Maximum scaling exponent for Fast Fourier Transform

Dr. Anando Gopal Chatterjee - 20/12/2024

Outline

- Background
- MPI Alltoall
- Scaling Studies
- Increase in Computation and Communication efficiency
- Network Topology

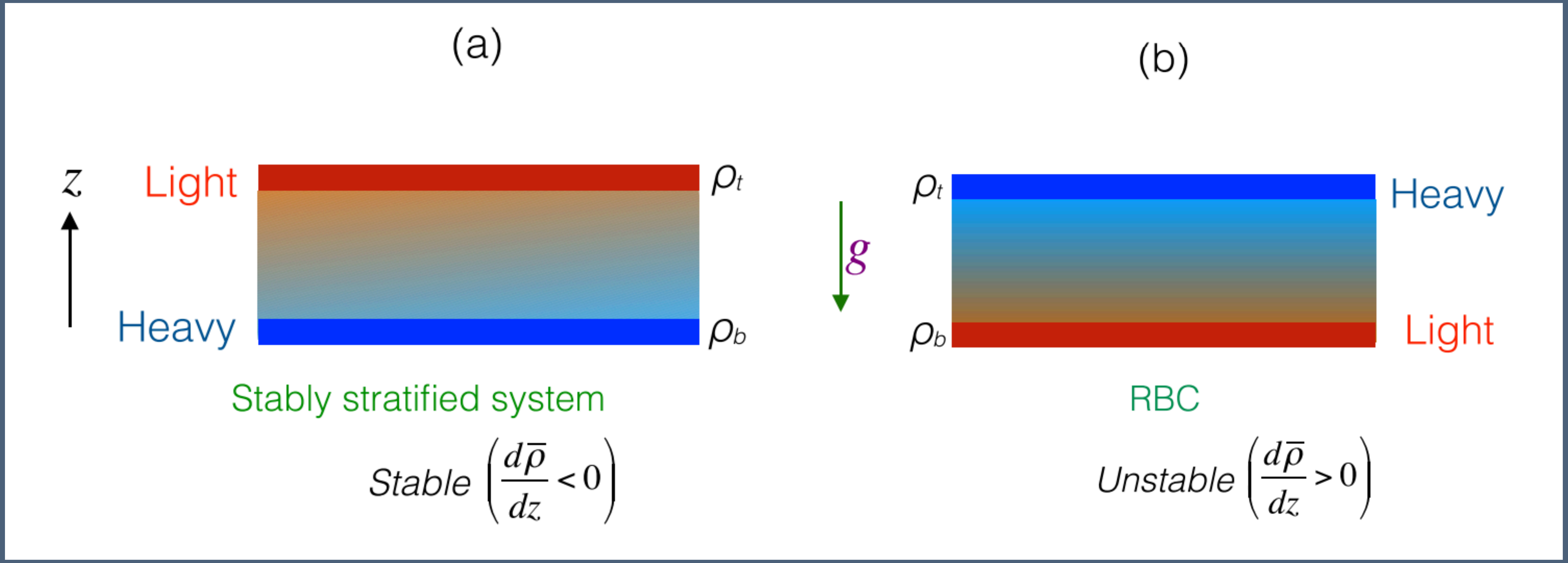
Background

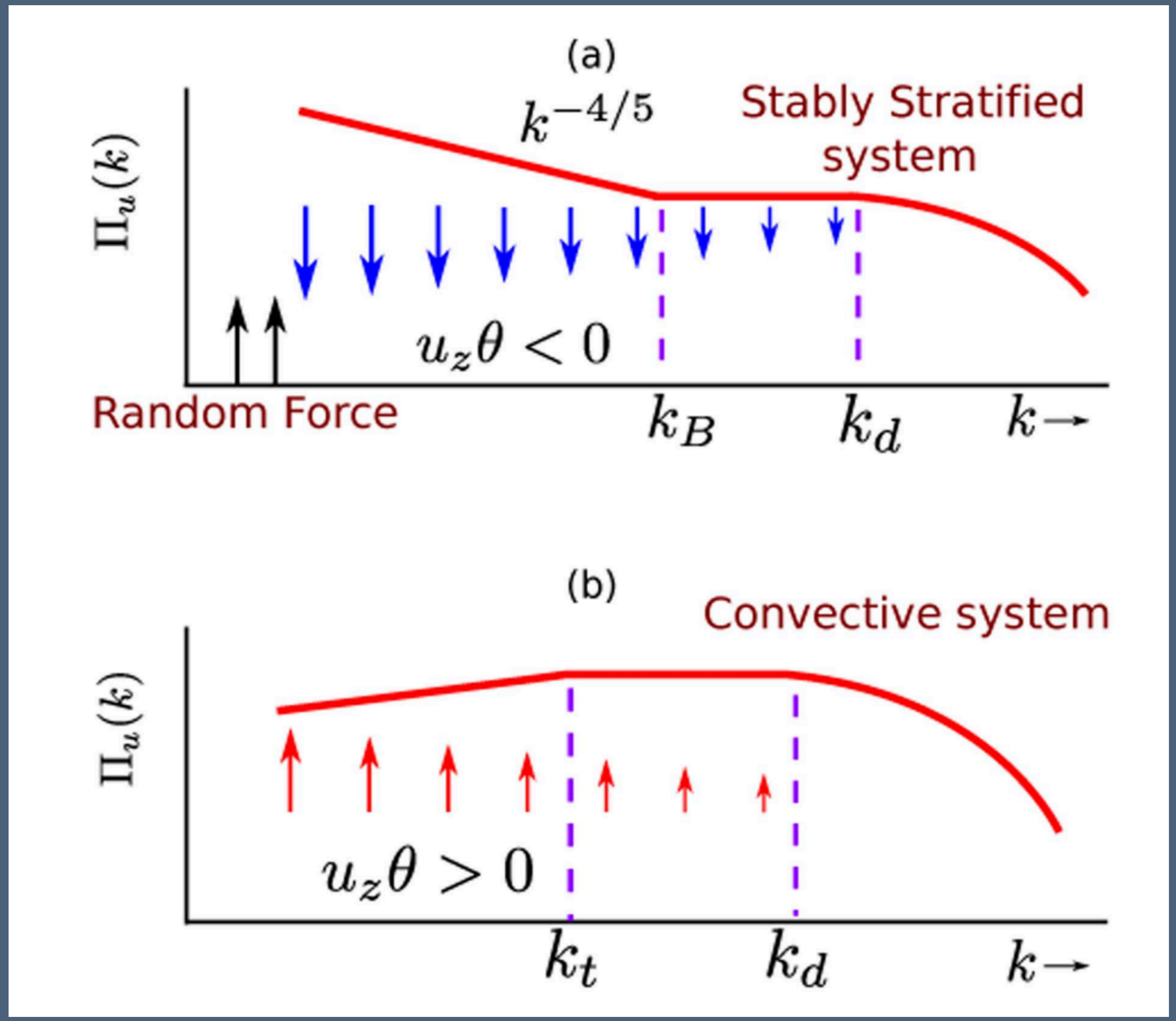
Why FFT

- Mobile devices receive 1000s of frequencies from various towers and electronic devices such as microwave oven and other electronics.
- Image compression such as JPEG.
- Radio Astronomy.
- Securing Radio Wave attacks over countries.
- Solving Fluid Dynamics in Pseudo-Spectral Space.

Advantages of Pseudo-Spectral Studies

Kolmogorov's Equation: $E(k) = K_{K_0}\Pi^{2/3}k^{-5/3}$





There are two RBC and two Stably Stratified layers in the atmosphere

Resolved turbulent fluids have zero energy at highest modes

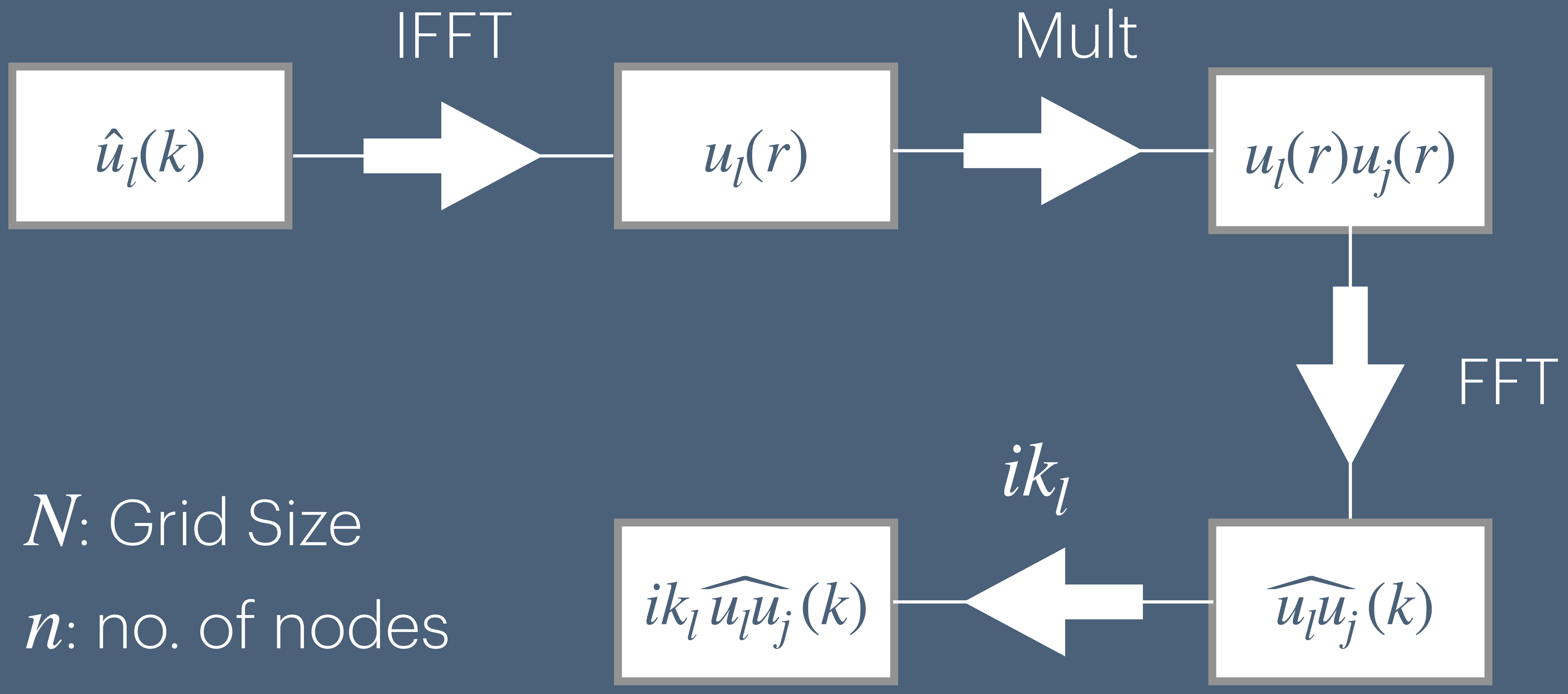
M. K. Verma, A. Kumar, and A. Pandey, *Phenomenology of buoyancy-driven turbulence: recent results*, New J. Phys., **19**, 025012 (2017).

Scaling of Fast Fourier Transform

- Pseudo-spectral method is widely used Fluid Dynamics due to it's High spatial accuracy.

$$\partial_t u_j(\mathbf{k}) = -ik_l \widehat{u_l u_j}(\mathbf{k}) - ik_j p(\mathbf{k}) - \nu k^2 u_j(\mathbf{k})$$

$$k_j u_j(\mathbf{k}) = 0$$



N: Grid Size

n: no. of nodes

In Spectral Approach:

Computations Required: N^3

In FFT Approach (pseudo-spectral):

Computations Required: $Nlog_2N$

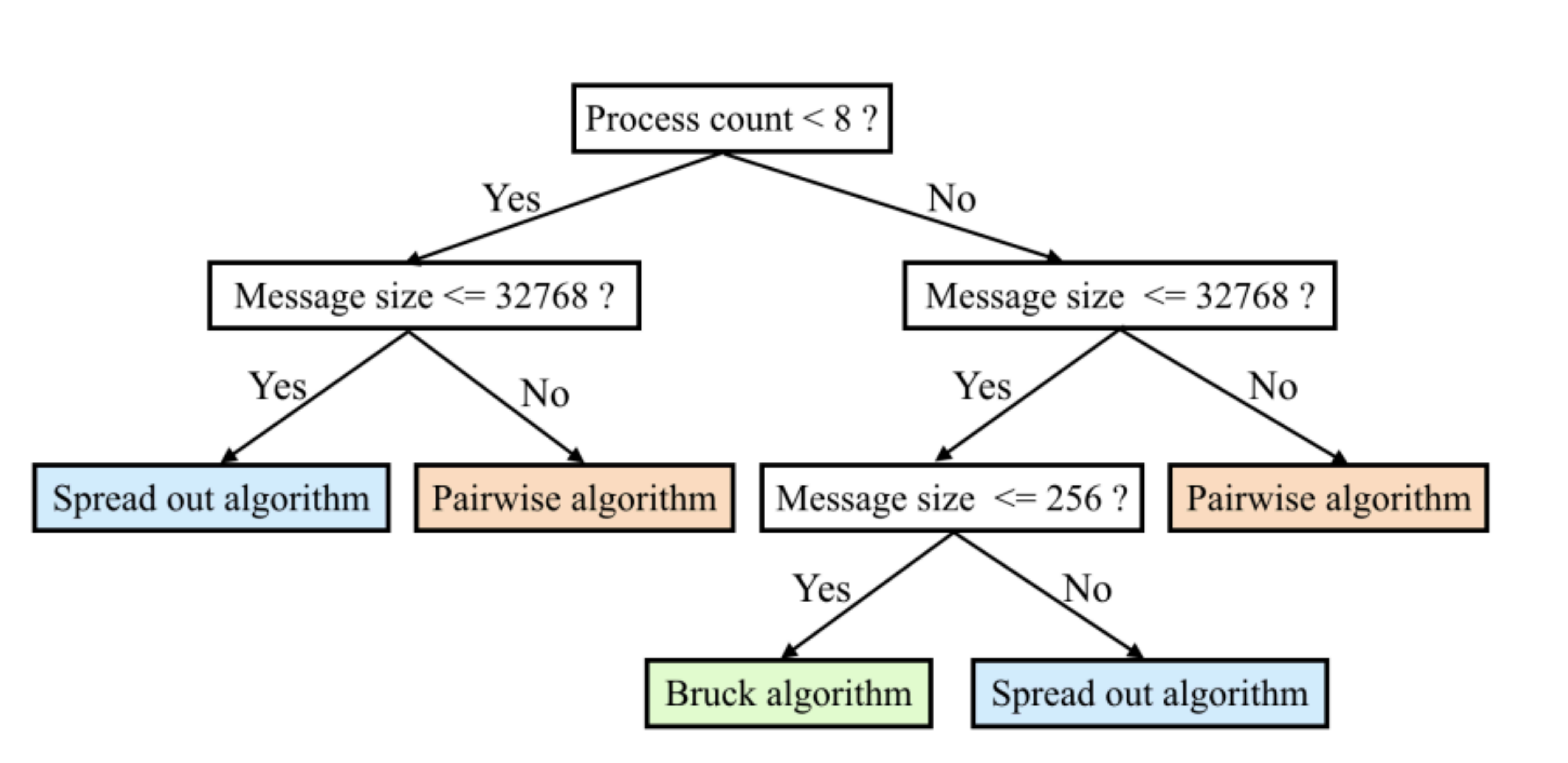
Communications Required: n^2

MPI Function: MPI_Alltoall

Performance of MPI functions

MPI Function	Relative Performance	Best For	Overhead
MPI_Send - MPI_Recv	Fast	Point-to-point data exchange	Low for small data
MPI_Bcast	Fast	Synchronizing single data point	Moderate
MPI_Scatter	Moderate	Distributing unique data to each process	Moderate
MPI_Gather	Moderate to High	Collecting data back to one process	High
MPI_Alltoall	Slow	Exchanging data between all processes	Very High
MPI_Allreduce	Fast	Aggregating results across processes	Low to Moderate
MPI_Reduce	Moderate	Summarizing results to one process	Moderate

MPI Alltoall

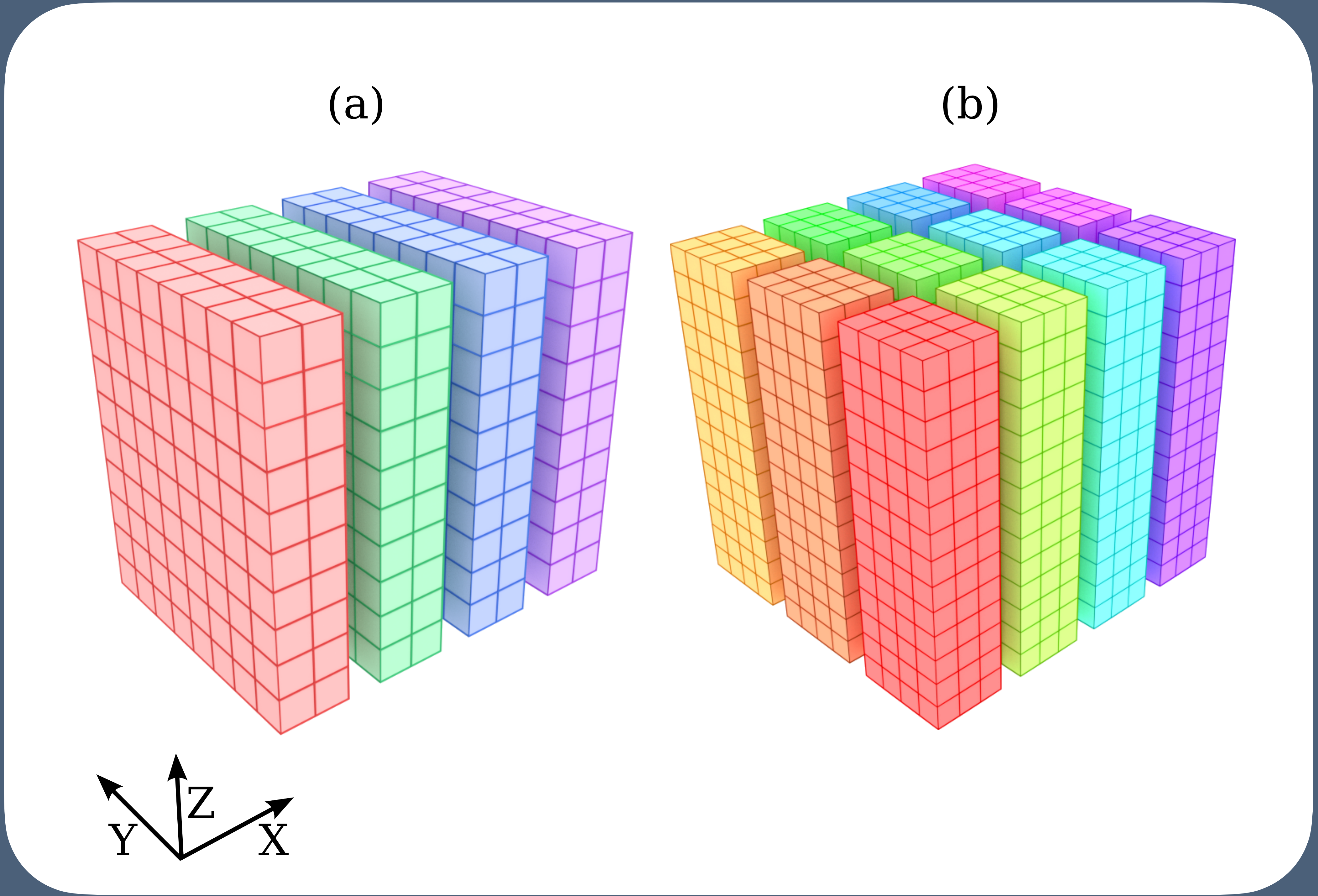


- Bruck algorithm
  $log_2 n$ Communications
- Spread out Algorithm
  n Communications
- Pairwise Algorithm
  n^2 Communicatuions

N. Netterville, K. Fan, S. Kumar and T. Gilray, "A Visual Guide to MPI All-to-all," 2022 IEEE 29th International Conference on High Performance Computing, Data and Analytics Workshop (HiPCW), Bengaluru, India, 2022, pp. 20-27 doi: 10.1109/HiPCW57629.2022.00008.

Scaling studies on Shaheen, KAUST

Data Parallelism



We have developed an FFT library named FFTK (FFT Kanpur)

We use FFTW for 1D Transforms

Since energy in high modes are zero,
we ignore the last mode in complex plane $(N/2+1)^{\text{th}}$
and use MPI_Alltoall for 2D/3D decomposition

Young Researcher Award by InSc (2023)

$$T = \frac{N}{B} = c_1 N \left(\frac{1}{p^{\gamma_1}}\right) + c_2 N \left(\frac{1}{n^{\gamma_2}}\right)$$

A. G. Chatterjee, M. K. Verma, A. Kumar, R. Samtaney, B. Hadri, and R. Khurram,
Scaling of a Fast Fourier Transform and a pseudo-spectral fluid solver up to 196608 cores, J. Parallel Distrib. Comput., 113, 77-91 (2017)

Strong Scaling

- A metric used to measure how the speedup of a program changes when the number of processors increases, while the problem size remains the same.

- When the number of cores/process (p) double computation time should come to half, but in practice it increases by a power law.

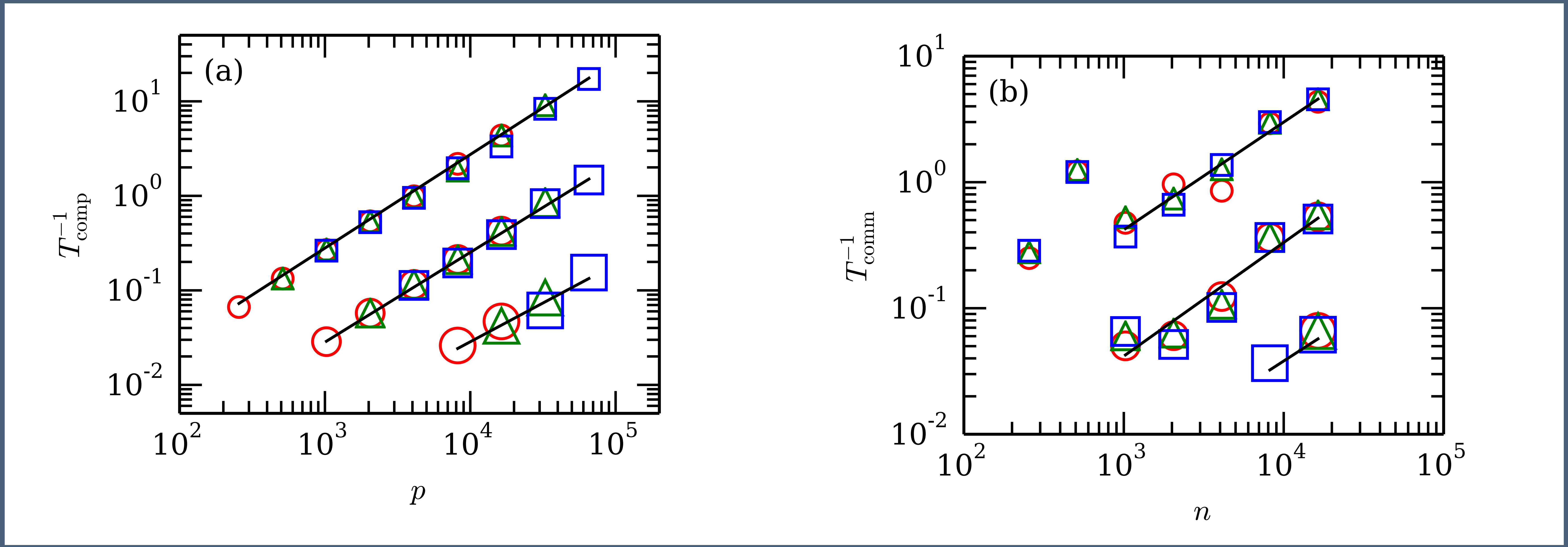
- When the number of nodes (n) double communication time should come to half, but in practice it increases by a power law.

$$T = T_{comp} + T_{comm}$$

$$T = c_1 N \left(\frac{1}{p^{\gamma_1}} \right) + c_2 N \left(\frac{1}{n^{\gamma_2}} \right)$$

Scaling on Bluegene-P up to 65,536 cores

Shaheen-I, KAUST, SA



$$T = c_1 N \left(\frac{1}{p^{\gamma_1}} \right) + c_2 N \left(\frac{1}{n^{\gamma_2}} \right)$$

$$\gamma^1 = 0.96 \quad \gamma^2 = 0.8$$

$$\text{Efficiency} = \frac{\text{Sustained GFLOPS/core}}{\text{Theoretical GFLOPS/core}}$$

$$= 10\,\%$$

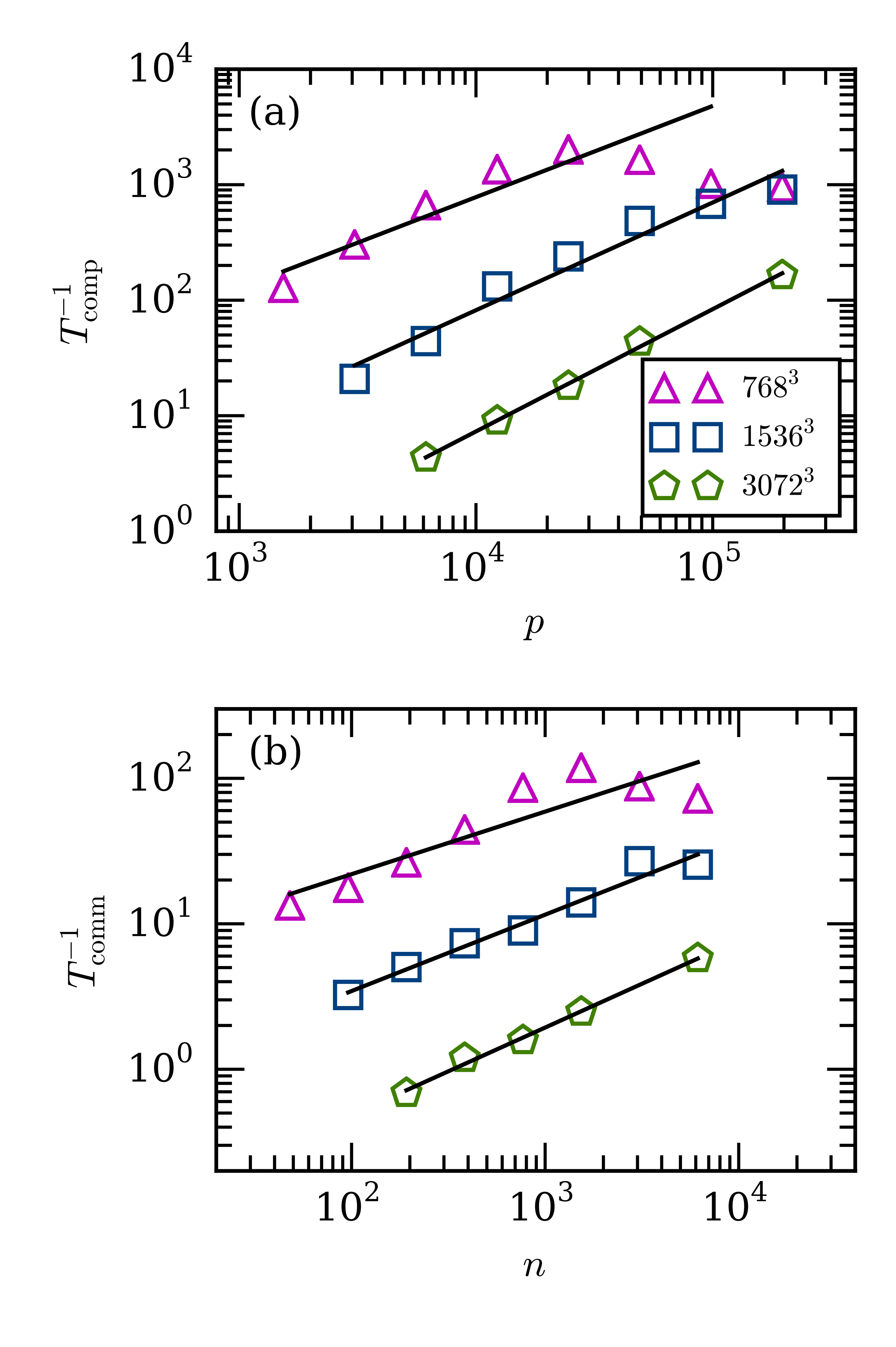
For communication intensive algorithms, like FFT, the cores wait for 90% of time for data to come.

Scaling on CRAY-XC40 up to 1,96,608 cores

Shaheen, KAUST, SA

$$T = c_1 N\left(\frac{1}{p^{\gamma_1}}\right) + c_2 N\left(\frac{1}{n^{\gamma_2}}\right)$$

$\gamma^1 = 0.97$ $\gamma^2 = 0.63$



$$\text{Efficiency} = \frac{\text{Sustained GFLOPS/core}}{\text{Theoretical GFLOPS/core}}$$

$= 2\,\%$

Increase in compute efficiency

Computation Speed

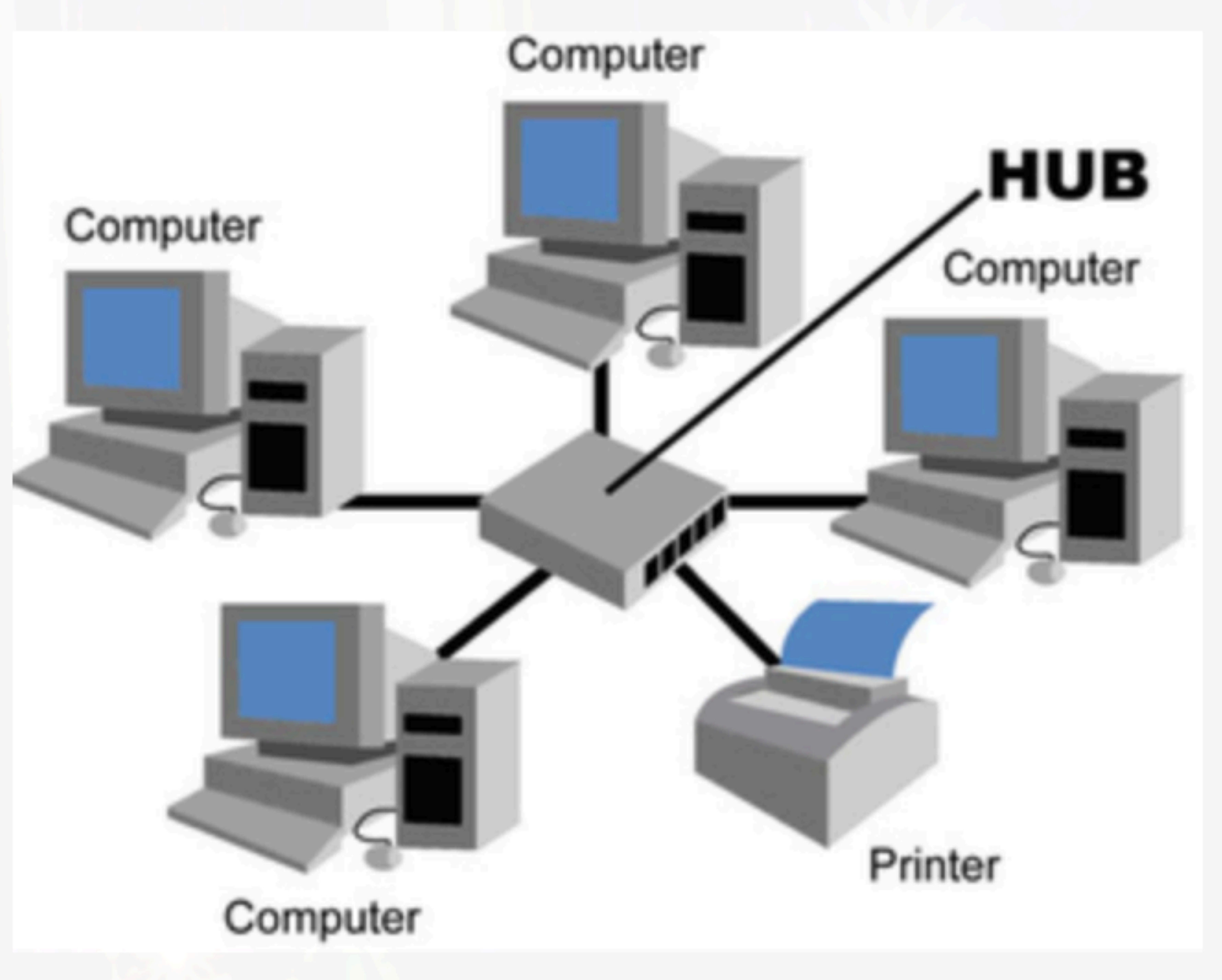
- Computers started with kiloFLOPS speed in 1970s
- Reached gigaFLOPS in 1990s
- The modern computers operate at around 20 gigaFLOPS
- Many such units are bundles together to form SuperComputers that have reached exaFLOPS in recent time.
- Moore's Law
- Compute power of top super computer (EL CAPITAN) $= 1.74 \times 10^{18}$

Communication Speed

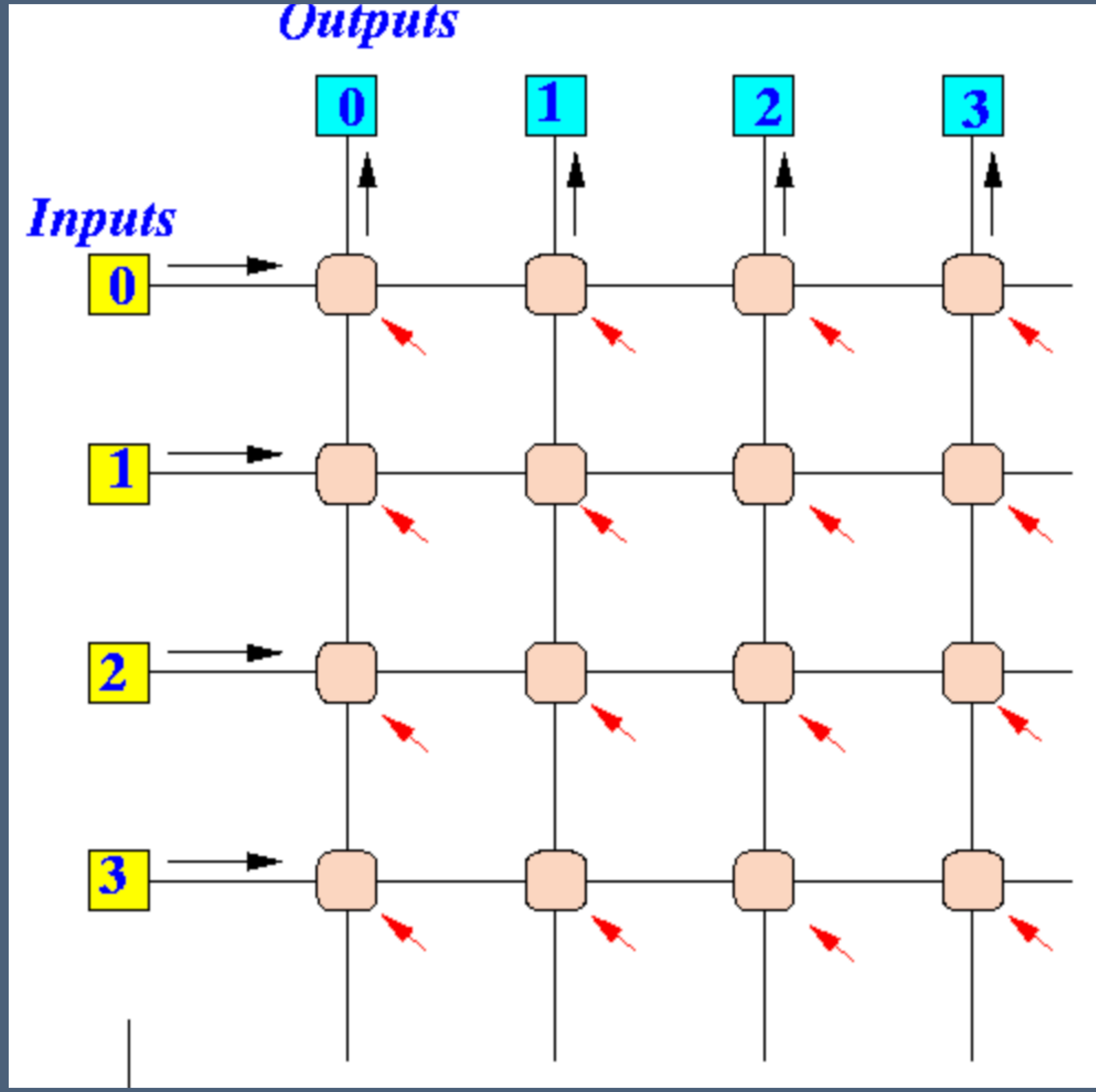
- Network Switch Speed has evolved from
  - kilobits in 1970s, to
  - 100 gigabits/second in 2010 (up to petaFLOPS computing), and
  - 800 gigabits/second in 2020 for exaFLOPS computing
- Typical Switches have 200 gigabits/second speed
- 200 gigabits/second = 25 gigabytes/second
- Actual Global Bandwidth depends on Topology
- Worst case Global Bandwidth is known as the bisection bandwidth

Network Topology

Star Topology



Regular routers/switches

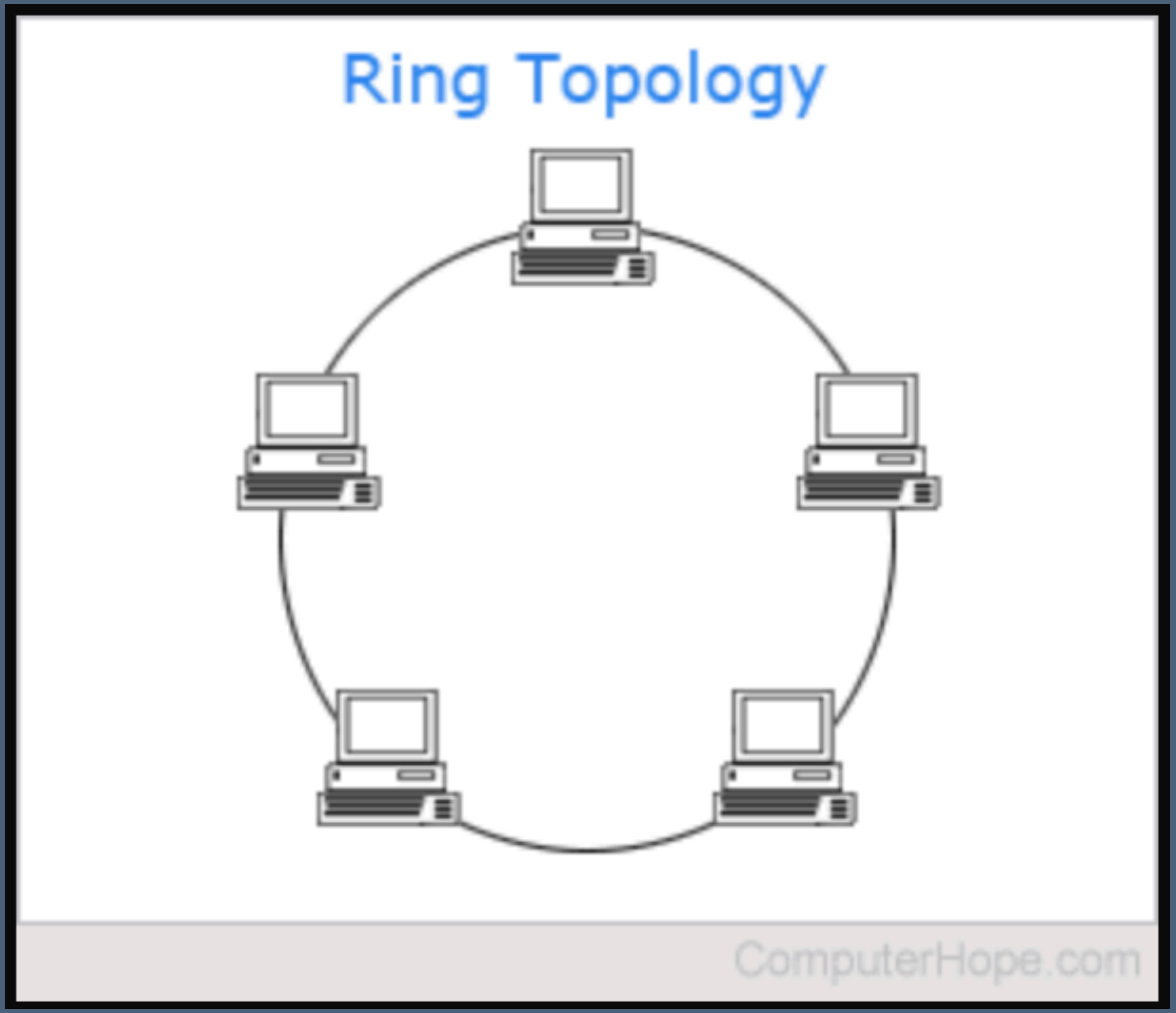


Crossbar technology

Bisection width (B_w) = number of nodes (n)

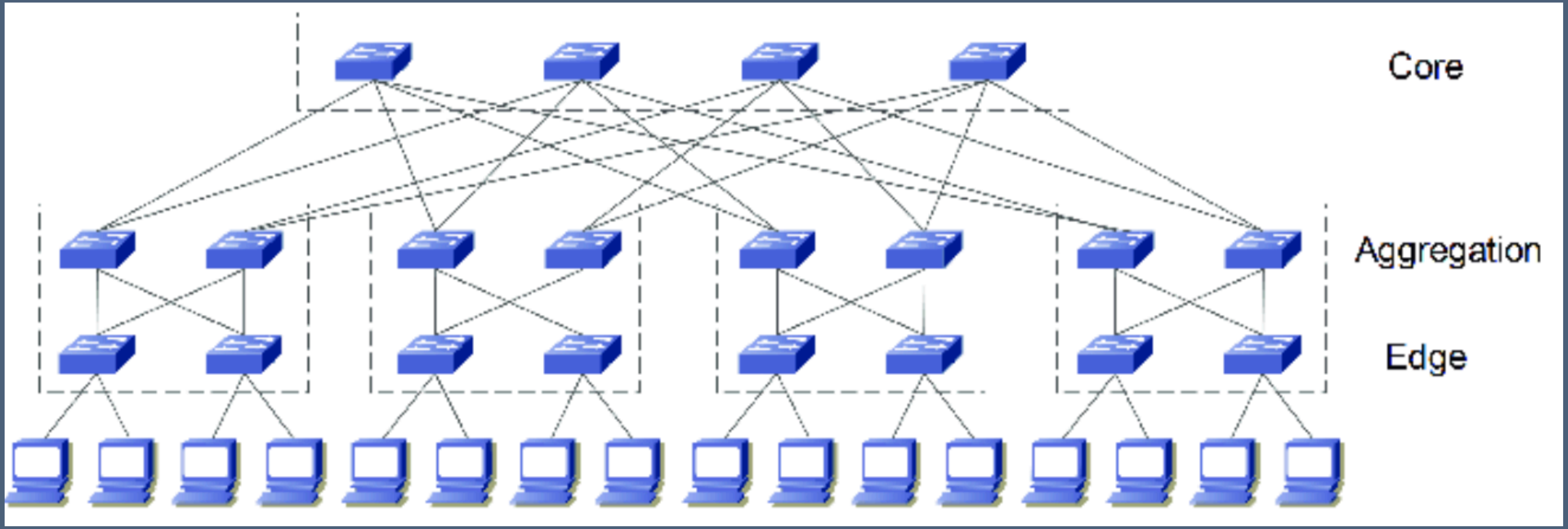
Image Source: https://everythingaboutcomputernetworks.weebly.com/star-topology.html
Image Source: https://www.cs.emory.edu/~cheung/Courses/355/Syllabus/90-parallel/CrossBar.html

Ring Topology



- Typically used in office spaces
- This can provide internet connection even if one connection fails
- $B_w = 2$

Fat Tree Topology



- All switches have same number of ports

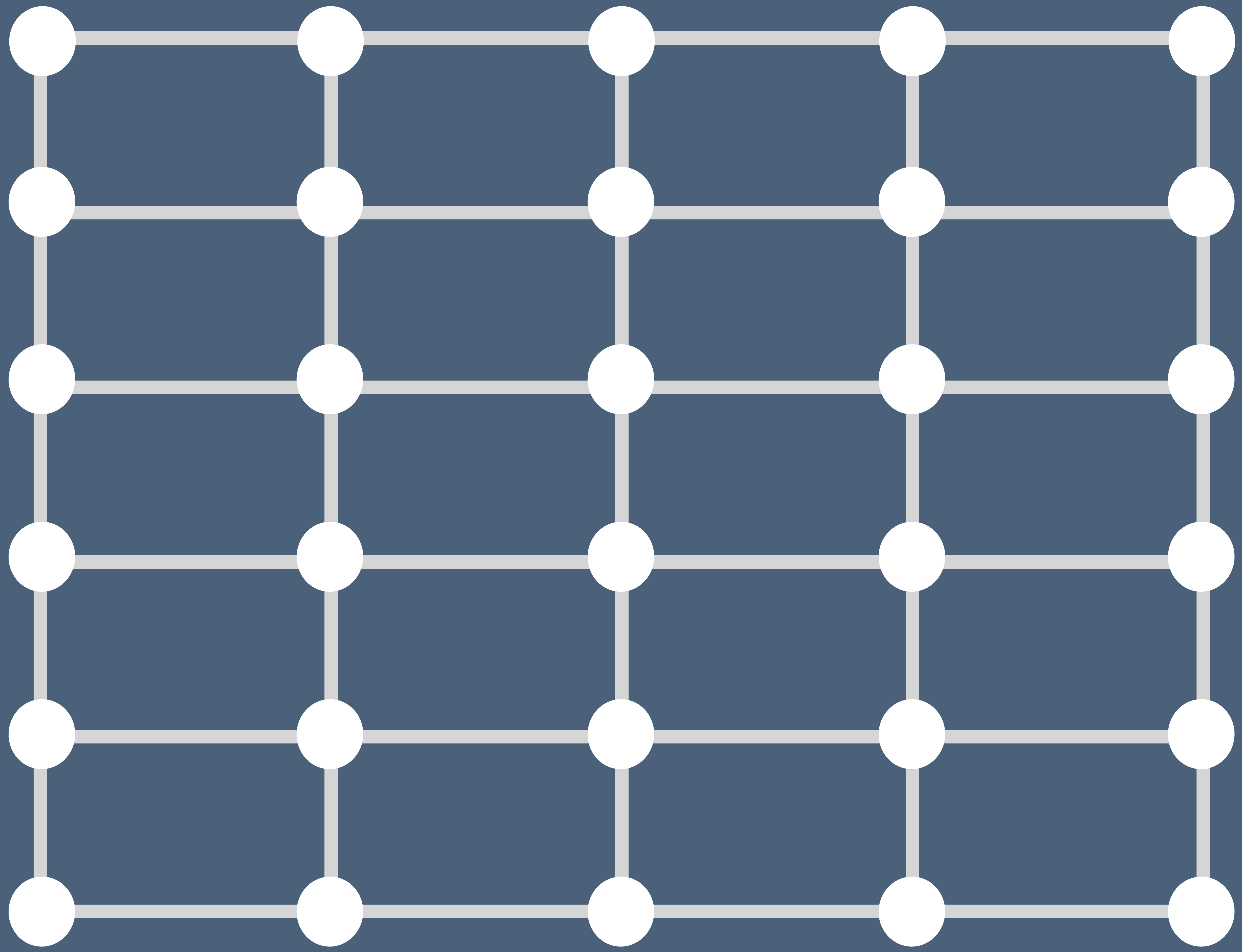
$$B_w = n/2$$

- Typically 32/64 port switches are used

- Most high-performance supercomputers on the Top 500 list, including recent leaders like Summit and Sierra, use a fat-tree network due to its high bandwidth and scalability

Image Source: https://networkengineering.stackexchange.com/questions/67729/fat-tree-topology

Mesh Topology



n	B_w
1	0
2	1
4	2
9	3
16	4
25	5

$$B_w = \sqrt{n}$$

3D Torus - Bluegene/P

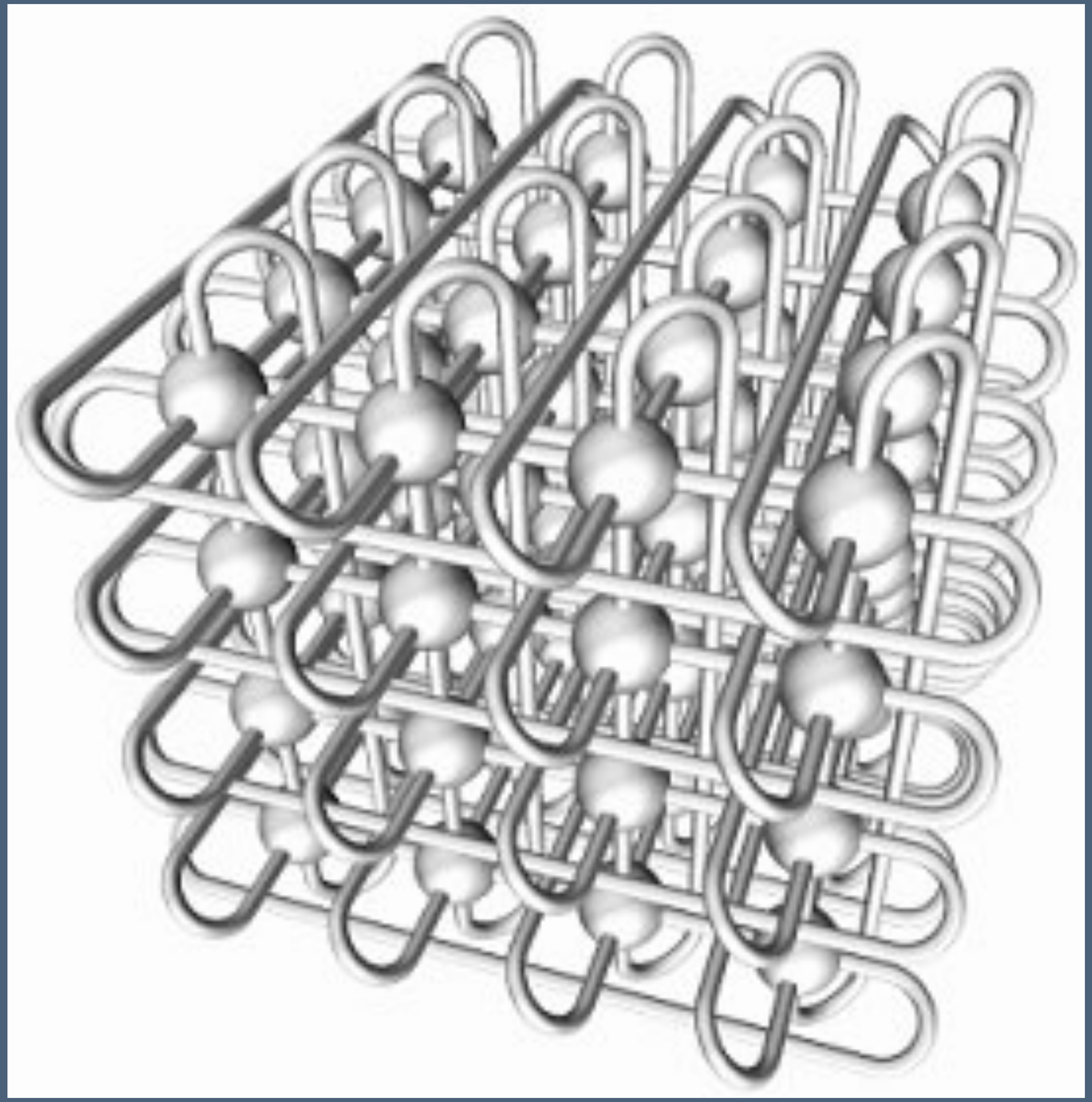


Image Source: https://stackoverflow.com/q/66723243/1525392

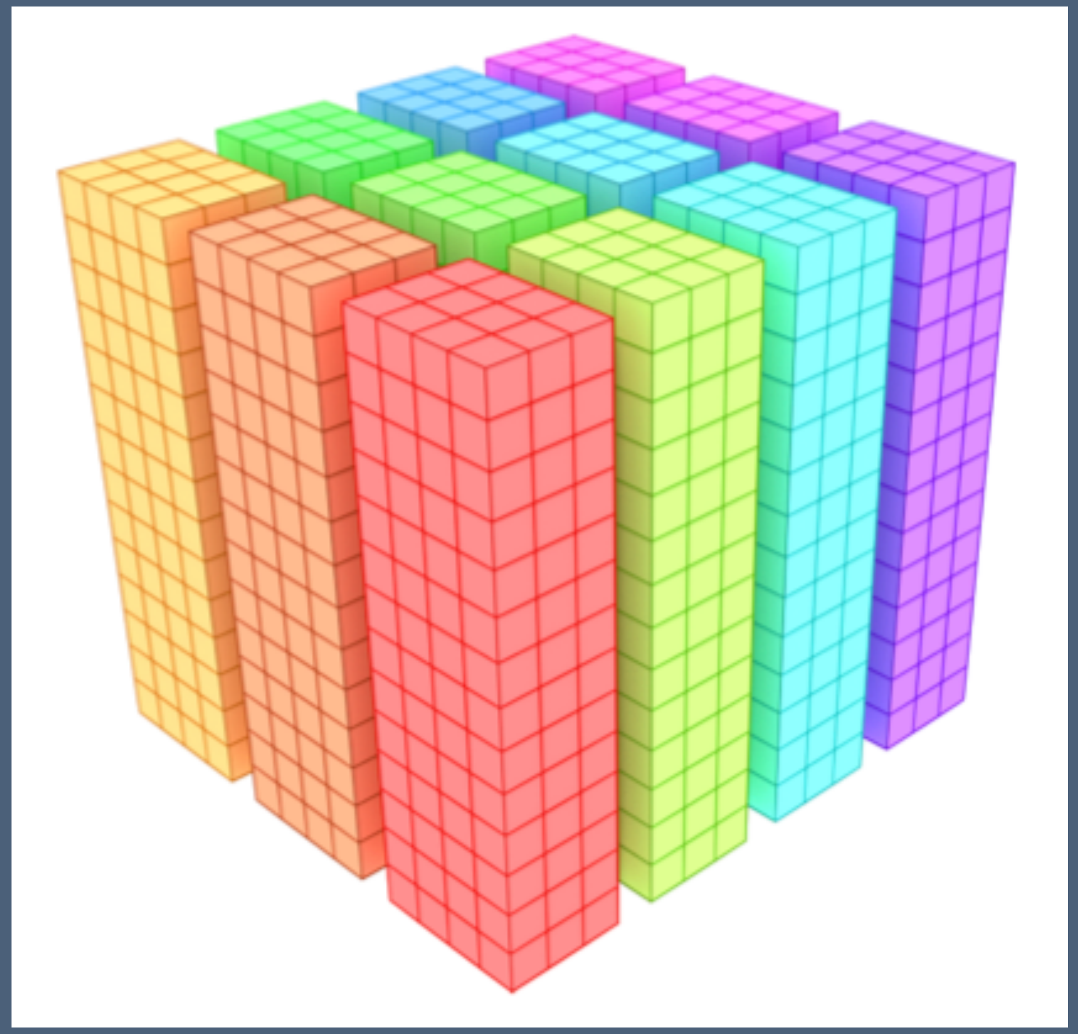
Bisection Bandwidth is proportional to the area

$$B \propto (n')^{2/3}$$

For Pencil Decomposition
square root of n nodes interact ant a time

$$n' = n^{1/2}$$

$$B \propto n^{1/3}$$



A. G. Chatterjee, M. K. Verma, A. Kumar, R. Samtaney, B. Hadri, and R. Khurram, *Scaling of a Fast Fourier Transform and a pseudo-spectral fluid solver up to 196608 cores*, J. Parallel Distrib. Comput.,**113**, 77 (2018)

Pencil Decomposition

Data per node $\propto N^3/n$

Data in wire: $D \propto N^3/n \cdot n^{1/2}$

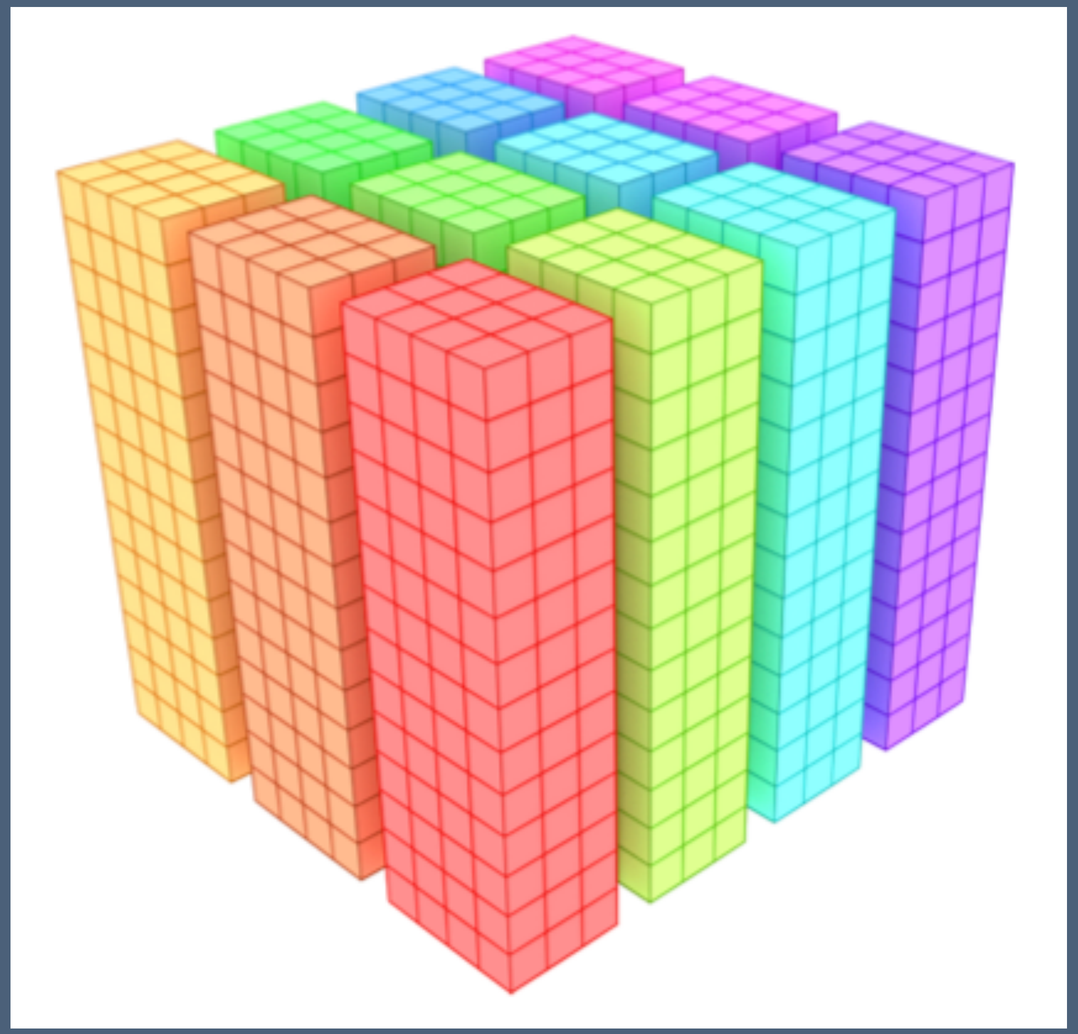
$$\propto N^3/n^{1/2}$$

By Definition $B = D/T_c$

$$T_c = D/B$$

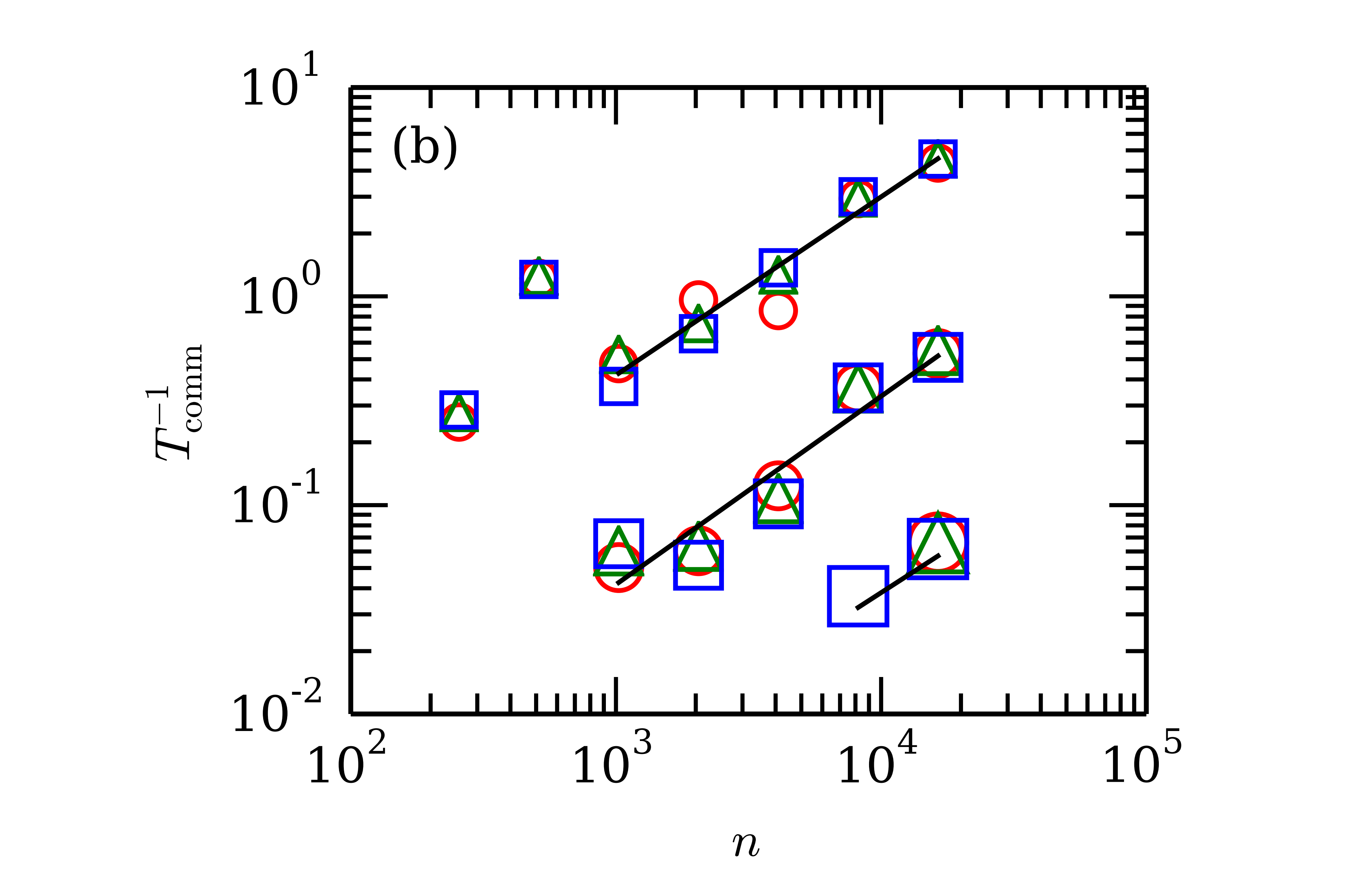
$$= \frac{N^3}{n^{1/2}} \cdot \frac{1}{n^{1/3}} = N^3 \cdot n^{6/5}$$

$$T_c \propto \frac{1}{n^{5/6}} \approx \frac{1}{n^{0.83}}$$



Communication Scaling on Bluegene-P

Shaheen-I, KAUST, SA



From FFTK Scaling: $T_c \propto \left(\frac{1}{n^{0.8}}\right)$

From Bisection Width Calculations: $T_c \propto \left(\frac{1}{n^{0.83}}\right)$

Earth Simulator: 2002–2009

JAMSTEC Yokohama Institute for Earth Sciences

- Earth Simulator Top-ranked the Global FFT at HPC Challenge Awards in 2010
- 640 x 640 single stage crossbar (Star topology)

Press Release: https://www.jamstec.go.jp/e/about/press_release/archive/2010/20101117.pdf

Conclusion

- By definition, Bisection Bandwidth is the worst-case performance metric.
- If we schedule on all/most of a supercomputer, we would not get a communication scaling better than Bisection Bandwidth scaling of it’s topology.
- If we schedule on small part of supercomputer we may get better communication scaling.
- For a Bluegene-P Supercomputer we have shown that upon simulating on the full supercomputer, communication scaling exactly matches it’s Bisection Bandwidth.

Thank you for your attention