Team names and Ids

Mohanad Mohamed Shebl :221001360

Malak Taymour: 221001266

Jana Elhadad: 211002077

Mini-HPC and Hybrid HPC-Big Data Clusters Project

**Introduction**

This project aimed to design, configure, and deploy two types of computing clusters using virtual machines (VMs): a traditional High-Performance Computing (HPC) cluster and a hybrid Big Data-HPC cluster. The primary focus was on enabling distributed machine learning (ML) workloads using both MPI and Apache Spark technologies. The overall goal was to explore the performance, scalability, and practical aspects of distributed ML on standard datasets (e.g., MNIST) and real-world bioinformatics data.

**Methods**

**1. Cluster Setup**

* **Environment:** Three Ubuntu 24.04 virtual machines were deployed using VirtualBox (1 master node and 2 worker nodes).
* **Networking:** SSH was configured for passwordless communication across nodes, tested successfully via terminal access.
* **Documentation:** All setup steps and configurations were documented to ensure reproducibility.

**2. Mini-HPC Cluster with MPI**

* **Tools Used:** OpenMPI and mpi4py were installed on all nodes.
* **Execution:** Distributed training of the MNIST dataset was done using an MPI-based Python script (distributed\_mnist.py).
* **Parallelization:** Each process trained on a portion of the dataset, resulting in accurate and efficient parallel execution.
* **Results:** The MPI model achieved 94.2% accuracy, with a mean training time of 0.044 seconds and total execution time of 0.073 seconds.

**3. Hybrid HPC + Big Data Cluster with Spark**

* **Tools Used:** Docker Swarm was initialized on all three nodes.
* **Apache Spark:** Deployed in cluster mode using a Docker Compose YAML configuration file.
* **Cluster Verification:** Spark Web UI confirmed successful registration of worker nodes.
* **Distributed ML:** A PySpark script (distributed\_gene\_expression\_analysis.py) was used to train a Random Forest model on a leukemia gene expression dataset.
* **Cluster Resources:** Each Spark worker node was configured with 2 cores and 2GB RAM.

**Conclusion**

This project provided hands-on experience in deploying and working with both classical HPC and modern Big Data platforms. Key takeaways include:

* Effective configuration of MPI for scientific computing and parallel processing.
* Practical experience with Spark for scalable ML workflows on biological datasets.
* Understanding of Docker-based orchestration for Spark cluster deployment.
* Real-world implementation challenges in distributed ML, such as data partitioning, communication, and resource monitoring.

The project successfully demonstrated the integration of distributed computing concepts in bioinformatics and machine learning, forming a strong foundation for future work in scalable data science.