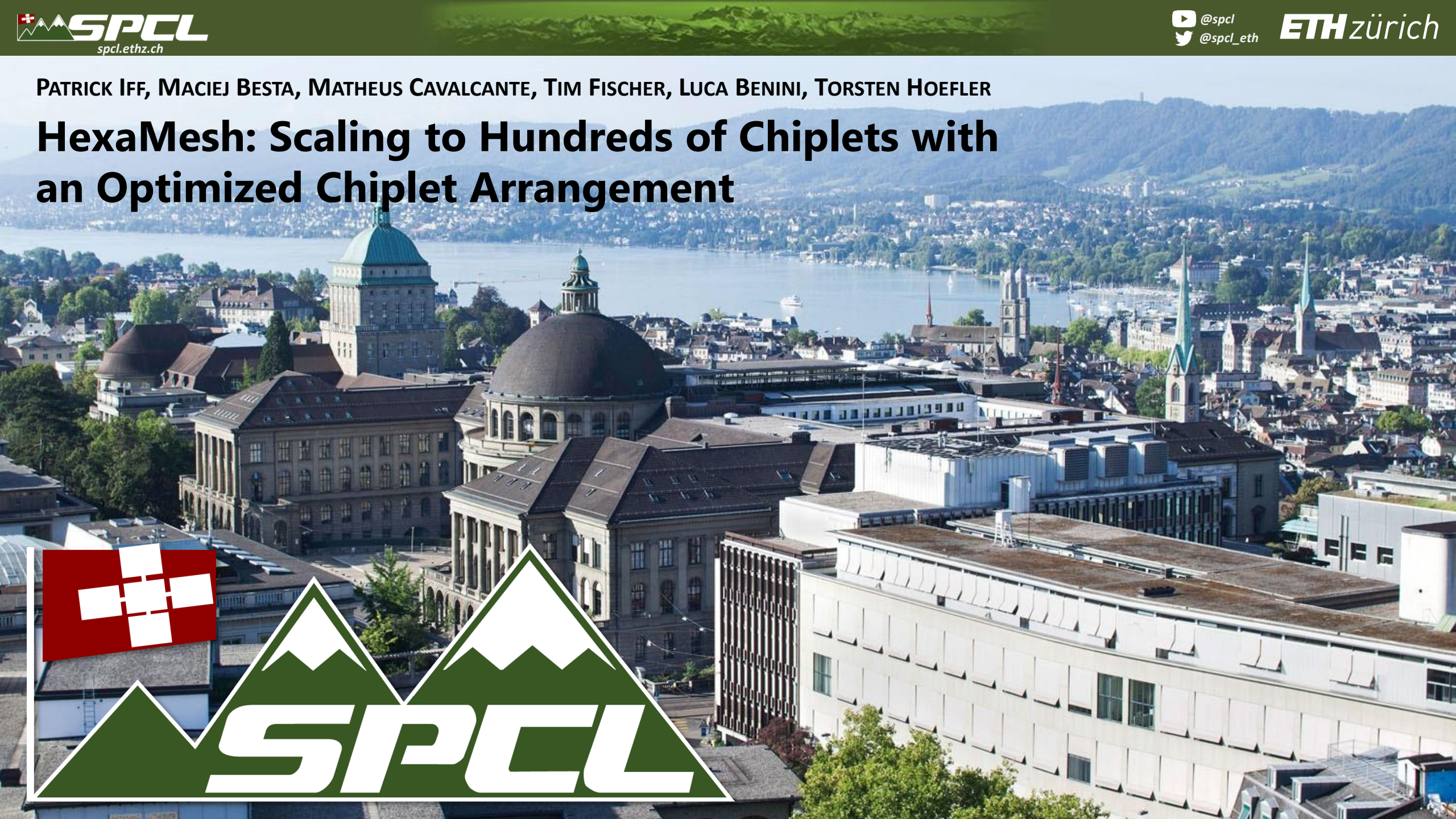
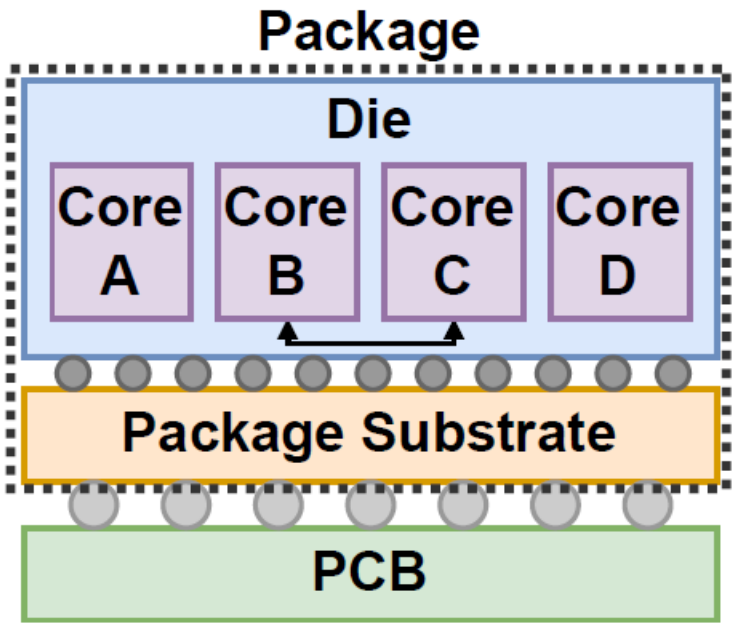
Patrick Iff, Maciej Besta, Matheus Cavalcante, Tim Fischer, Luca Benini, Torsten Hoefler

HexaMesh: Scaling to Hundreds of Chiplets with an Optimized Chiplet Arrangement



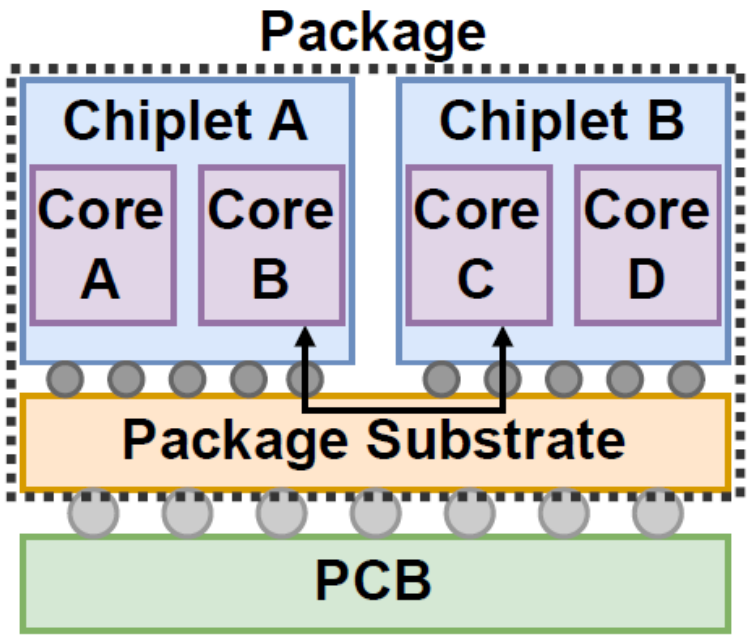
2.5D Integration

Monolithic Chip

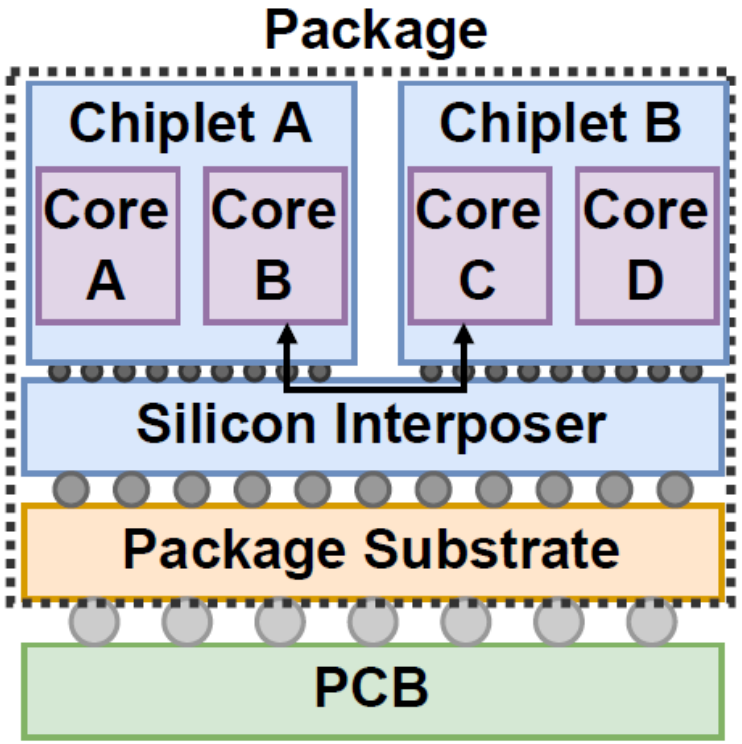


2.5D Integration

Organic Package Substrate



Silicon Interposer



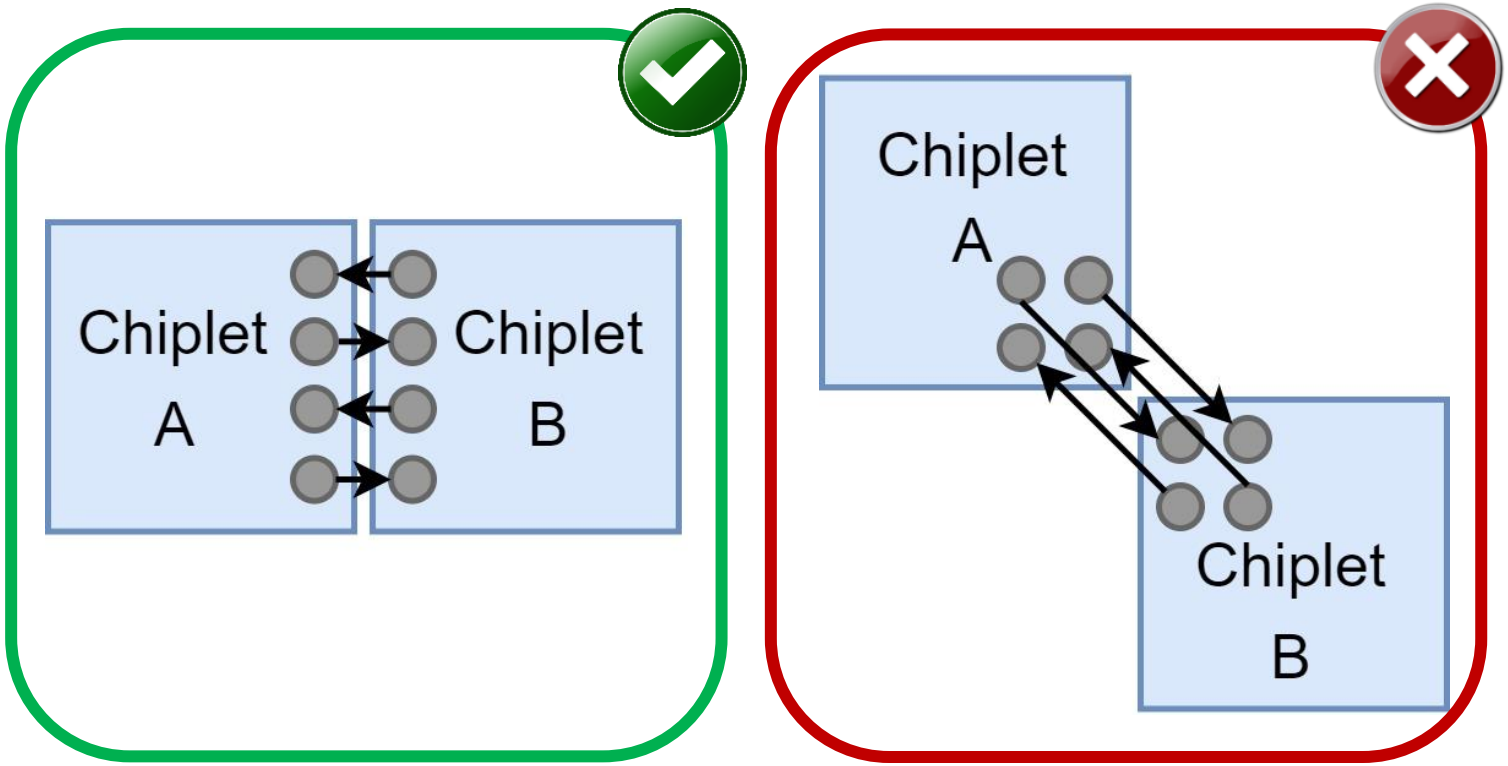
Legend: Solder ball (500-1000µm) C4 bump (150-200µm) Micro-bump (30-60µm)

2.5D Integration

Benefits	Drawbacks
Heterogeneity	Area- and Power Overhead due to PHYs
Reuse	The Inter-Chiplet Interconnect is the Bottleneck
Improved Yield	
Per-chiplet Binning	

Key Insights

- Number of bumps is limited
- Number and datawidth of links is limited
- Maximize link throughput
- Maximize link frequency
- Minimize link length
- Only connect adjacent chiplets



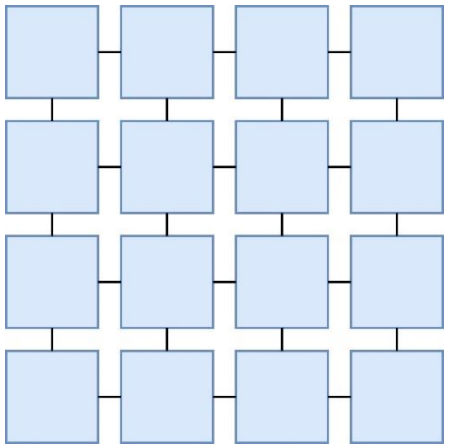
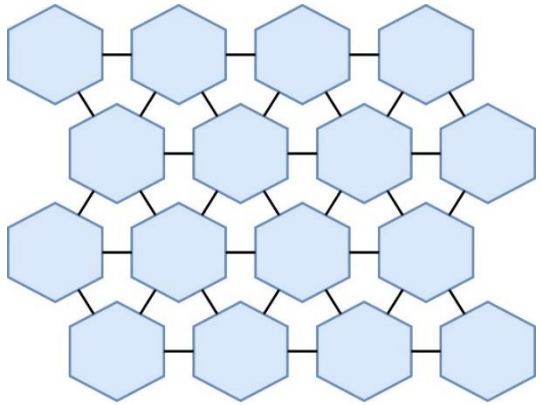
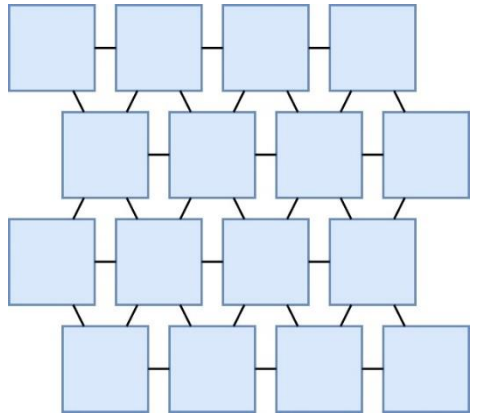
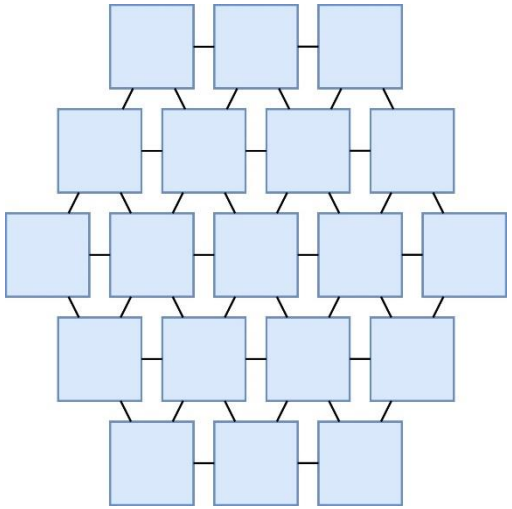
Chiplet **shape** and **arrangement** are important for the inter-chiplet interconnect performance

Problem Statement

Optimize the Shape and Arrangement of Chiplets

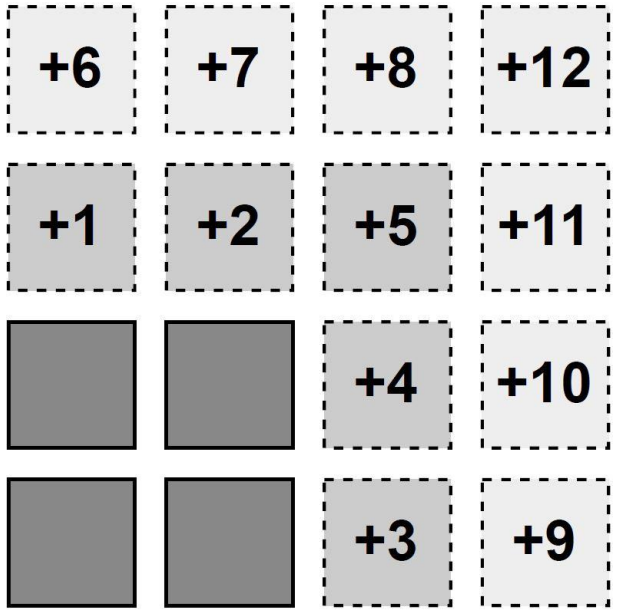
Goals	Constraints
Minimize network diameter (proxy for latency)	All chiplets must have the same shape
Maximize bisection bandwidth (proxy for throughput)	All chiplets must be rectangular

Optimizing Chiplet Arrangement

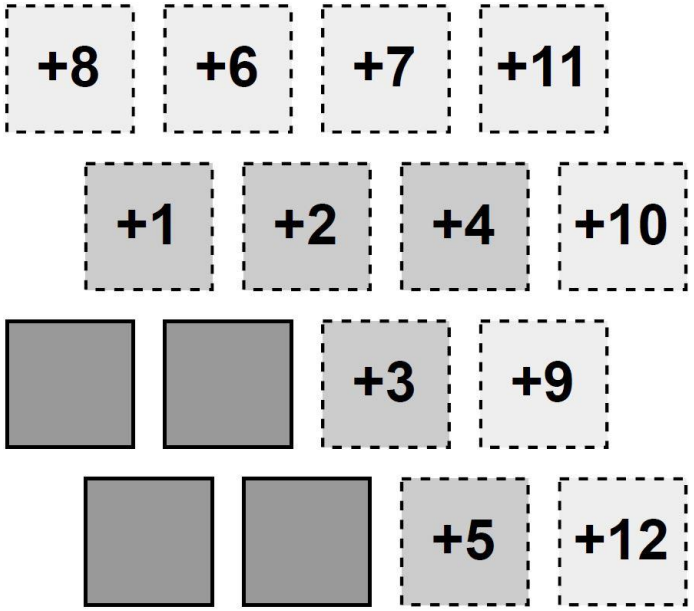
Grid	Honeycomb	Brickwall	HexaMesh
			
Pro: Most straight-forward arrangement	**Pro**: Six neighbors per chiplet (asymptotically optimal)	**Pro**: Like honeycomb but with rectangular chiplets	**Pro**: Each chiplet has at least three neighbors
Con: At most four neighbors per chiplet	**Con**: Non-rectangular chiplets hard to build	**Con**: Some chiplets only have two neighbors	**Pro**: Reduced network diameter compared to brickwall

APPLICABILITY

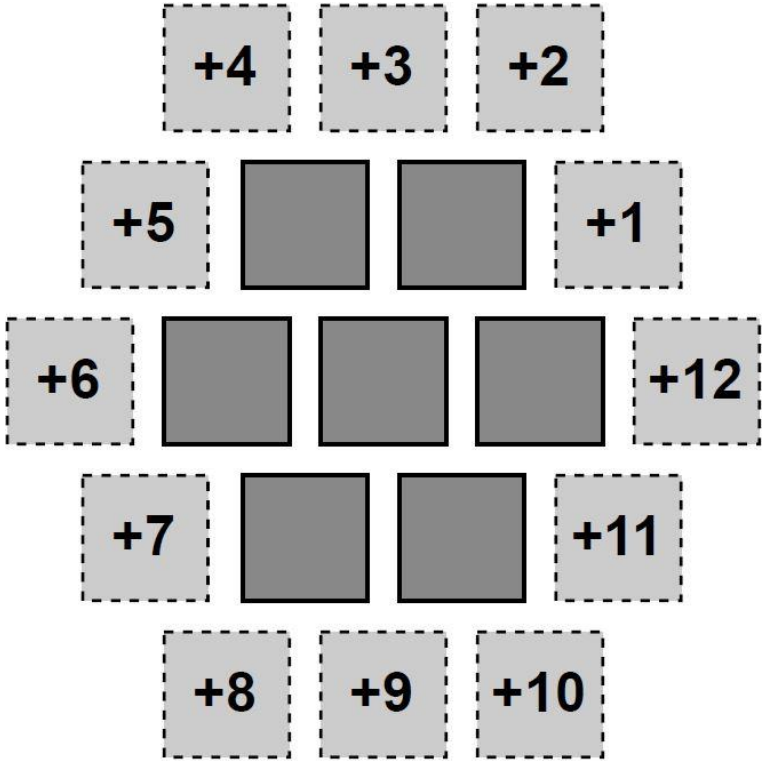
Grid



Brickwall

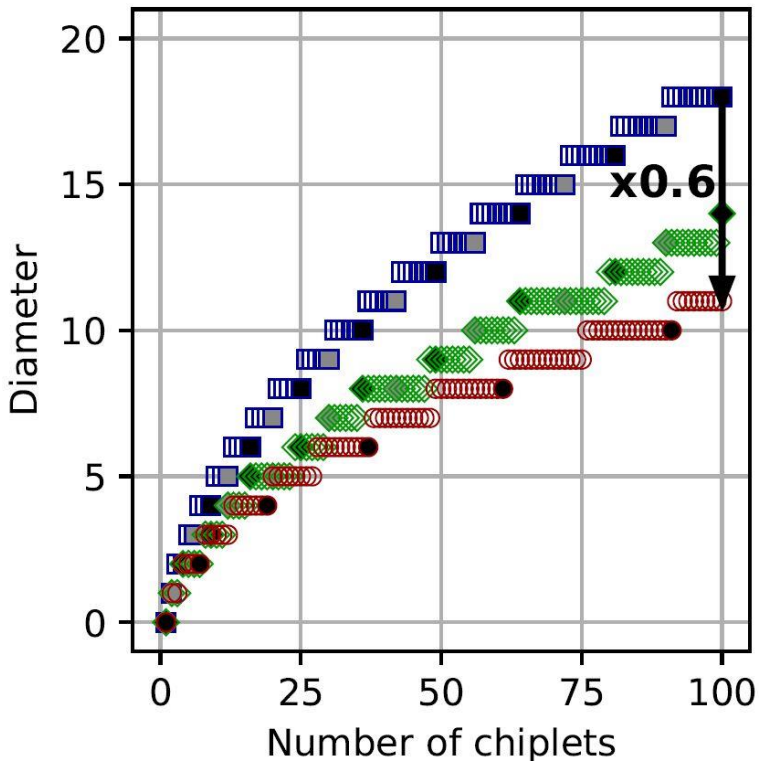


HexaMesh

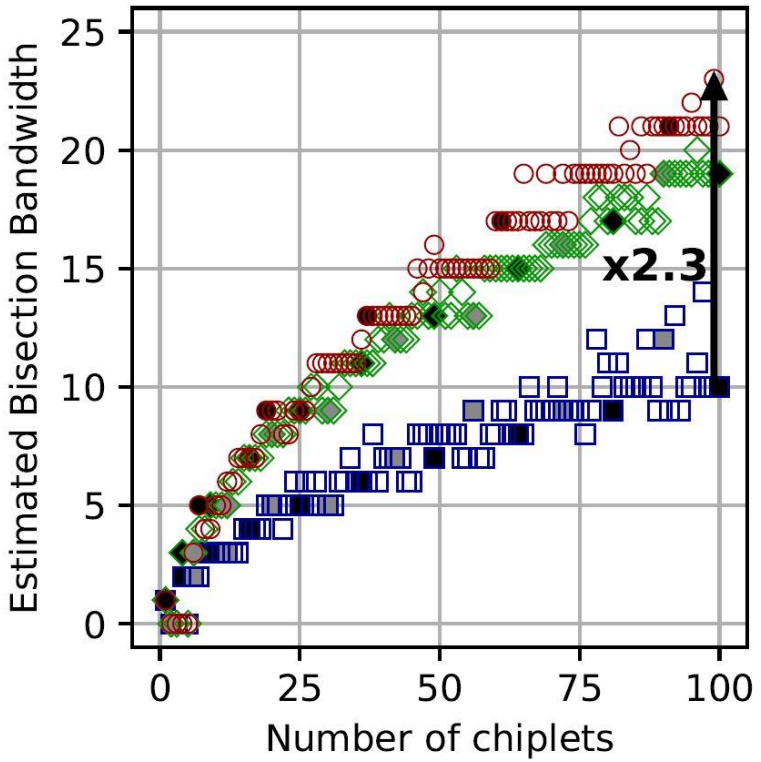


Evaluation of Performance Proxies

Network Diameter



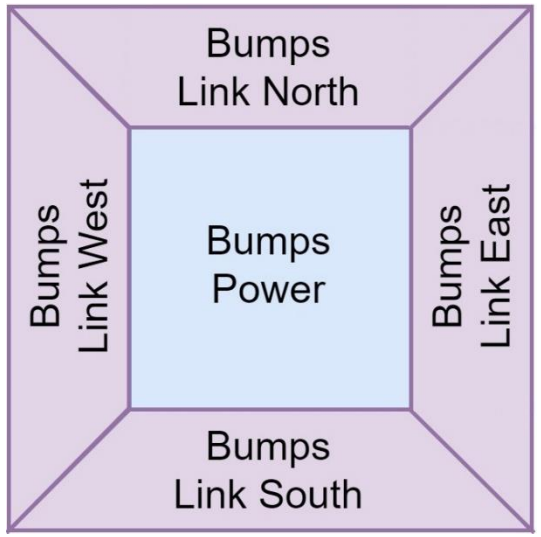
Bisection Bandwidth



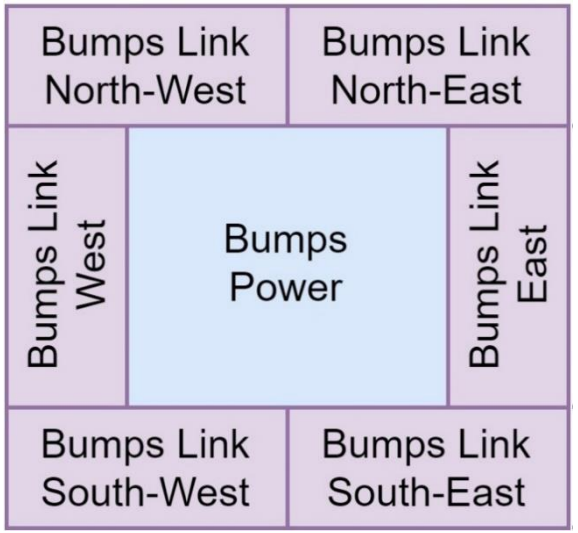
Grid (regular)
Grid (semi-regular)
Grid (irregular)
Brickwall (regular)
Brickwall (semi-regular)
Brickwall (irregular)
HexaMesh (regular)
HexaMesh (irregular)

Shortcomings of the Performance Proxies

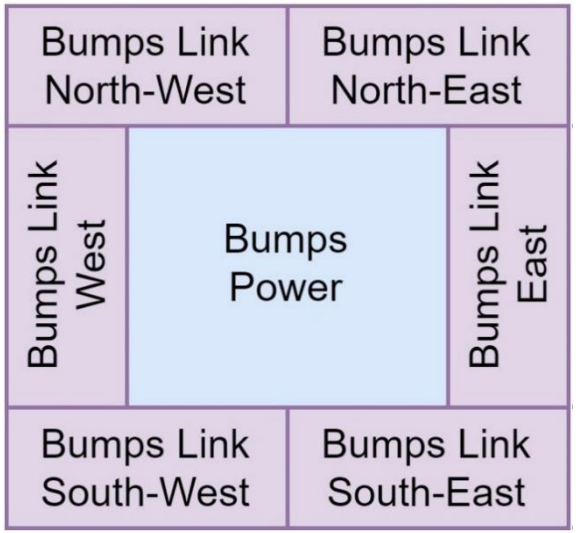
Grid



Brickwall



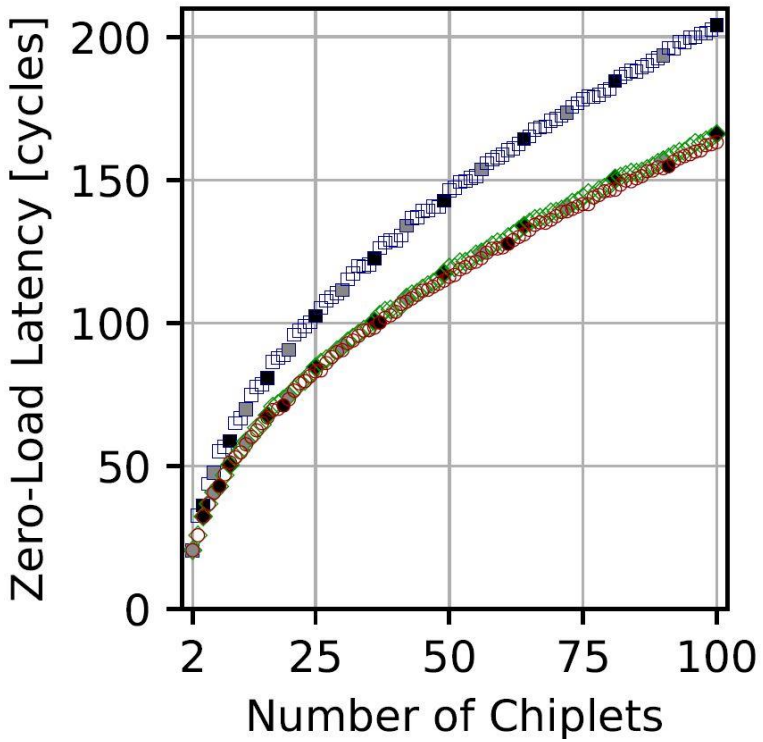
HexaMesh



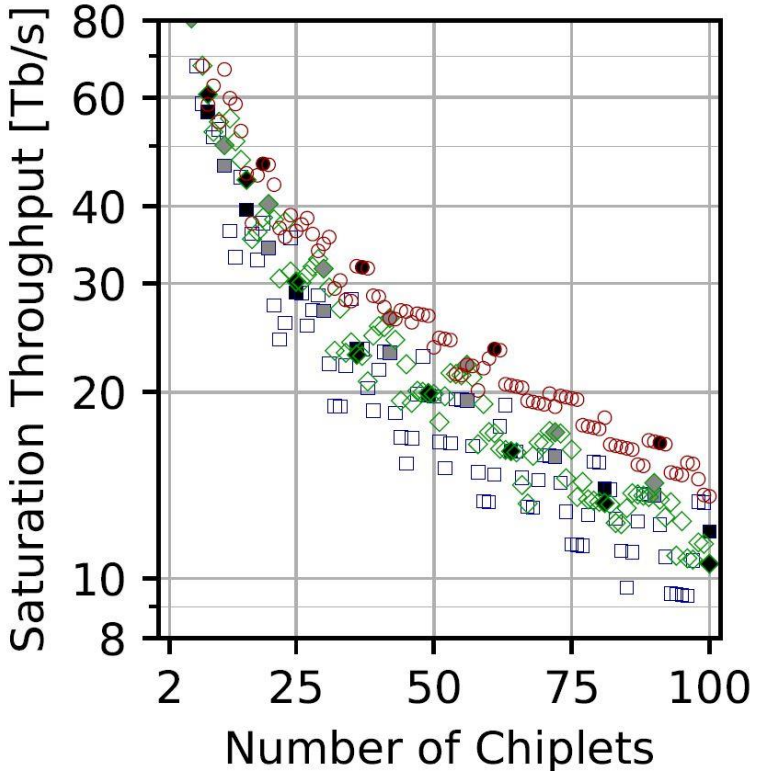
Compute per-link bandwidth based on number of available bumps

Evaluation based on Cycle-Accurate Simulations

Average Latency



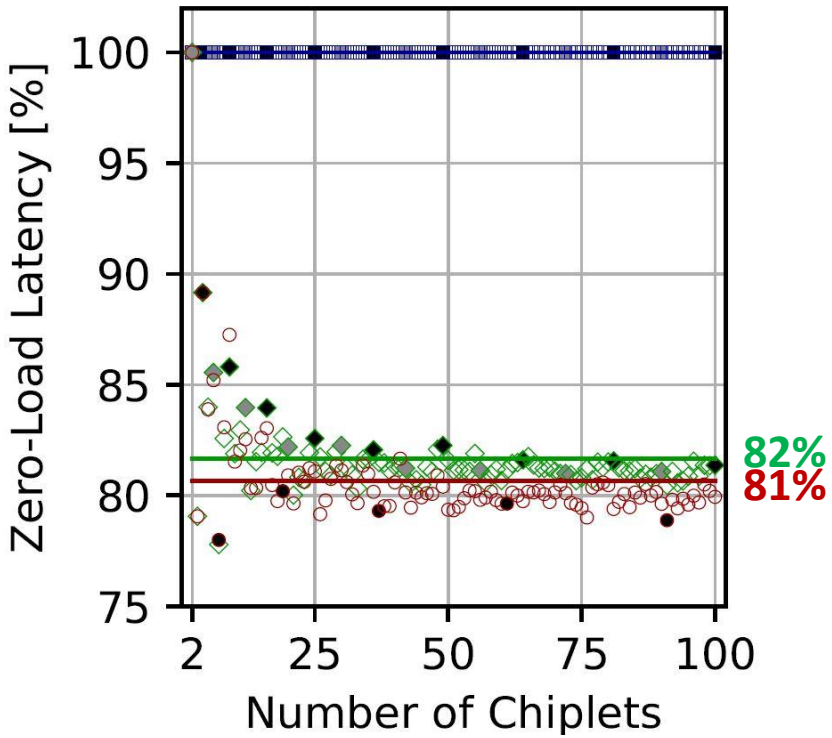
Saturation Throughput



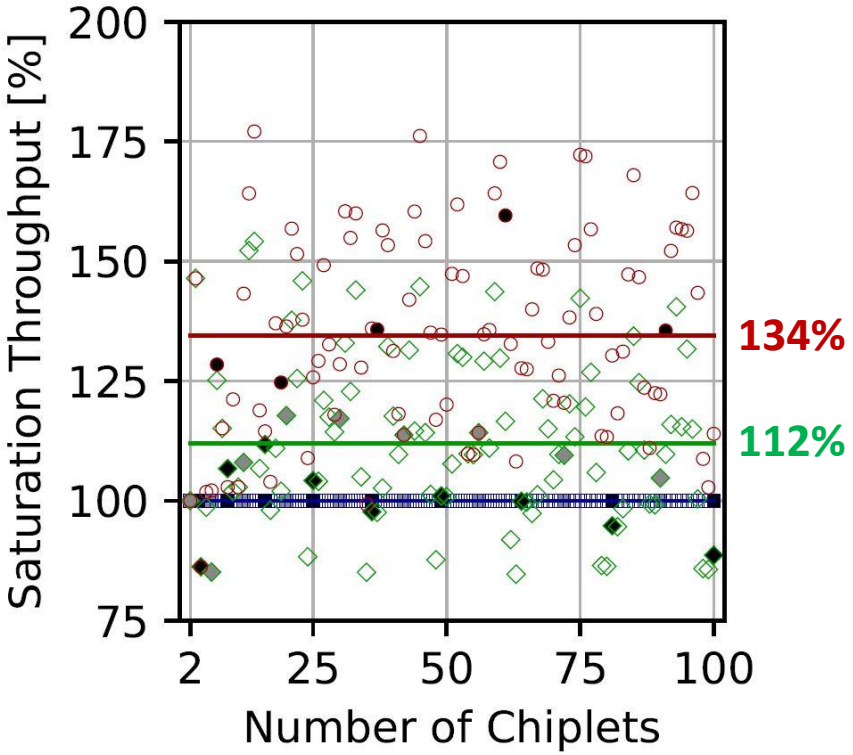
Grid (regular)
Grid (irregular)
Brickwall (regular)
Brickwall (irregular)
HexaMesh (regular)
HexaMesh (AVG)
Grid (semi-regular)
Grid (AVG)
Brickwall (semi-regular)
Brickwall (AVG)
HexaMesh (irregular)

Evaluation based on Cycle-Accurate Simulations

Average Latency



Saturation Throughput



Grid (regular) | Grid (irregular) | Brickwall (regular) | Brickwall (irregular) | HexaMesh (regular) | HexaMesh (AVG)
Grid (semi-regular) | Grid (AVG) | Brickwall (semi-regular) | Brickwall (AVG) | HexaMesh (irregular)

Conclusion

We outperform a grid arrangement in theory:
- **Diameter** reduced by **42%**
- **Bisection bandwidth** improved by **130%**

We outperform a grid arrangement in practice
- **Latency** reduced by **19%** (on average)
- **Throughput** improved by **34%** (on average)

We do not increase the design or manufacturing complexity as we use uniform and rectangular chiplets.

More of SPCL's research:

youtube.com/@spcl — 180+ Talks

twitter.com/spcl_eth — 1.4K+ Followers

github.com/spcl — 2K+ Stars

… or spcl.ethz.ch