UNIVERSITÀ
DEGLI STUDI
DI TRIESTE

Cloud Computing

Final Exam Exercise

Giovanni Lucarelli

May 17, 2025

Introduction

Goal

1. **Build** a cluster of virtual nodes as:
   - Virtual Machines $\rightarrow$ VirtualBox
   - Containers $\rightarrow$ Docker
2. **Assess** and **compare** the performance:
   - CPU
   - Memory
   - Disk I/O
   - Network

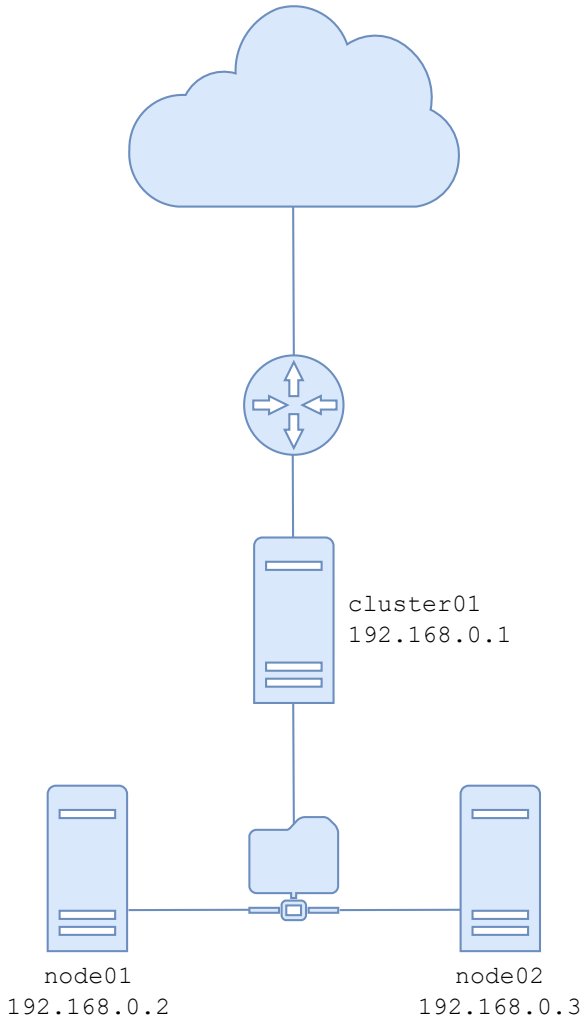
(Virtual) Hardware Specification

Host Machine:

CPU	Intel Core i7-8550U CPU @ 1.80GHz, 4 Cores / 8 Threads
Memory	8 GB
Disk	256 GB SSD
OS	Ubuntu 24.04.2 LTS

Cluster Nodes:

CPU	2 Cores
Memory	2048 MB
Disk	20 GB
OS	Ubuntu 22.04.5 live server



Methodology

Virtual Machine Setup

1. Build a template machine (hardware & software) & clone (reinitializing the MAC)
2. Network Adapters (VirtualBox GUI):
   - 2.1 NAT, internal net
   - 2.2 Port Forwarding and SSH (Host→Master)
3. Master Node:
   - 3.1 `etc/hostname`, `etc/hosts`
   - 3.2 DHCP, DNS, gateway
   - 3.3 Shared file system (NFS)
4. Worker Node:
   - 4.1 `etc/hostname`
   - 4.2 SSH (Master→Worker)
   - 4.3 Shared file system (NFS)

Container Setup (1/2)

1. Build a template machine (Dockerfile)

```
# Download the latest official Ubuntu image
FROM ubuntu:latest

# Update and install all the required software
RUN apt-get update && apt-get install -y \
...
# Expose the SSH port
EXPOSE 22
```

Container Setup (2/2)

2. Build a cluster (Docker Compose)

```
services:
  cluster01:
    ...
  node01:
    ...
  node02:
    ...
networks:
  internal-
net:
    driver: bridge
volumes:
  shared-
data:
    driver: local
```

```
cluster01:
  build: .
  container_name: cluster01
  hostname: cluster01
  networks:
    internal-net:
  deploy:
    resources:
      limits:
        cpus: "2"
        memory: 2G
  ports:
    - "2220:22"
  volumes:
    - shared-data:/shared
```

Benchmarks

hpcc	suite of tests to measure the performance of high-performance computing systems
stress-ng	stress-testing CPUs, memory and other components under heavy load
sysbench	evaluating system parameters such as CPU, memory, and other components
iozone	measure filesystem I/O performance
iperf	measure network performance

Assessing the Cluster

General guidelines:

- no heavy processes on the host during the tests
- repeat each test multiple times to account for variability ($1 \leq n_i \leq 5$)
- monitorate the host resources during the tests
- repeat the tests on the host*

Running the tests

1. `mpirun -np 4 -hostfile hosts <test>`

   File `hosts`:

   ```
   node01 slots=2
   node02 slots=2
   ```

2. bash-script to automatize multiple repetition of each test

Results

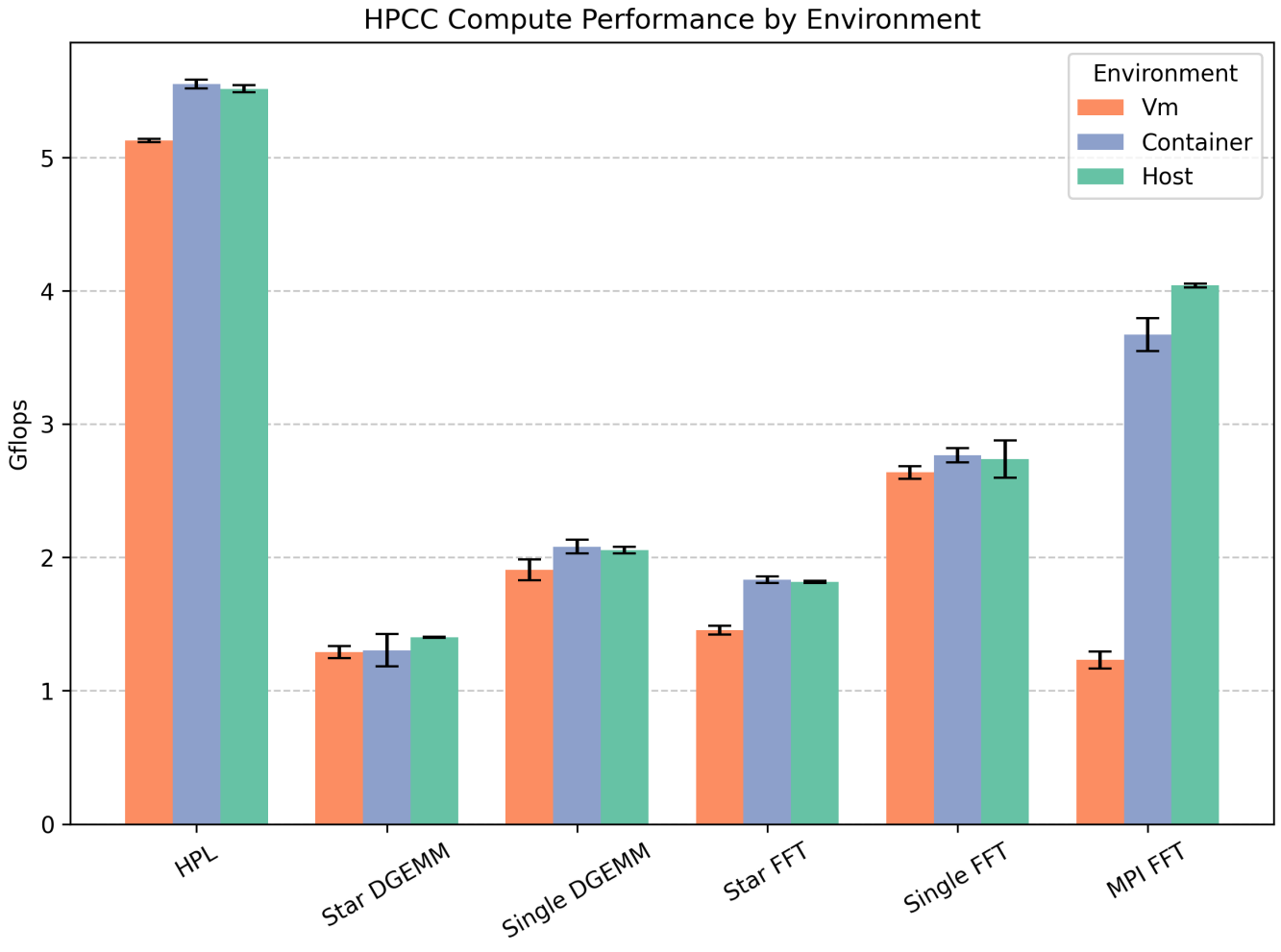
High Performance Computing Challenge (HPCC)

Number of repetition: **3**

Benchmark types:

- **Computational**: HPL, DGEMM, FFT
- **Memory**: STREAM, PTRANS, RandomAccess
- **Communication**: PingPong, (PTRANS)

HPCC: Computational Performance



HPCC: Nominal Memory Bandwidth

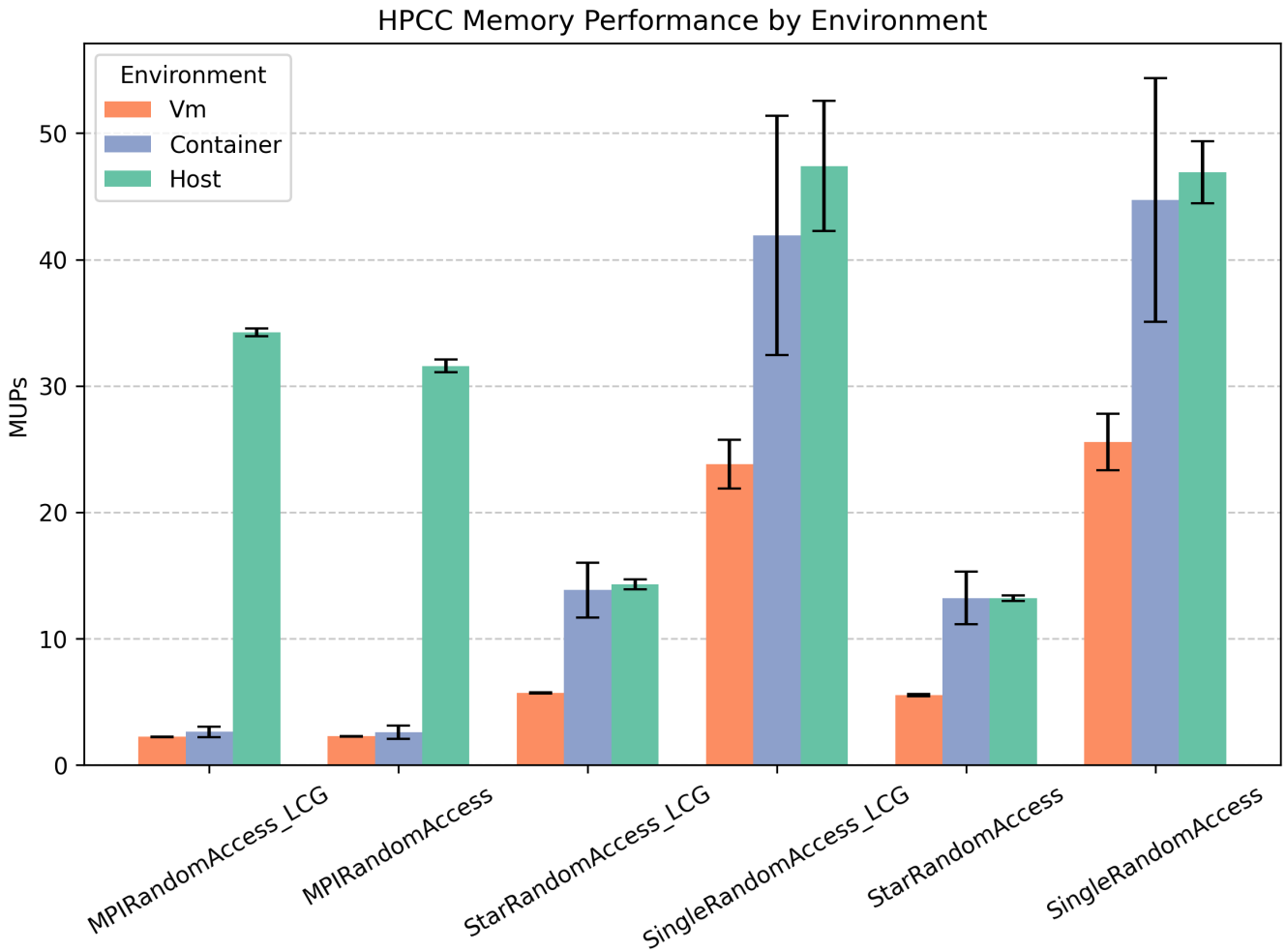
```
sudo dmidecode --type memory

Configured Memory Speed: 1867 MT/s
Bus width per channel: 64 bits = 8 bytes
Number of channels: 2
```

$$\text{Bandwidth}\left[\frac{\text{GB}}{\text{s}}\right] = 2\,\text{C} \times 1.867\,\frac{\text{GT}}{\text{s}} \times 8\,\frac{\text{B}}{\text{C}\,T} = 29.9\,\frac{\text{GB}}{\text{s}}$$

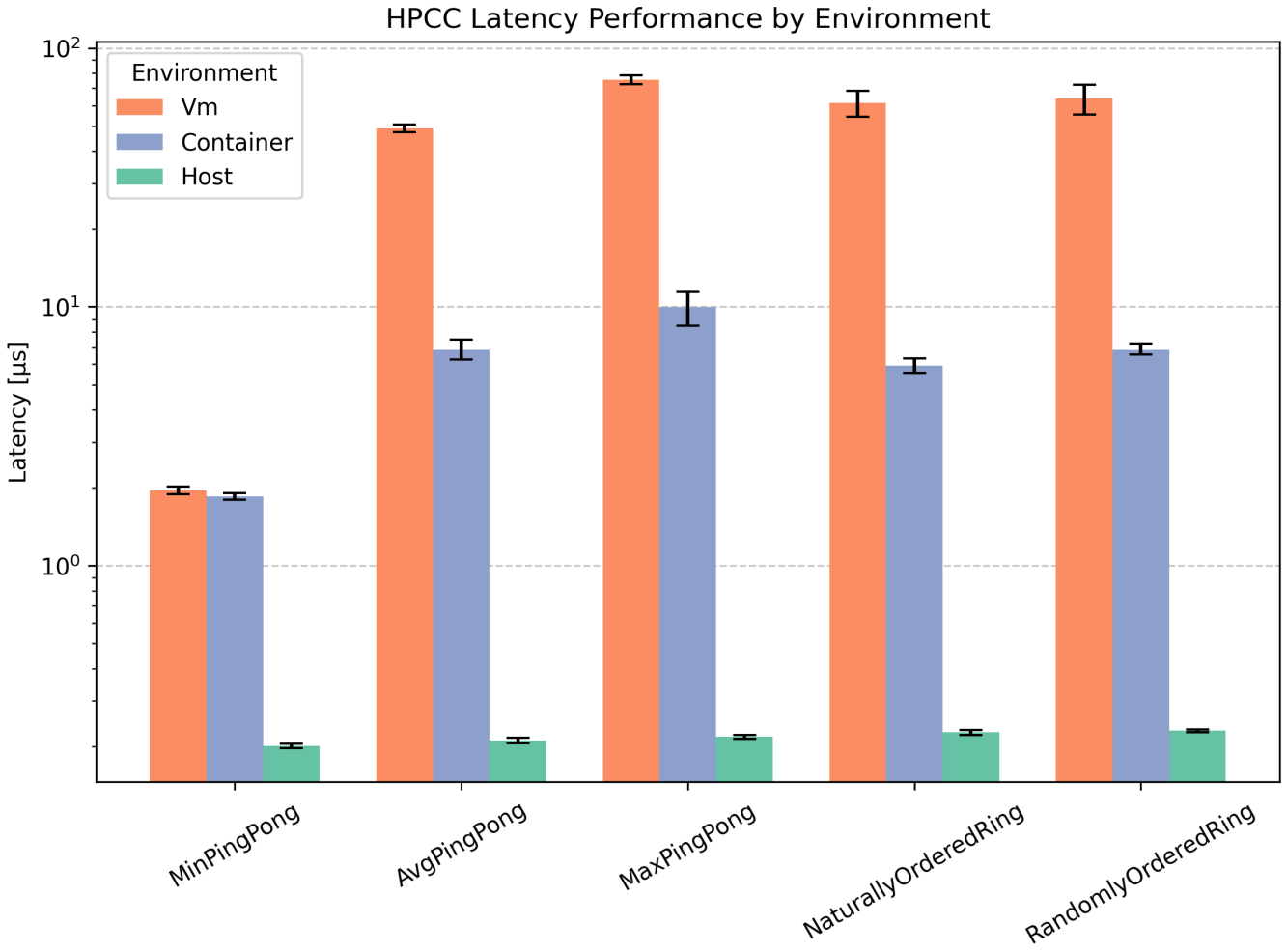
HPCC: Memory Performance (1/2)

Benchmark	VM	Container	Host
SingleSTREAM (GB/s)			
Copy	22.30 ± 0.32	**24.11 ± 0.20**	23.44 ± 0.06
Scale	13.26 ± 0.19	14.23 ± 0.06	14.06 ± 0.12
Add	14.40 ± 0.24	15.38 ± 0.16	15.06 ± 0.14
Triad	14.44 ± 0.28	15.48 ± 0.13	15.22 ± 0.05
StarSTREAM (GB/s)			
Copy	5.03 ± 0.03	**5.41 ± 0.03**	5.39 ± 0.02
Scale	3.34 ± 0.03	3.55 ± 0.01	3.56 ± 0.01
Add	3.75 ± 0.01	4.08 ± 0.02	4.07 ± 0.01
Triad	3.72 ± 0.04	4.02 ± 0.02	4.00 ± 0.02
PTRANS (GB/s)	0.196 ± 0.014	1.181 ± 0.239	**1.495 ± 0.019**

HPCC: Memory Performance (2/2)



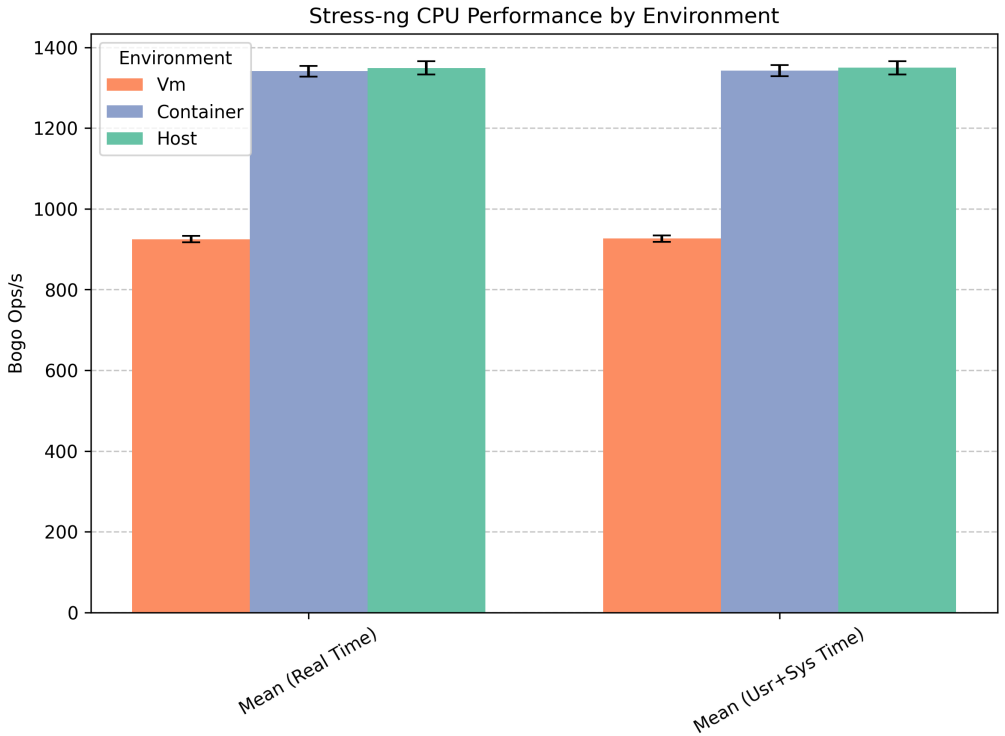
HPCC: Comunication Performance



Note the log scale!

Stress-ng: CPU

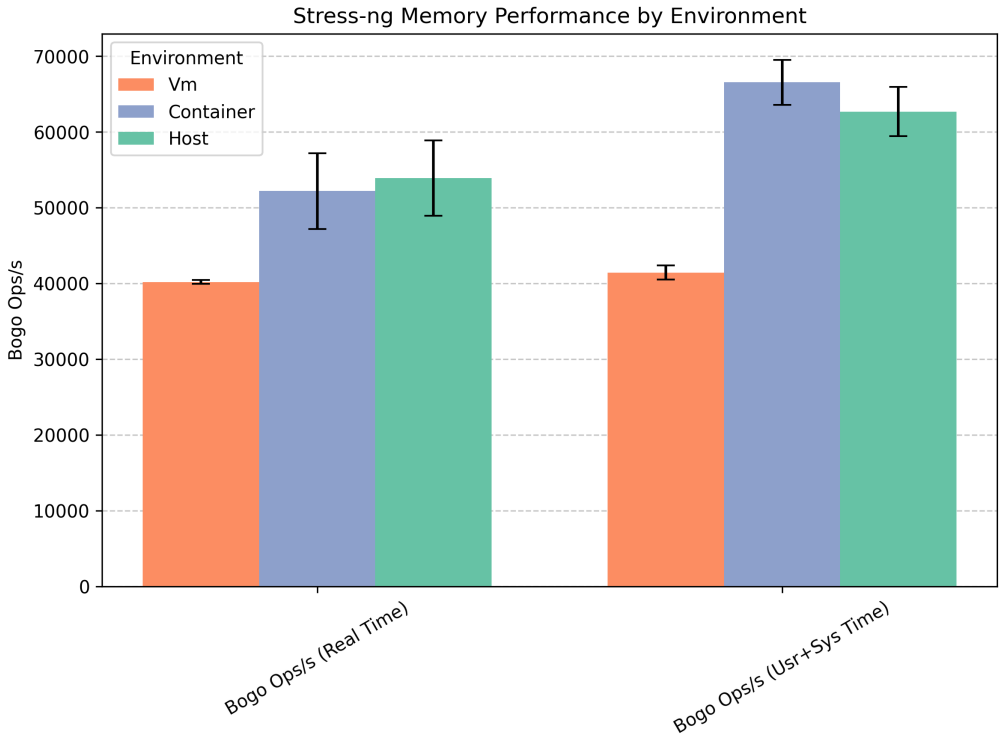
Repetitions: **5**



- `real` ≈ `usr + sys`: no significant waiting time
- VMs: fewer operations, higher CPU time per operation

Stress-ng: Memory

Repetitions: **5**



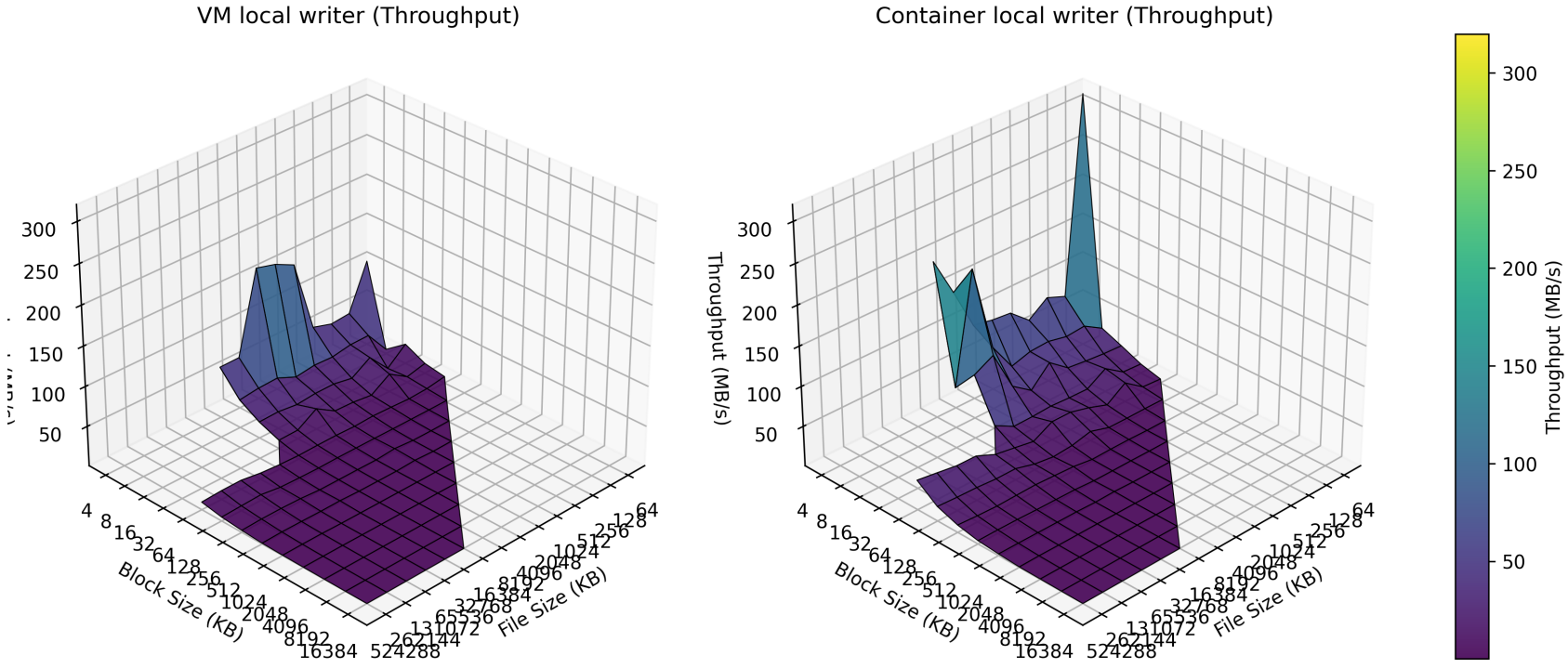
- Containers and Host: `real` > `usr + sys` → some waiting time
- VMs: `real` ≈ `usr + sys` → less waiting, but still higher CPU time per operation

Sysbench

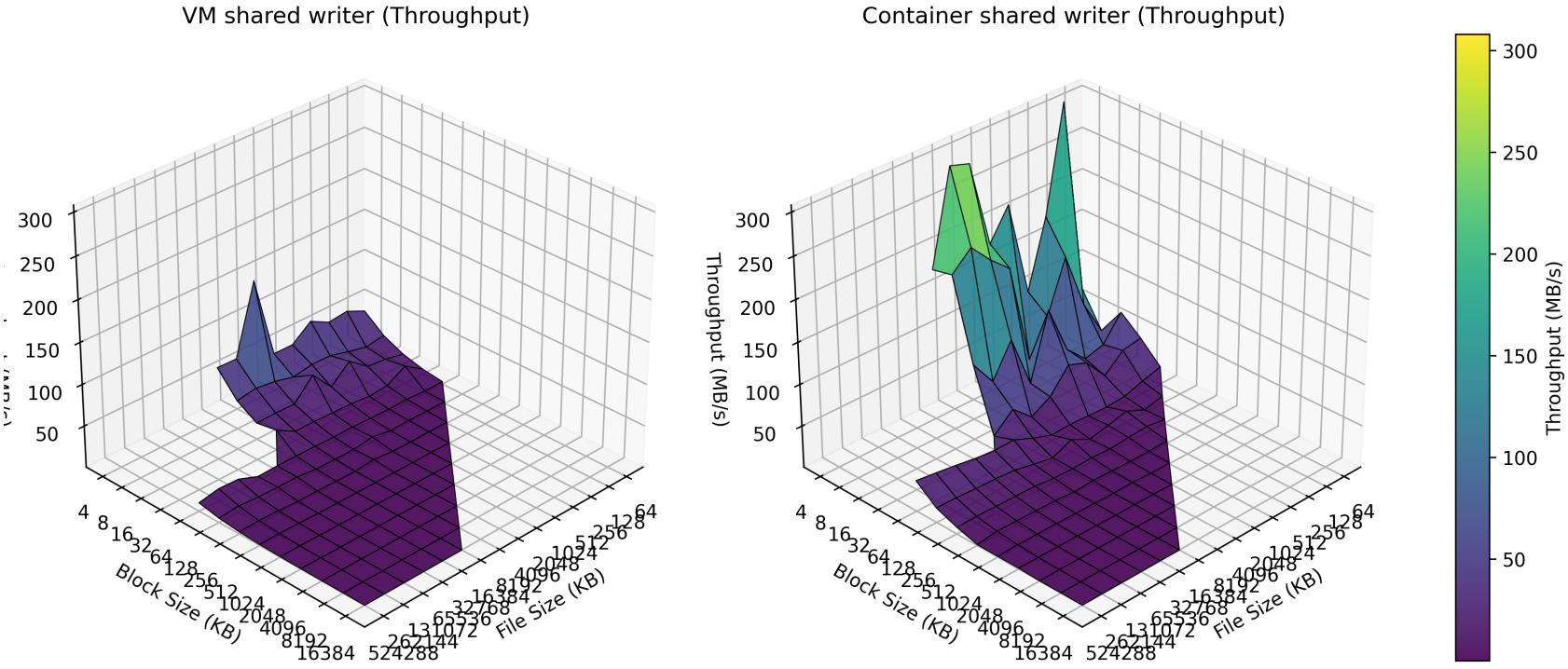
Number of repetition: **5**

Benchmark	VM	Container	Host
CPU			
Events/s	453.97 ± 1.28	459.74 ± 2.54	452.38 ± 6.76
Latency sum (ms)	9998.15 ± 0.70	9999.55 ± 0.51	9999.56 ± 0.72
Memory			
Throughput (Gib/s)	3.88 ± 0.02	**5.51 ± 0.09**	5.19 ± 0.08
Latency sum (ms)	1066.09 ± 8.01	**839.99 ± 14.63**	895.79 ± 18.40

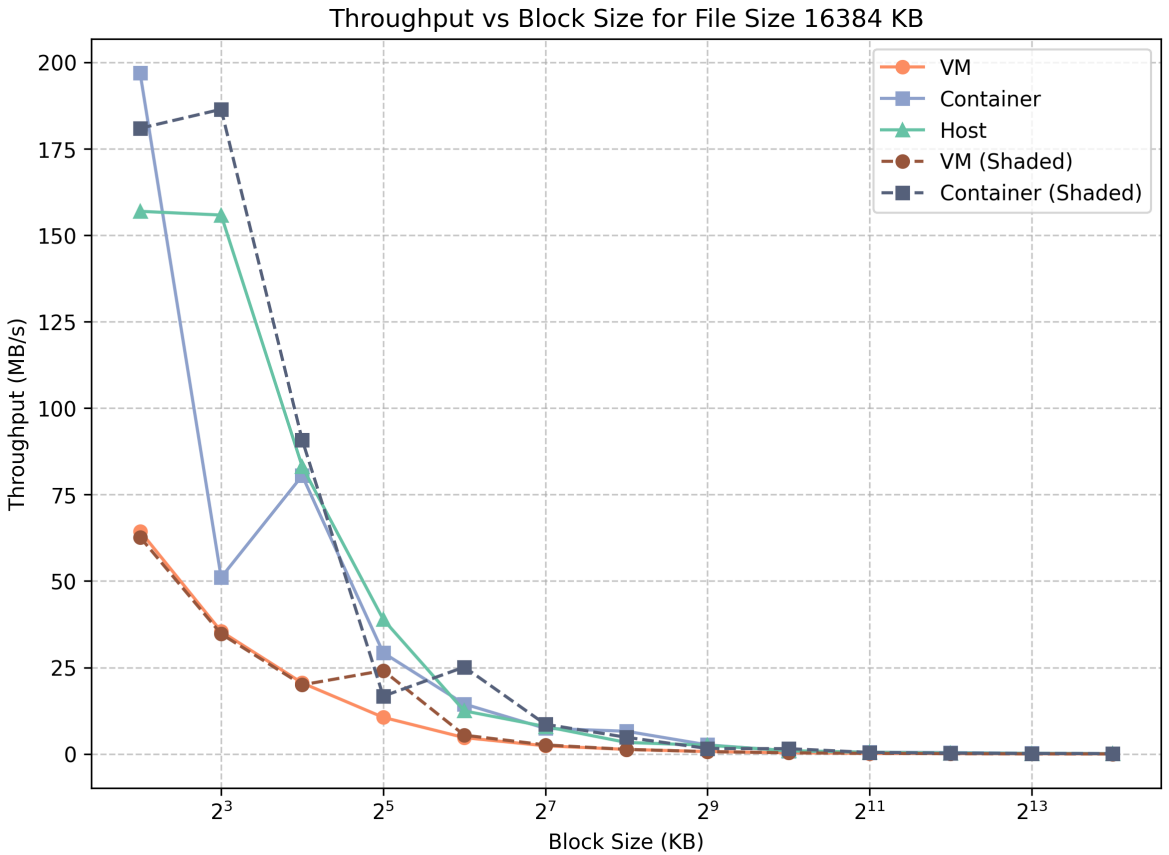
IOZone: write local



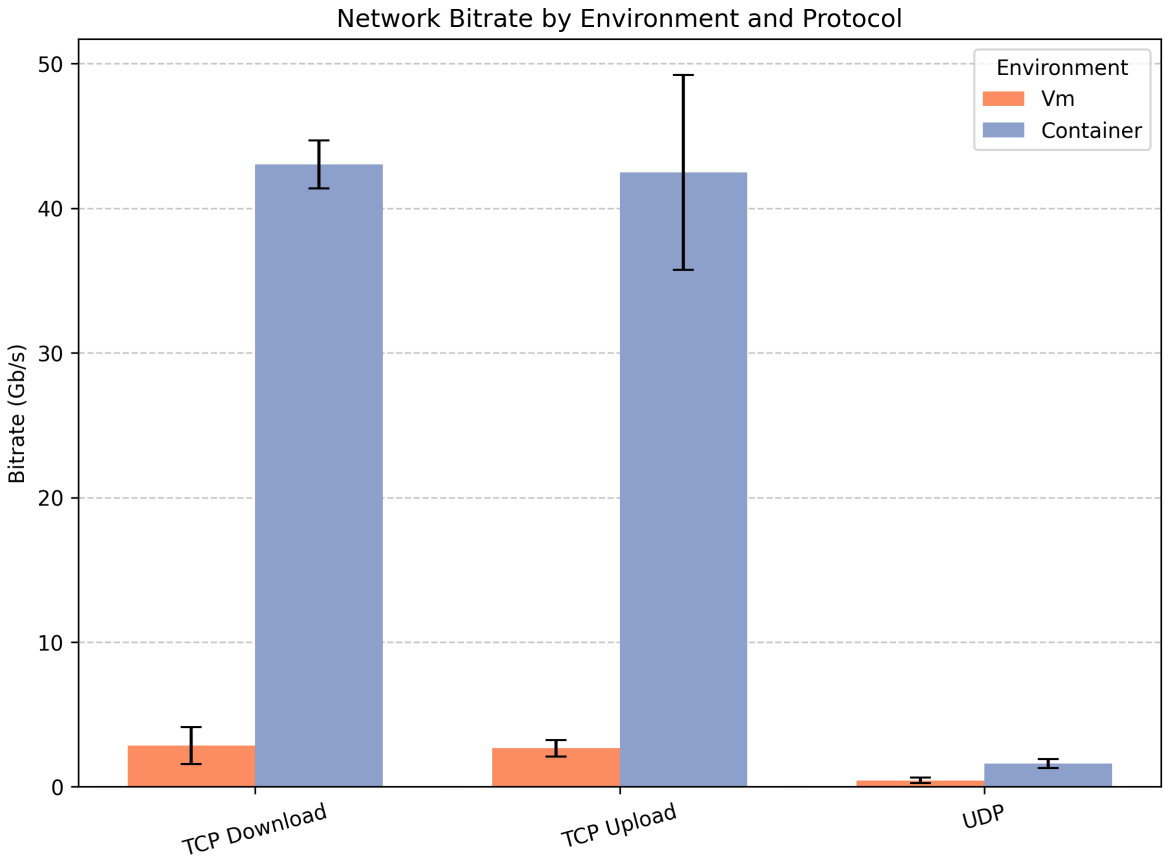
IOZone: write shared



IOZone: writing comparison



Iperf


Network Bitrate by Environment and Protocol
Environment
Vm
Container
Bitrate (Gb/s)
0
10
20
30
40
50
TCP Download
TCP Upload
UDP

Conclusion

Conclusion

Docker Cluster

- Easier to configure (shared kernel and lightweight setup)
- Near-native or better performance in CPU and memory benchmarks
- More scalable

VirtualBox Cluster

- More time-consuming to configure (multiple full OS instances)
- Higher overhead due to full hardware virtualization
- Lower memory bandwidth and network bitrate observed

Thank You!