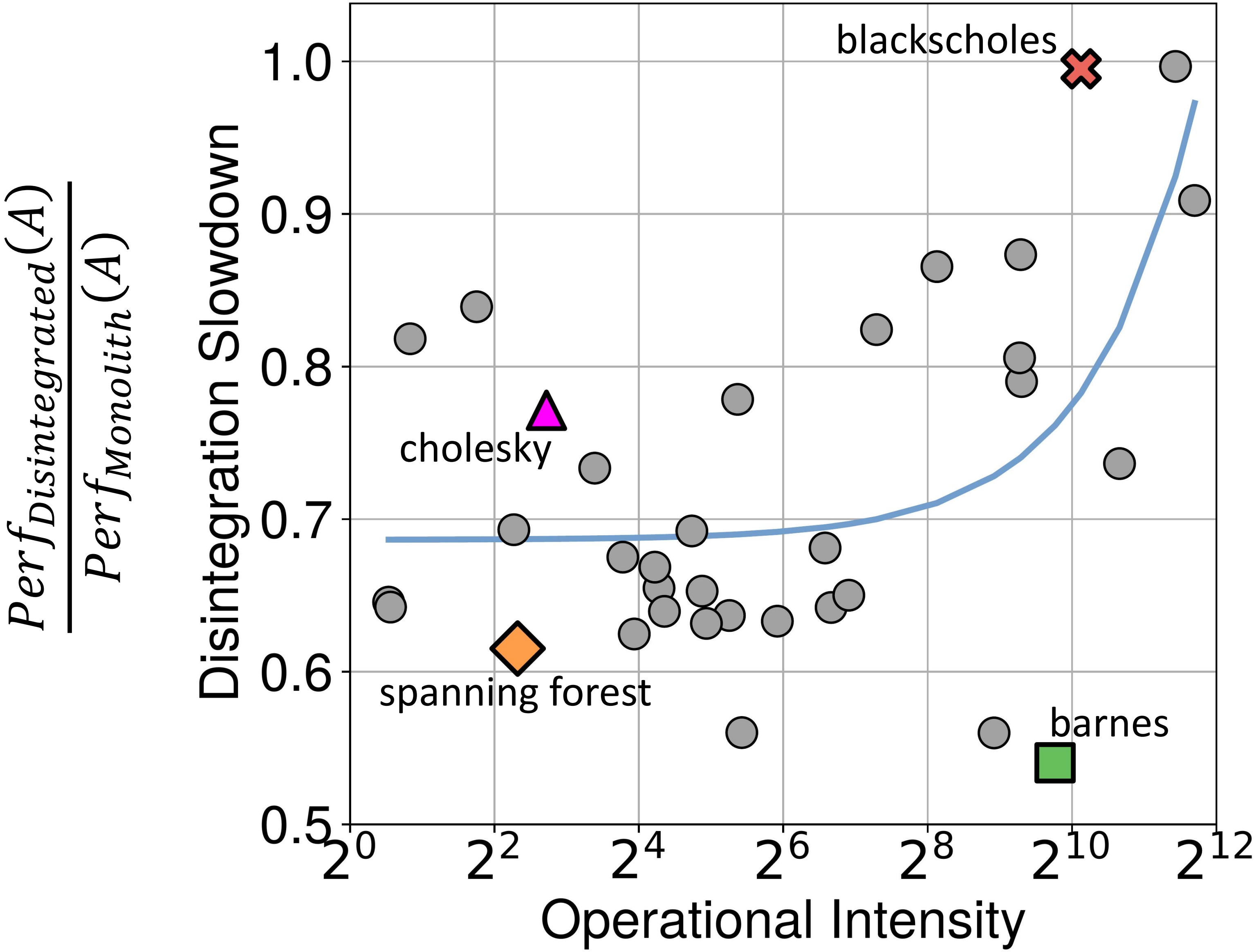
Results

What application characteristics predict slowdown on disintegrated systems?

What Metrics Correlate with Slowdown?



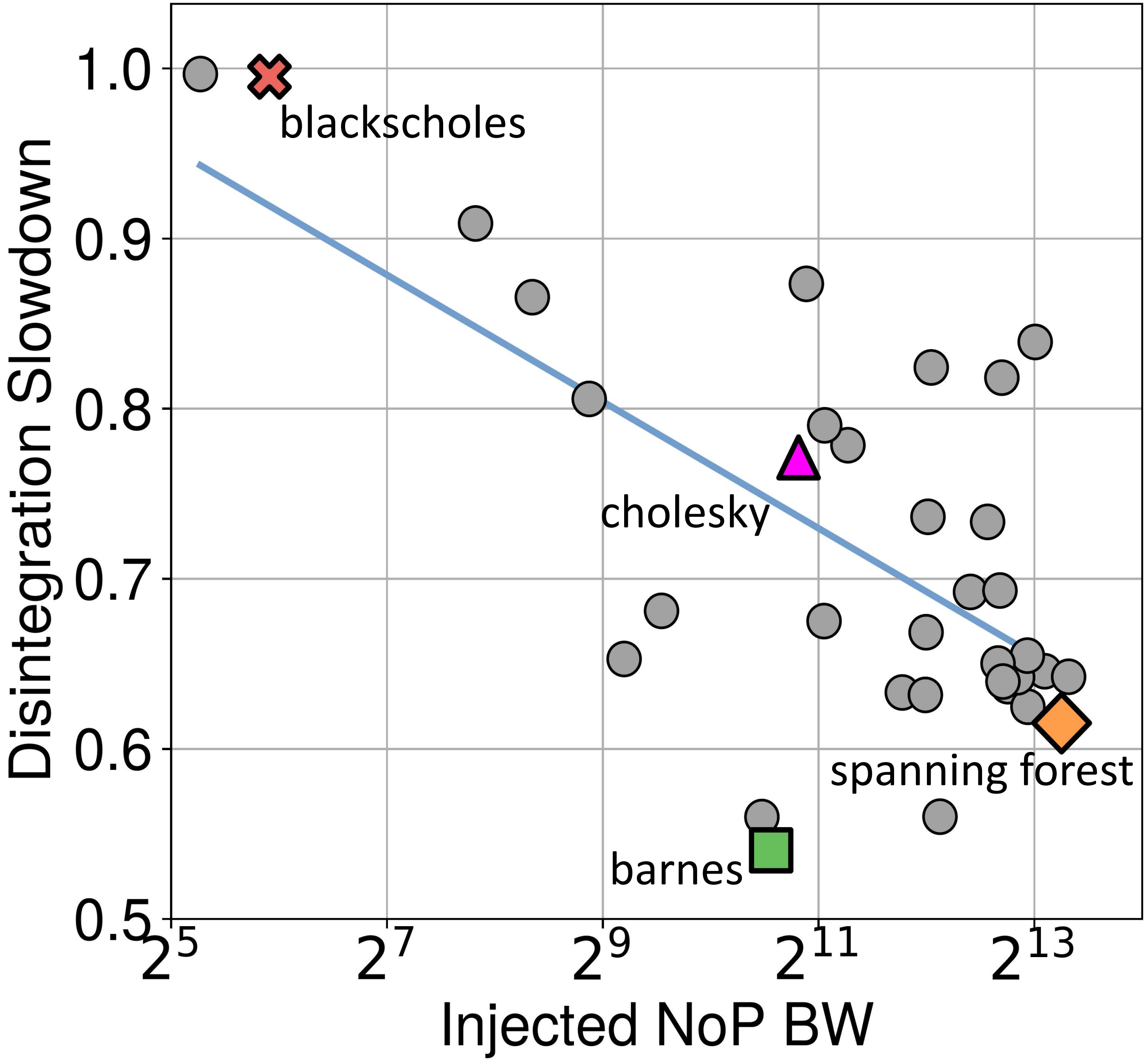
Performance Metrics: Do Not Explain Disintegration Slowdown



Operational Intensity:
instrs/byte
correlation up to 0.32

Network Bandwidth: Correlates Better With Disintegration Slowdown

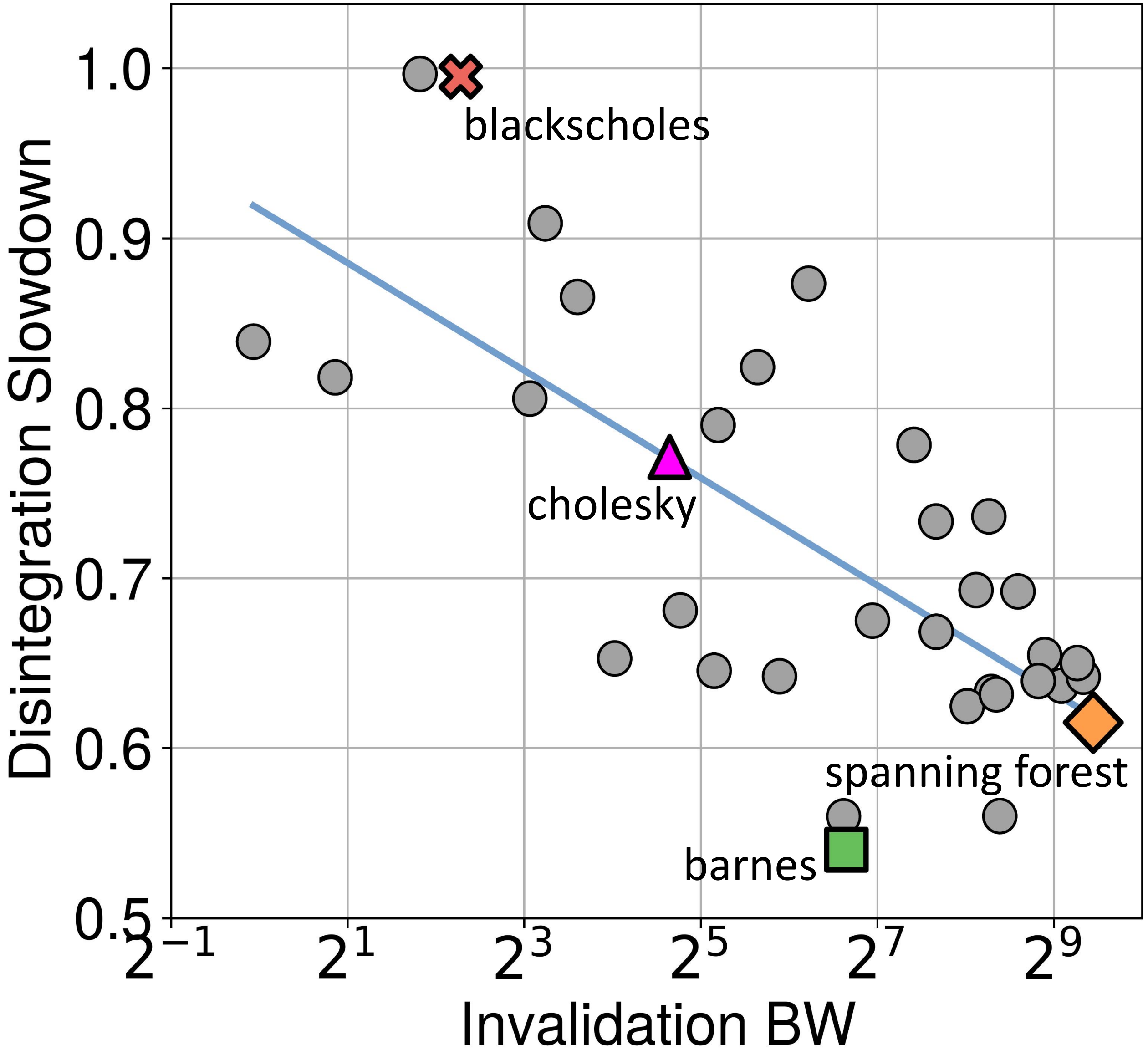
$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)}$$



Network bandwidth:
MB/s/tile
correlation up to 0.42

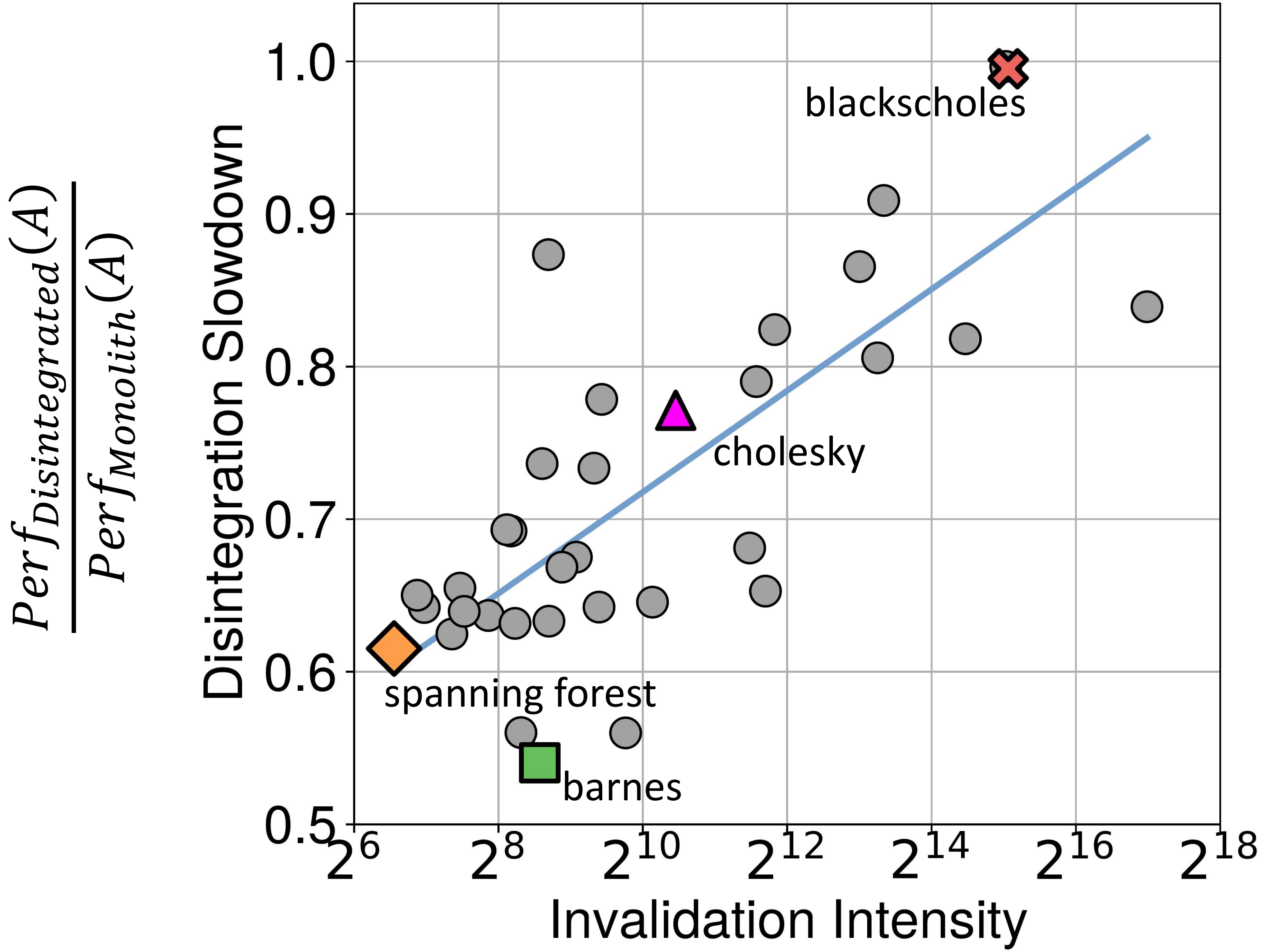
Invalidation Bandwidth: Predict Slowdown Better Than Total Bandwidth

$$\frac{Perf_{Disintegrated}(A)}{Perf_{Monolith}(A)}$$



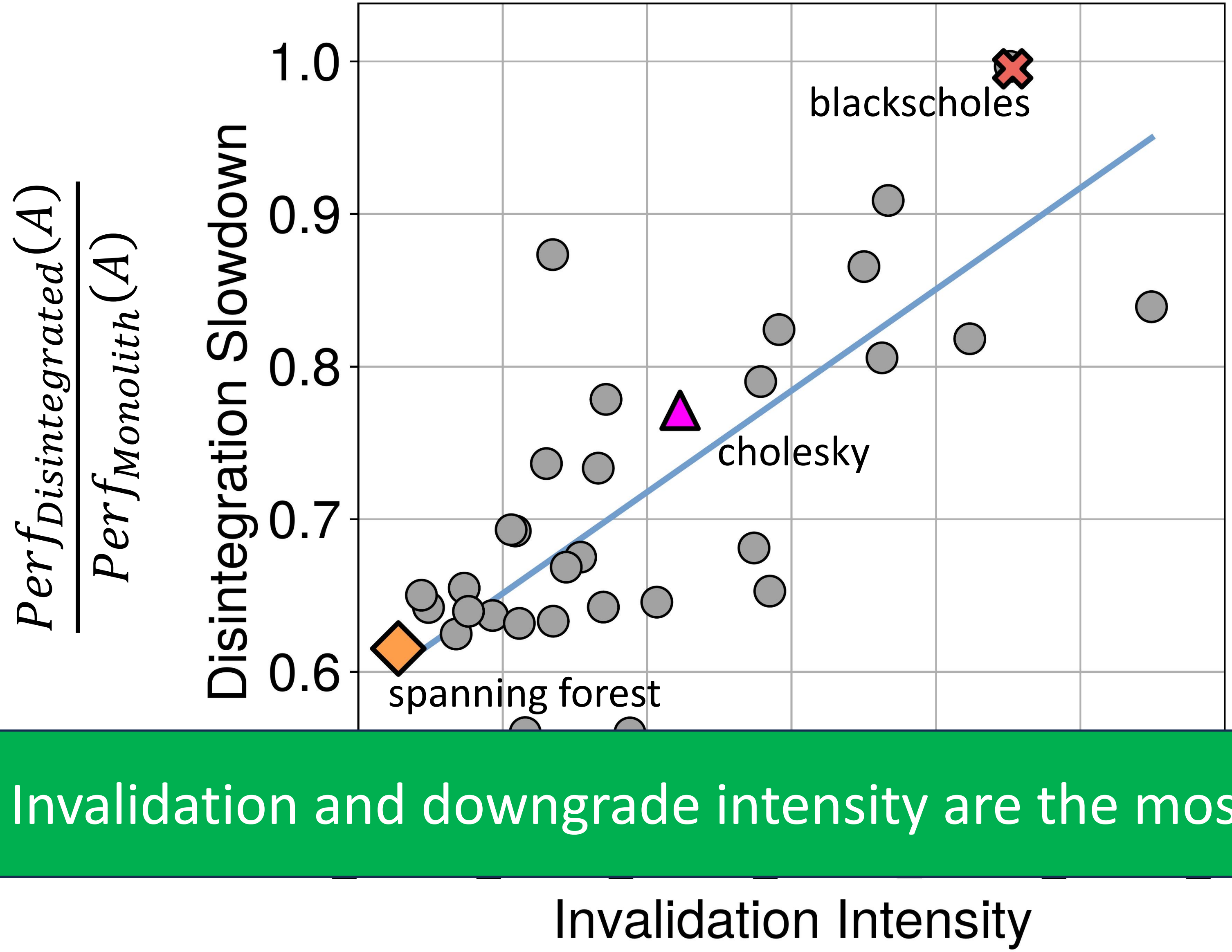
Invalidation bandwidth:
MB of Invalidations /s/tile
correlation up to 0.50

Our Data Sharing Metrics: Have The Best Observed Correlation



Invalidation intensity:
Instructions / Invalidation
correlation up to 0.58

Our Data Sharing Metrics: Have The Best Observed Correlation



Invalidation intensity:
Instructions / Invalidation
correlation up to 0.58

Invalidation and downgrade intensity are the most predictive known metrics

Metric Group	Metric	Correlation (ideal is 1.0)
Performance	IPC	0.14
	Operational Intensity	0.32
	Consumed Memory Bandwidth	0.13
Base Network	Network Injection Bandwidth	0.42
	Average Network Latency	0.06
Network Injection Bandwidth	GetS	0.41
	GetX	0.49
	Inv (Invalidate)	**0.50**
	InvX (Downgrade)	**0.57**
	Data Response	0.46
	PutS (Clean Eviction)	0.10
	PutX (Dirty Eviction)	0.05
Data Sharing	Read Sharers	0.07
	Invalidation Intensity	**0.58**
	Downgrade Intensity	**0.59**
	Sharing Fraction [Ferdman+, 2012]	0.31

What application characteristics correlate with disintegration slowdown?

Metric Group	Metric	Correlation (ideal is 1.0)
Performance	IPC	0.14
Base Network	Network Injection Bandwidth	0.42
	Average Network Latency	0.06
Network Injection Bandwidth	GetS	0.41
	GetX	0.49
	Inv (Invalidate)	**0.50**
	InvX (Downgrade)	**0.57**
	Data Response	0.46
	PutS (Clean Eviction)	0.10
	PutX (Dirty Eviction)	0.05
Data Sharing	Read Sharers	0.07
	Invalidation Intensity	**0.58**
	Downgrade Intensity	**0.59**
	Sharing Fraction [Ferdman+, 2012]	0.31

What application characteristics correlate with disintegration slowdown?

Metric Group	Metric	Correlation (ideal is 1.0)
Performance	IPC	0.14
Base Network	Network Injection Bandwidth	0.42
	Average Network Latency	0.06
Network Injection Bandwidth	GetS	0.41
	GetX	0.49
	Inv (Invalidate)	**0.50**
	InvX (Downgrade)	**0.57**
	Data Response	0.46
	PutS (Clean Eviction)	0.10
	PutX (Dirty Eviction)	0.05
Data Sharing	Read Sharers	0.07
	Invalidation Intensity	**0.58**
	Downgrade Intensity	**0.59**
	Sharing Fraction [Ferdman+, 2012]	0.31

Conclusion

Industry has adopted disintegrated systems to reduce cost

The penalty of disintegration varies across applications

Disintegration constrains inter-chiplet links

Sharing intensity is correlated with worse disintegration slowdown

Future work: Support performance for **all** applications on disintegrated systems

Q & A

Industry has adopted disintegrated systems to reduce cost

The penalty of disintegration varies across applications

Disintegration constrains inter-chiplet links

Sharing intensity is correlated with worse disintegration slowdown

Future work: Support performance for **all** applications on disintegrated systems

Isidor R. Brkić, Mark C. Jeffrey

UNIVERSITY OF TORONTO