

ANALYSIS OF PARALLEL COMPUTING METHODS AND ALGORITHMS

¹Yusupova Janar, ²Davronov Murodjon, ³Choponov Otajon, ⁴Allayarov Shohzodbek, ⁵Omonov Sardorbek

¹Teacher of Urganch branch of TUIT named after Muhammad al- Khorazmi

² Programmist of PERSPECTIVE TEAM LLC

^{3,4,5}Students of Urganch branch of TUIT named after Muhammad al- Khorazmi

<https://doi.org/10.5281/zenodo.8226125>

Abstract. *Parallel computing is the process of executing multiple sets of instructions simultaneously. This reduces the total time to compute. Parallelism can be achieved by using parallel computers, which are computers with a large number of processors. Parallel computers require an operating system that supports parallel algorithms, programming languages, compilers, and multiprocessing.*

Keywords: *online judge, program, learning programming, online platforms, ACM ICPC.*

More than half a century ago, component development began to accelerate rapidly in the midst of time, leading to exponential growth in the capabilities of devices and equipment. This phenomenon was famously observed in Moore's Law - all critical indicators increased significantly - memory capacity at all levels, access time to memory, and processor speed. According to Moore's Law (one of the founders of Intel, Gordon Moore), characteristic values double every six months. The number of processors integrated into computers also increased. These changes largely contributed to the parallelization of calculations. One of the aspects related to the parallelization process was the reprocessing of instruction streams. Now, the process of executing instruction streams can be done in a parallel manner. The reprocessing of instruction streams was carried out in the pipeline, allowing several instructions to be prepared simultaneously. Instructions that are not dependent on each other can be executed in parallel, which is already true parallelism. Some computer architectures incorporate multiple processors, allowing them to perform logical and arithmetic operations on whole numbers. Several processors execute operations in parallel on floating-point numbers. A long instruction indicates the actions that each of the processors needs to perform. This provided the possibility of demonstrating operations that are required to be performed for vector and matrix processors at the hardware level. The set of instructions for these processors includes the basic operations on vectors and matrices. The parallel processing of data can significantly increase the efficiency of applications in this class. [1]

Sequential Computing

- Tasks are divided into discrete instructions.
- Instructions are executed sequentially.
- The tasks are executed one after another in sequence.
- Only one task can be executed at any given time.

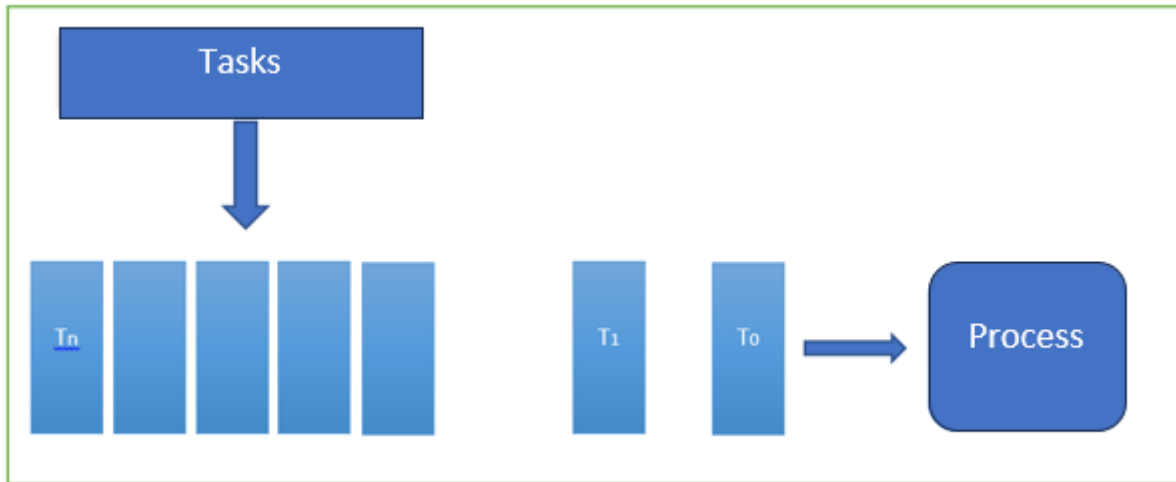


Figure 1. Sequential Computing.

Parallel Computing - this is the process of reprocessing multiple sets of instructions at the same time, reducing the overall computing time. Parallelism can be achieved using **parallel computers**, which are **computers** equipped with multiple processors. Parallel computers require parallel algorithms, programming languages, compilers, and an operating system that supports multitasking[2]. In simple terms, parallel computing means solving a computational problem by using multiple computing resources simultaneously:

- The task is divided into discrete parts that can be solved independently.
- Each part is further divided into additional instructions.
- The instructions of each part are executed on different processors simultaneously.
- A general control/synchronization mechanism is used to coordinate and manage the parallel execution.

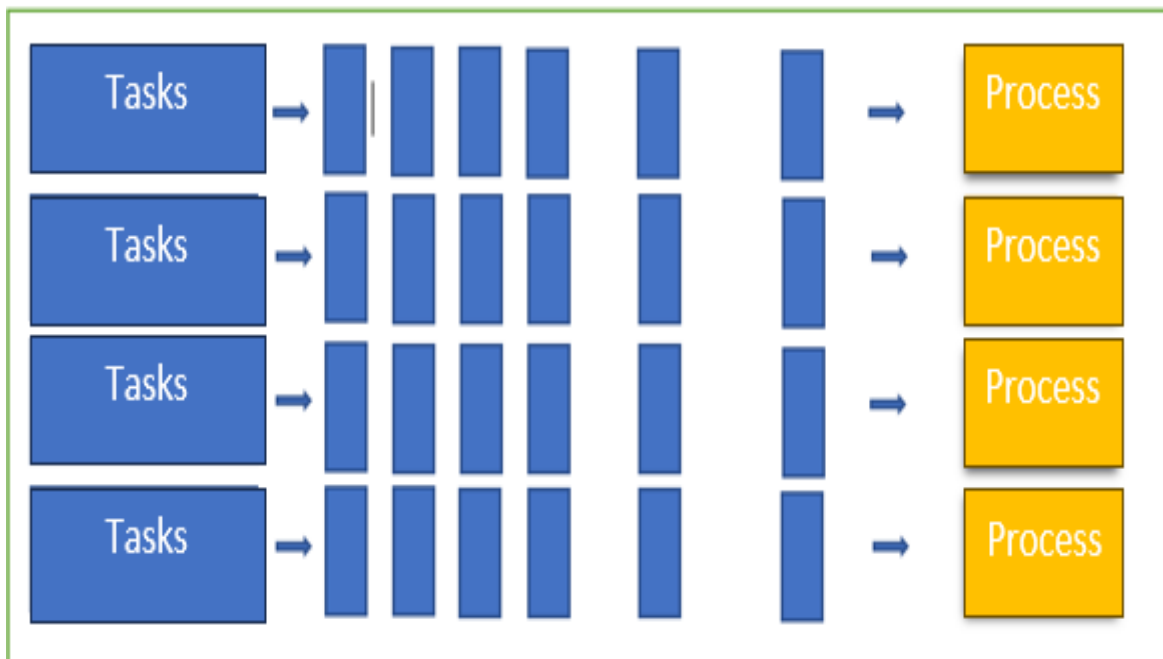


Figure 2. Parallel computing

Today, almost all personal computers are parallel in terms of hardware. Most computers that perform large parallel computing actions in the world are supercomputers. Supercomputers are generated from the aggregate of computers. They connect with one through the network and become a whole system.

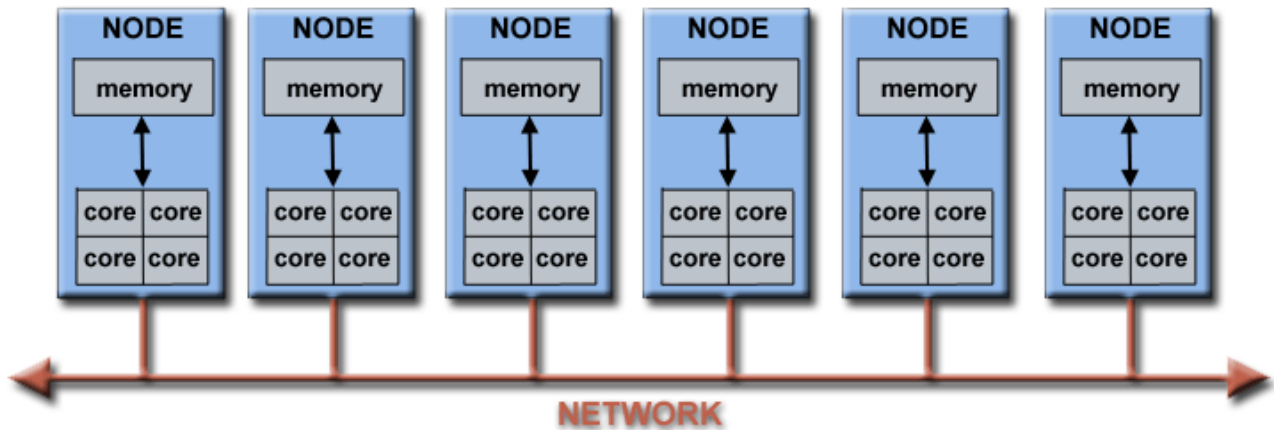
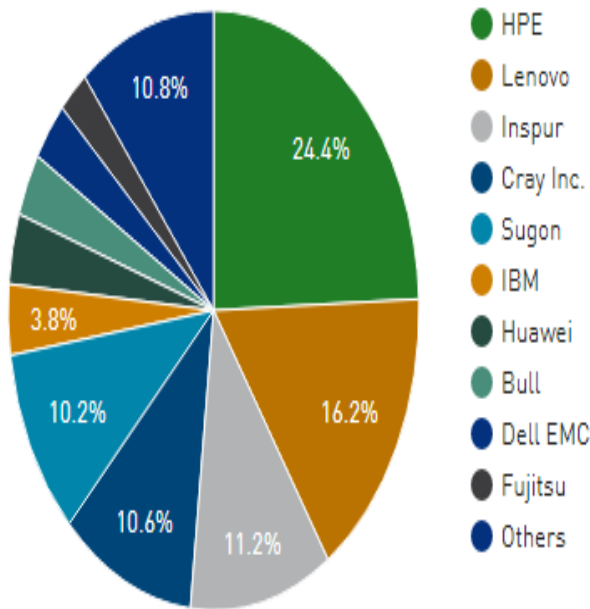


Figure 3. Connecting computers via network

The list of companies producing supercomputers in the world can be seen from Figure 4.

Vendors System Share



Vendors	Count	System Share (%)
HPE	122	24.4
Lenovo	81	16.2
Inspur	56	11.2
Cray Inc.	53	10.6
Sugon	51	10.2
IBM	19	3.8
Huawei	19	3.8
Bull	17	3.4
Dell EMC	16	3.2
Fujitsu	12	2.4
Penguin Computing	10	2
NUDT	4	0.8
PEZY Computing / Exascaler Inc.	4	0.8

Figure 4. Supercomputer manufacturing firms.

The power of supercomputers is measured by how many actions per second are performed. For example, the highest efficiency of a Hiot-based TaihuLight supercomputer device is 125.43 petaflops, which means it has the ability to perform 125,430,000,000,000 operations. By distributing such a calculation in parallel with assignments, it will be possible to fully use the capabilities of the supercomputer. Through Parallel computing, modeling, mimicking and understanding of real-world phenomena is appropriate. In recent years, trends shown by fast networks, distributed systems and multiprocessor computer architectures have made it clear that parallelism is the future of computing. During the same time, the performance of supercomputers increased more than 500,000 times, now its end is not visible.

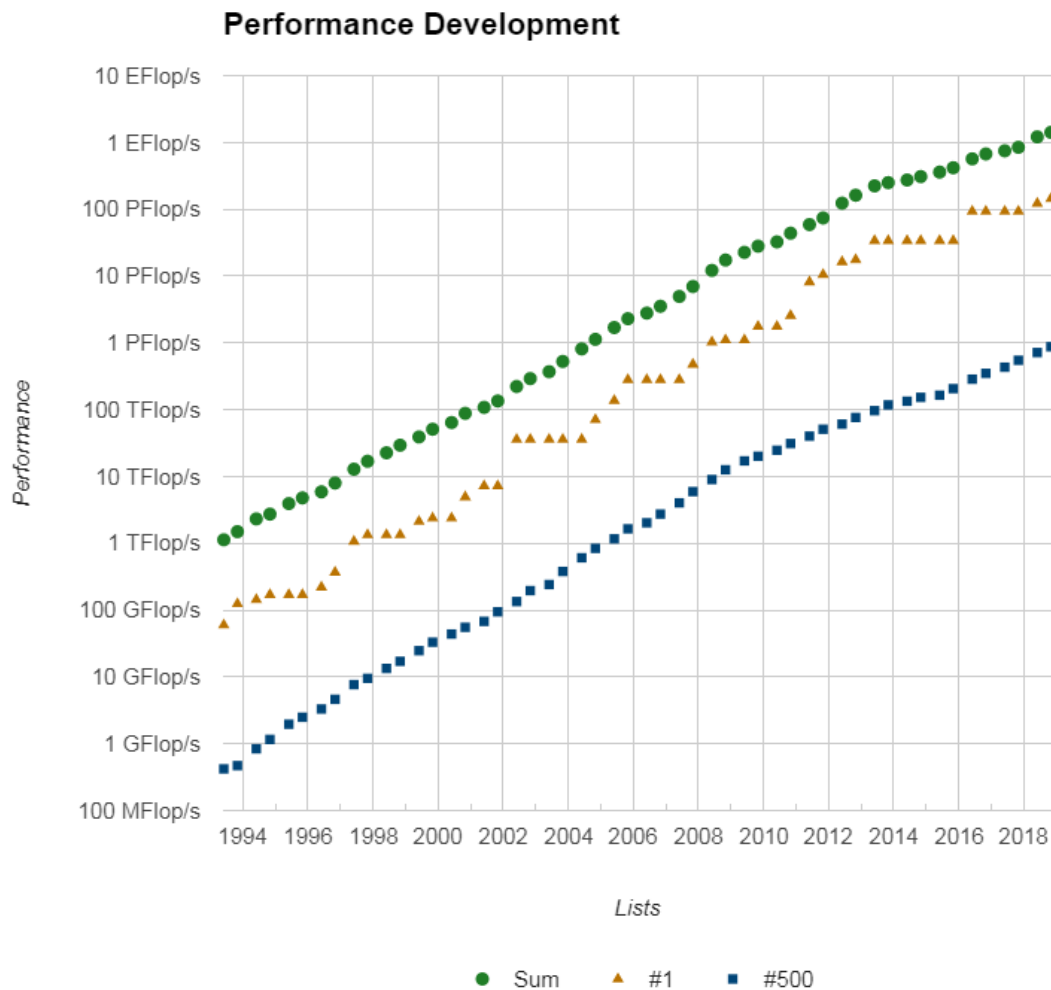


Figure 5. The increase in the speed of supercomputer calculations over the years.

Algorithm - is a sequence of precise instructions that must be followed to achieve a given result. An algorithm is not limited to computer-related tasks only; it can be applied to any process that can be carried out according to given instructions. It is a specific rule (program) for performing actions related to a certain type of problem, using the basic concepts of cybernetics and mathematics. The term "Algorithm" comes from the Latinized pronunciation of the name Al-Khwarizmi. In the 9th century, Muhammad ibn Musa Al-Khwarizmi introduced arithmetic operations in a decimal number system through a manual he created, which led to the adoption of the decimal number system in Europe. These rules were introduced as "according to Al-Khwarizmi" and, due to pronunciation over time, came to be expressed as "algorithm."

Currently, an algorithm is understood as a well-ordered series of clearly defined instructions that need to be followed to solve a problem or perform a specific task. The concept of an algorithm can be broadly analyzed. For instance, if we have a question about how to get from one address to another using city transportation, we can recommend a specific algorithm. In a cookbook, there are recipes that outline the rules for cooking various dishes. These cooking algorithms are also known as cooking recipes. In general, we mostly talk about algorithms related to computation. We define the signs and requirements specific to algorithms. Any algorithm must possess the following essential characteristics:

- Quality of being deterministic
- Obtaining a single-valued answer with given initial values;

Quality of being generalizable

Having the capability to find solutions with various initial values for a specific type of problem;

Quality of discreteness

The possibility of executing the algorithm on a computer (Electronic Computing Machines - EHM) or by a human without any uncertainty, in some specific straightforward steps.

Quality of productivity

The presence of a solution for any initial values, and in cases where there is no solution, it is accepted as the result of the algorithm's operation, for example, as "no solution exists."

Based on the mentioned qualities, it is possible to describe and articulate the rules for expressing and executing the algorithm. There are three main methods for representing algorithms in practice: textual representation, schematic (graphic) representation, and representation in some algorithmic language (programming language).

In developing an algorithm, it is necessary to consider the architecture of the computer on which the algorithm will be executed. In terms of architecture, computers can be classified into two types:

- Sequential computer
- Parallel computer [3]

According to the computer's architecture, we can have two types of algorithms:

- Sequential algorithm - An algorithm where sequential instructions are executed in chronological order to solve the problem.

- Parallel algorithm - A problem is divided into smaller sub-problems, and these individual tasks are executed in parallel to obtain individual results. Later, these individual outputs are combined to obtain the final desired result[5].

Dividing a large problem into smaller sub-problems is not always easy. Additional problems may arise due to their interdependencies. Therefore, processors need to communicate with each other to solve the problem.

It was found that the time required for processors to communicate with each other is much greater than the actual execution time. Thus, in developing a parallel algorithm, the proper utilization of the processor to achieve an effective algorithm should be taken into account [4].

CONCLUSION

Parallel computing is considered the "high gear" of computation and is used to model complex problems in various fields of science and engineering:

- Atmosphere, Earth, and space environments.
- Physics, nuclear physics, particle physics, condensed matter, high pressure, thermonuclear reactions, photonics.
- Biology, biotechnology.
- Chemistry, molecular sciences.
- Geology, seismology.
- Mechanical engineering - aerospace applications.
- Electrical engineering, circuit design, microelectronics.
- Computer science, mathematics.

In science and engineering, parallel computing is employed to solve the following problems:

- "Big data," databases.
- Artificial intelligence (AI).
- Web search engines, web-based business services.
- Medical imaging and diagnostics.
- Financial and economic modeling.
- National and multinational corporation management.
- High-performance graphics and virtual reality.
- Streaming video and multimedia technologies.

REFERENCES

1. GL Miller, R. Peng, and SC Xu. Parallel graph decompositions using random shifts. In SPAA, pages 196–203, 2013.
2. J. Reif. Optimal parallel algorithms for integer sorting and graph connectivity. TR-08-85, Harvard University, 1985.
3. Y. Shiloach and U. Vishkin. An $O(\log n)$ parallel connectivity algorithm. *J. Algorithms*, 1982.
4. Y. Gu, J. Shun, Y. Sun, and GE Blelloch. A top-down parallel semisort. In SPAA, 2015.
5. J.Yusupova, O.Choponov, Sh.Allayarov. “Parallel data testing on "online hakam" systems for programming students”. *Science and innovation International scientific journal*, pages 331-334, 2023. <https://doi.org/10.5281/zenodo.8102514>.